Key Action: KA2: Cooperation for Innovation and the Exchange of Good Practices, KA201 - Strategic Partnerships for school education
Project name: STEAM education and learning by Robotics, 3D and Mobile technologies - FabLab SchoolNet
Project No.: 2018-1-LT01-KA201-047064

INTELLECTUAL OUTPUT 7 - FABLAB SCHOOL NET IMPLEMENTATION

Open / online / digital education – E-learning course / module	
Activity Leading Organisation	2 EPAL TRIKALON
Participating Organisations	Siauliu Didzdvario gimnazija Consiglio Nazionale delle Ricerche Universitatea “Dunarea de Jos“ din Galati Varnenska morska gimnazia "Sv. Nikolai Chudotvorec" FabLab Palermo APS

Due Submission Date:	28/01/2019
Actual Submission:	30/10/2021
Project Number	2018-1-LT01-KA201-047064
Instrument:	Strategic Partnerships for school education
Start/Finish Date of Project:	01.11.2018 – 31.10.2021
Duration:	36 months

Abstract

This output is the main outcome of the project. A course to apply the FabLab Schoolnet methodological approach was implemented at large scale. The course consisted of several units, which cover the entire path to use innovative technologies in real contexts. The course was specialised on the three technologies (robotics, 3D printing, mobile devices) that was developed in the project. The lessons learned in the pilot course (O6) were applied, in order to be more effective from a didactic perspective.

As an example, for the 3D technology, the course started with basic concepts, such as the various types of 3D printers on the market and their characteristics, the attention being focused on the process of computer assisted 3D modelling, using special software, printing processes, different types of materials and critical issues.

As a common learning path, all students were introduced to the basic concepts of self-entrepreneurship and marketing, in order to make the learner immediately ready for the world of work. Moreover, models were also used to support non-formal learning and encouraging the creativity of learners. In this output a platform for sharing best practices in using FabLab Schoolnet technologies was created.

This platform was used for sharing the experiences of the schools in applying the FabLab School net approach. This social platform has the aim of collecting all the learning experiences carried out by the schools involved in the project. The analysis of these experiences will highlight good practices in the use of FabLab School net technologies in educational contexts providing good examples to foster the diffusion of FabLab network amongst European schools.

Open / online / digital education – E-learning course / module

Activity Leading Organisation	2 EPAL TRIKALON
Participating Organisations	Siauliu Didzdvario gimnazija Consiglio Nazionale delle Ricerche Universitatea “Dunarea de Jos“ din Galati Varnenska morska gimnazia "Sv. Nikolai Chudotvorec" FabLab Palermo APS

CONTENT

Chapter 1. FABLAB SCHOOLNET IMPLEMENTATION IN GREECE

Robotics 1: Creating light shows

Description:

The students create light shows changing the brightness and the color of the robot LED's.

Learning Objectives:

- to understand what a loop is
- to be more familiar with the hardware of mbot robotics
- to build and execute iteration structures using robotics

Expected results:

The students will be able to create programs using their computer, connect the robots and run the code. They will recognize the LED's of the robot and create iteration structures changing the brightness and the color of LED's.

Key issues:

programming, robotics, iteration structure

Technologies:

mBot Ranger Robotics

Software:

mBlock

Age of students:

16-18

Number of students

70 (6 classes)

Didactic Hours:

2 per class

Assessment:

The students in each class were divided in 4 teams and used laptops in order to create the code. The students acquired the necessary skills for their autonomous use of the software and the robots. They expressed their satisfaction for the educational material and enjoyed the procedure of creating a light show. Some technical problems were reported about the connection between the laptop and the robot.

YouTube Link:

https://www.youtube.com/watch?v=5QK23iGbUxU

Worksheet for the students

Introduction to the RGB LEDs of mBot Ranger

The 12 RGB LEDs in Me Auriga of mBot Rangers are mounted in a circle. Each RGB LED can be programmed to control the brightness of three colors (red, green and blue) and combine these three colors to produce different colors of light.

How to control the RGB LED with blocks:

The "All" option determines the number of RGB LEDs . The default value of this tab is "all". The "all" option means that we can control all 12 RGB LEDs in Me Auriga . When we select eg "2", it means that we can only control the 2nd RGB LED in Me Auriga .

Option [0] controls the brightness of red, green and blue in the range 0 to 255. "0" means no output and the LED is off. "255" is the maximum output and the indicator light is fully activated.

By setting values for these three colors you can create different light colors.

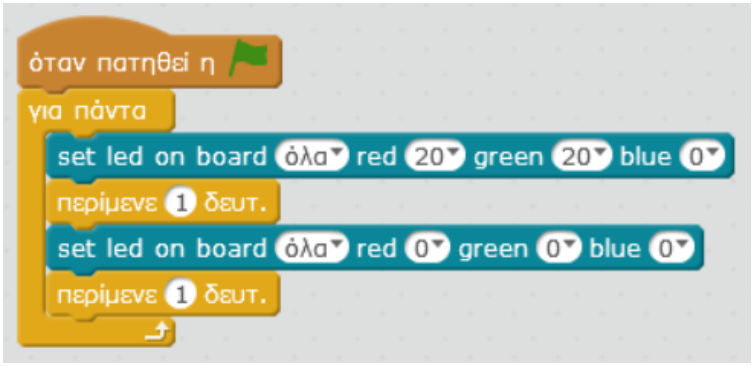
Try the command:

Task 1

If you want some commands to be executed continuously, then you place them in a "forever" command, which you find in the "Control" command group.

Because of the changes in the LEDs are done quickly, you can use the "wait" command, which you find in the "Control" command group.

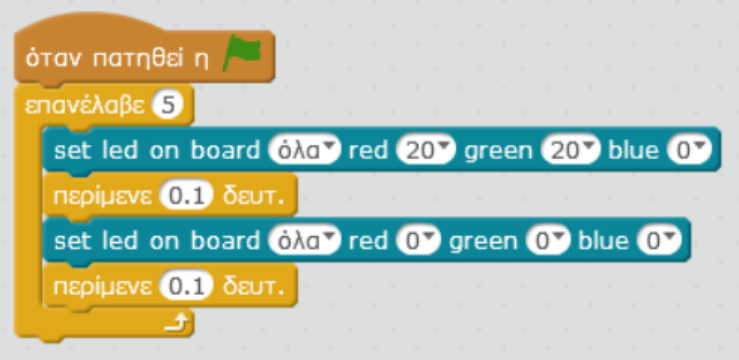
Try the following program. What do you notice?



Task 2

If you want some commands not to be executed forever but for a certain number of iterations, then use the "repeat" command

Try the following program. What do you notice?



Task 3

Change the previous program so that the red LEDs flash first 5 times, after the green LEDs 5 times and then the blue LEDs 5 times.

Task 4

Can you work with the lamps individually? Change the "all" option and create a program where the lights will light up in order, for example (first the 1st, then the 2nd, etc.). Make your own light show !

Robotics 2: Working with the Light Sensors of the robot

Description:

The students create code making the robots execute different commands depending on the value of the light sensor.

Learning Objectives:

to understand what a light sensor is

to understand what a variable is

to be more familiar with the hardware of mbot robotics

to use variables to store the data of a light sensor and interact with the robot

to build and execute selection structures using robotics

Expected results:

The students will be able to create programs using their computer, connect the robots and run the code. They will recognize the light sensors of the robot and create selection structures changing the brightness and the color of LED's.

Key issues:

programming, robotics, selection structure

Technologies:

mBot Ranger Robotics

Software:

mBlock

Age of students:

16-18

Number of students

70 (6 classes)

Didactic Hours:

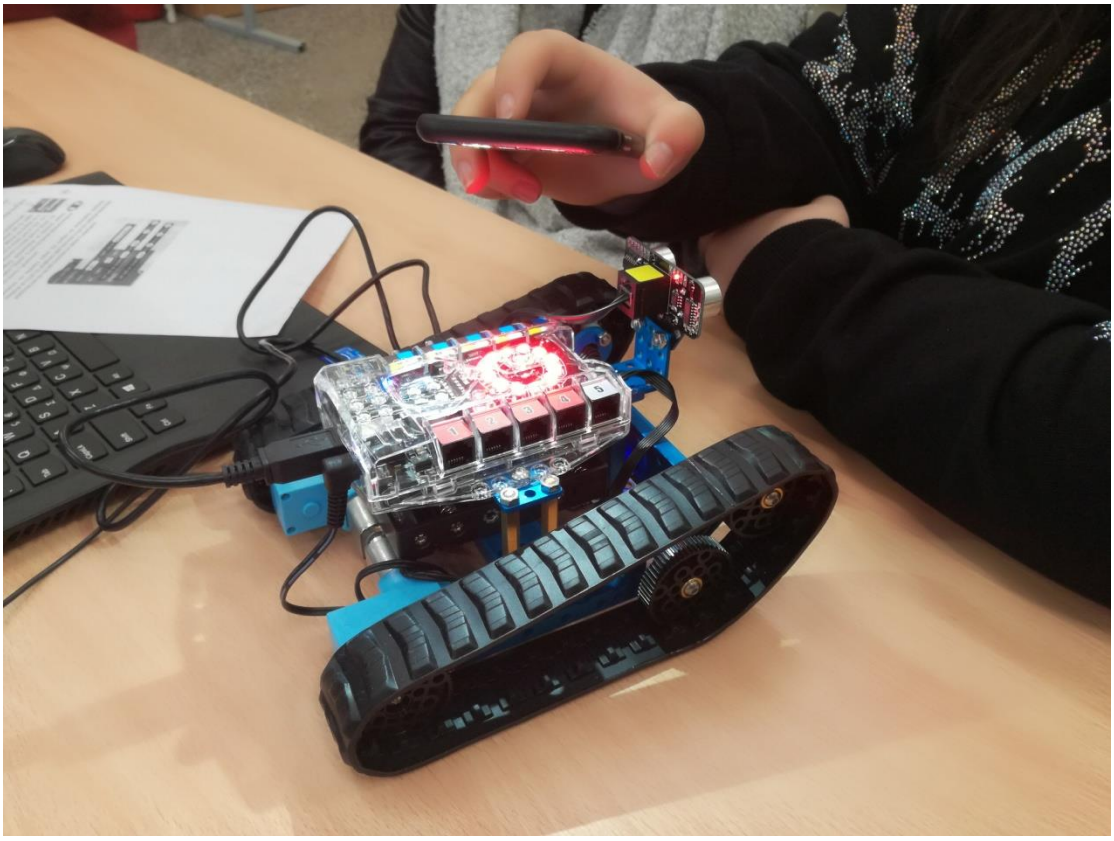
2 per class

Assessment:

The students in each class were divided in 4 teams and used laptops in order to create the code. The students acquired the necessary skills for their autonomous use of the software and the robots. They expressed their satisfaction for the educational material. No technical problems were observed.

YouTube Link:

https://www.youtube.com/watch?v=5QK23iGbUxU



Worksheet for the students

The Light Sensor of the mBot Ranger

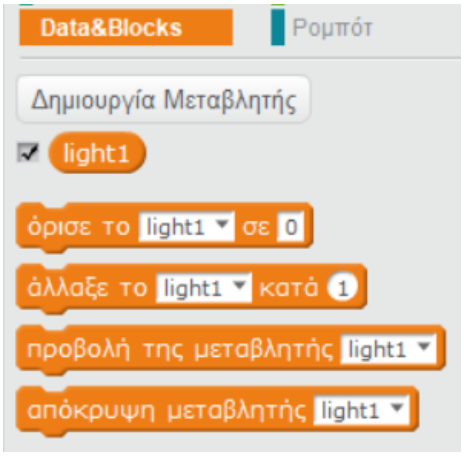
The Me Auriga of mBot Ranger has integrated two Light Sensors. To display the value of a Light Sensor we will use the corresponding tile from the "Robot" group in the mBlock program .

Sensor values range from 0 to 970. The "on board 1" option corresponds to light sensor 1 of Me Auriga while the "on board 2" option corresponds to light sensor 2.

Task 1

Create a variable for the sensor value

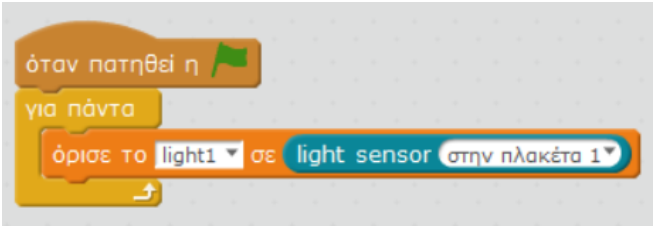
We will create a variable in which we will store the value of the sensor. Variables are used to store numbers or text in memory. To create a variable, go to the " Data & Blocks " group and click the "Create Variable" button. In the box that opens, enter the name of the variable. Name the variable light1 for this activity and press ok. In the next image we now see the available commands for this variable light1.



Task 2

Saving the sensor value variable to the variable

Then create the following code:



Connect the robot and press the green flag. You will notice that the light sensor values are displayed in the variable light1. If you cover the light sensor 1 with your hands, you will notice that the value of the light1 variable in the upper left corner of the scene is constantly changing. The closer your hand is to the light sensor, the less light the sensor detects, so the value of the variable will be lower.

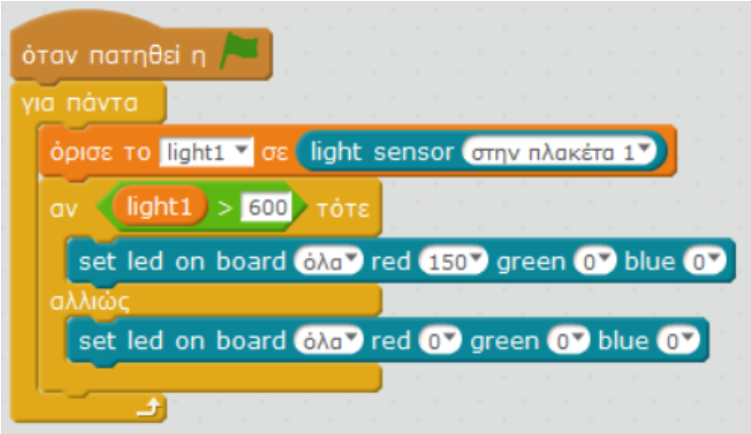
Task 3

The robot can execute different commands depending on the value of the light sensor. For example, we will create a program with which if the lighting is bright then the LEDs of the robot will turn red otherwise they will go out.

We decide that the lighting is considered bright when the value of the light sensor is greater than 600.

To check if the value of the light sensor is greater than 600, we use the command "if .. then .. else" from the command group "Check". Between "if" and "then" we put a condition. If the condition is true then the commands after "then" are executed. If the condition is false then the commands after "else" are executed. To create the condition we use the hexagonal green tiles from the group "Operators". In this exercise we use the tile with the operator > (larger).

Generate the following code. In the condition of this case we check if the value of the variable light1 (which is the value of the light sensor) is greater than the value of 600. Execute the code. What do you notice?



Robotics 3: Working with the Ultrasonic Sensor of the robot

Description:
The students create code making the robots move to different directions depending on the value of the ultrasonic sensor.

Learning Objectives:
to understand what a ultrasonic sensor is
to understand what a variable is
to be more familiar with the hardware of mbot robotics
to use variables to store the data of a ultrasonic sensor and interact with the robot
to build and execute selection structures using robotics

Expected results:
The students will be able to create code using their computer. The programs will be executed by the robot autonomously. They will recognize the ultrasonic sensor of the robot and create selection structures changing the movements of the robot.

Key issues:
programming, robotics, selection structure, sensors

Technologies:
mBot Ranger Robotics

Software:
mBlock

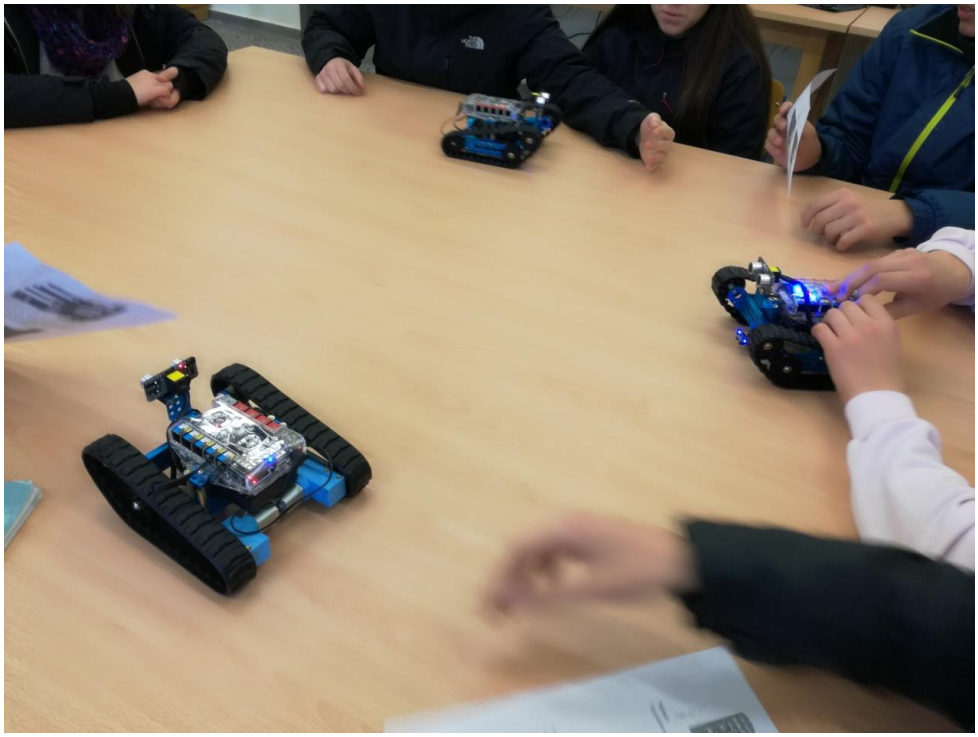
Age of students:
16-18

Number of students
70 (6 classes)

Didactic Hours:
2 per class

Assessment:
The students in each class were divided in 4 teams and used laptops in order to create the code. The students acquired the necessary skills for their autonomous use of the software and the robots. They expressed their satisfaction for the educational material and enjoyed the procedure of making the robot move. Some technical problems were reported about the connection between the laptop and the robot and the function of the motors.

YouTube Link:
https://www.youtube.com/watch?v=5QK23iGbUxU



Worksheet for the students
The Ultrasonic Sensor of the mBot Ranger

The ultrasound unit of the mBot Ranger consists of a transmitter, a receiver and a control circuit. When the ultrasonic transmitter receives a command, it emits high frequency sound waves. When the reflected sound waves are received from the receiver, Auriga calculates the elapsed time and converts the data into distance.

To display the value of Ultrasonic Sensor we will use the corresponding tile from the "Robot" group in the mBlock program . The default value "Port10" corresponds to the port to which the ultrasonic sensor is connected and of course can be changed.

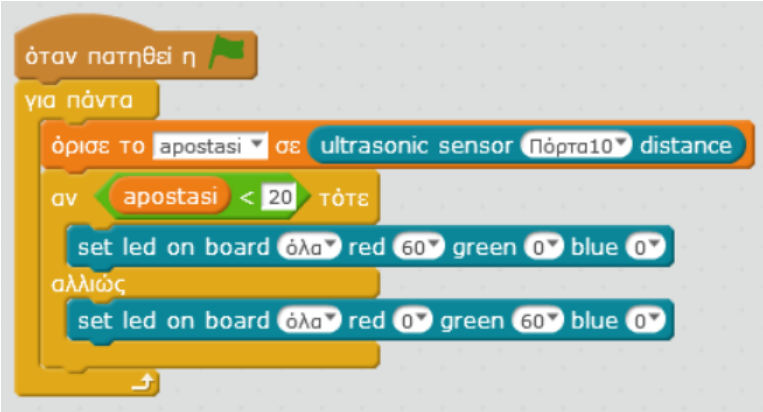
Task 1
Save the sensor value to a variable

In the "Data & Blocks" group, click the "Create Variable" button and create a variable named "apostasi". In this variable we will store the value of the sensor. Generate the following code:

Place your hand in front of the sensor and observe the change in the value as your hand approaches or moves away from the sensor. The value indicates the distance between your hand and the ultrasonic sensor. The price ranges from 3 to 400 cm.

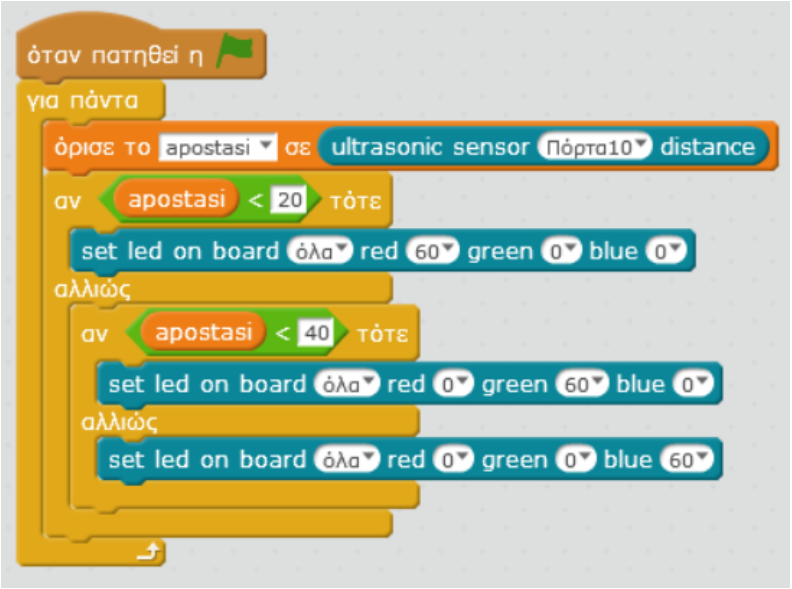
Task 2

The robot can execute different commands depending on the value of the ultrasonic sensor. For example, we will create a program with which if the distance between your hand and the ultrasonic sensor is less than 20 cm then the robot's LEDs will turn red otherwise they will turn green. Generate the following code.



Task 3

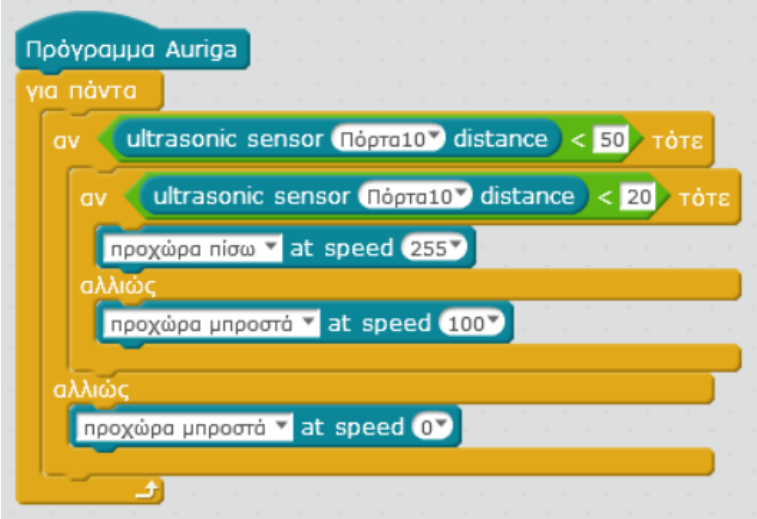
Generate the following code. What do you notice?



Task 4

When we want the program to be executed by the robot autonomously (without the robot being connected to the computer) then we use the "Auriga Program" as the first tile. After creating the code, right-click on "Auriga Program" and then select "upload to arduino". Then

press the "Upload to Arduino" button and the program is transferred and saved to the robot. Generate the following code. What do you notice?



Robotics 4: Moving the robot with the Line-Follower Sensor

Description:

The students create code making the robots move on a black line depending on the value of the Line-Follower Sensor.

Learning Objectives:
to understand what a Line-Follower Sensor is and how it works
to understand what a variable is
to be more familiar with the hardware of mbot robotics
to use variables to store the data of a Line-Follower sensor and interact with the robot
to build and execute selection structures using robotics

Expected results:
The students will be able to create code using their computer. The programs will be executed by the robot autonomously. They will recognize the Line-Follower sensor of the robot and create selection structures making the robot move on a black line.

Key issues:
programming, robotics, selection structure, sensors

Technologies:
mBot Ranger Robotics

Software:
mBlock

Age of students:
16-18

Number of students
70 (6 classes)

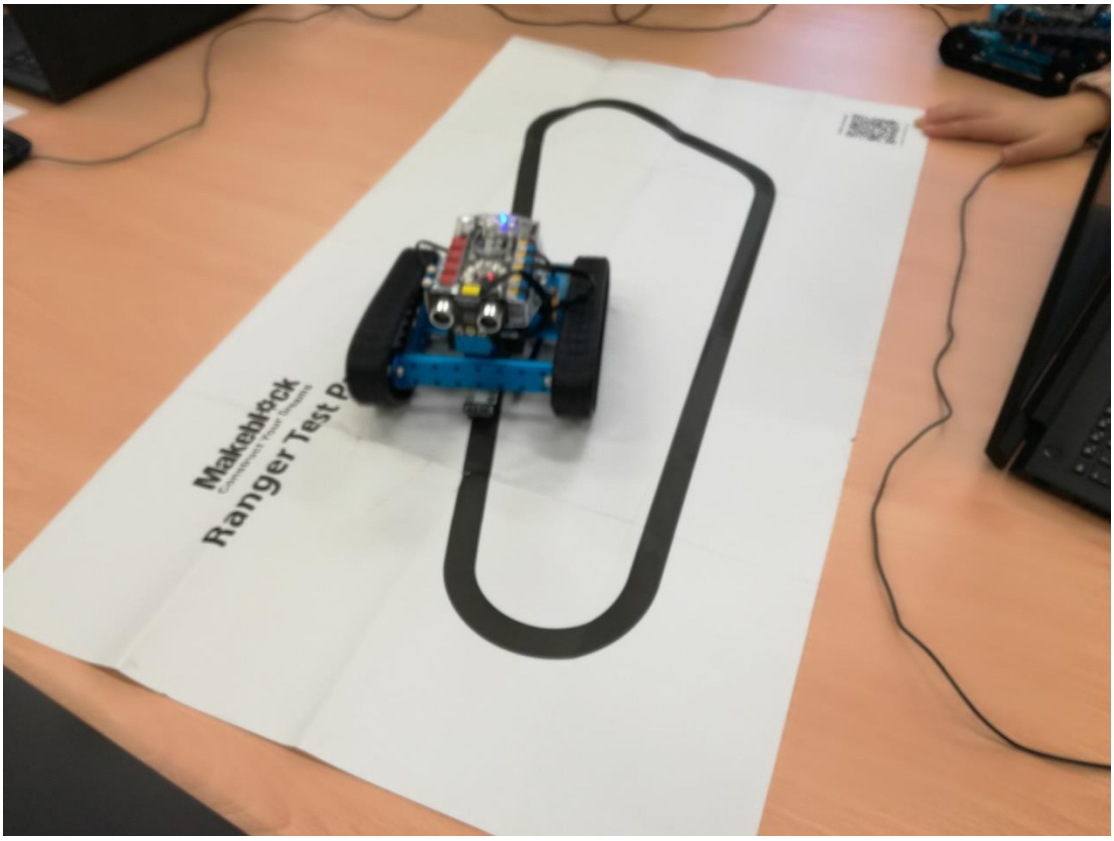
Didactic Hours:
1 per class

Assessment:
The students in each class were divided in 4 teams and used laptops in order to create the code. The students acquired the necessary skills for their autonomous use of the software and the robots. They expressed their satisfaction for the educational material and enjoyed the

procedure of the lesson. They experimented and tried to create the most effective code. No technical problems were reported.

YouTube Link:

https://www.youtube.com/watch?v=5QK23iGbUxU



Worksheet for the students

The Line-Follower Sensor of the mBot Ranger

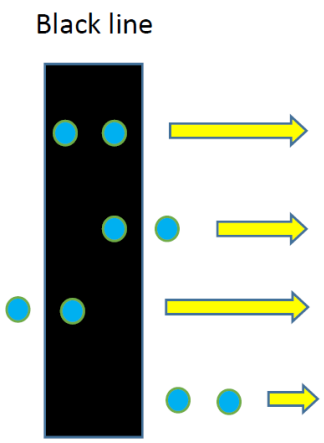
The Line-Follower Sensor will be used to move the mBot Ranger on a black orbit. To display the value of the Line-Follower Sensor we will use the corresponding tile from the "Robot" group in the mBlock program . The default value "Port9" corresponds to the port to which the sensor is connected .

Task 1

Save the sensor value to a variable

In the "Data & Blocks" group, click the "Create Variable" button and create a variable named "grammi". In this variable we will store the value of the sensor. Generate the following code:

Possible values are 0, 1, 2, 3. Put the robot so that the sensor is directly on the black line. The value should be 0. Move the robot left and right a little and see the changes in the value.



Left side sensor 1	Right side sensor 2	Line-follower sensor value
Black	Black	0
Black	White	1
White	Black	2
White	White	3

Task 2

Movement of the robot on the black orbit

Generate the following code. What do you notice?

3D Printing 5: Using the technology

Description:

The purpose of the lesson is to learn the basic concepts of designing and printing of 3D models. At the end of the course the students must be able to create a real object using the 3D printer. The modules of the course are:

- introduction to 3D printing technology by the teacher with the following topics: How 3D printers work, what can be made with a 3D printer, searching for 3d models in the world wide web (Thingiverse, MyMinifactory).
- design using TinkerCad Software. The teacher creates classes and nicknames for the students in the TinkerCad platform in order to organize the learning procedure. The link with the class code and the nicknames are shared to the students. The students visit the tinkercad website, watch tutorials and study in order to learn how to create a 3D model.
- slicing with Cura Software. The students print their own 3D model.

The shared presentations were produced during the implementation of the FabLab project.

Learning Objectives:

Students acquire knowledge including:

- the parts and the way that 3d printers function
- the materials used in 3d printing
- the areas of human activities in which 3d printing is used
- the available web free 3d design tools
- use of 3d applications and their tools, to design models for 3d printing including tinkercad

Students also acquire skills like

- operate a 3d printer
- set the proper properties to the printing software, in order to have a reliable print out
- prepare their 3d models for 3d printing
- recognize and to use productively the basic tools of a 3d design application.
- search the web to find proper 3d applications that cover their design needs.
- use the mouse and hot keys to operate and move in a 3d design space

Expected results:

At the end of the course students must be able to

- name and describe the basic parts of a 3d printer and the basic printing materials and 3d printing techniques
- describe the basic services of a 3d printing software
- describe the capabilities of a 3d modeling system
- name and describe the common functions of a 3d design software
- model and print their own creations

- Find designs online and create their own

Key issues:
3D design, 3D printer, creativity, 3D modelling

Technologies:
3D Technologies

Software:
TinkerCad, Cura

Age of students:
16

Number of students
70 (6 classes)

Didactic Hours:
6 per class

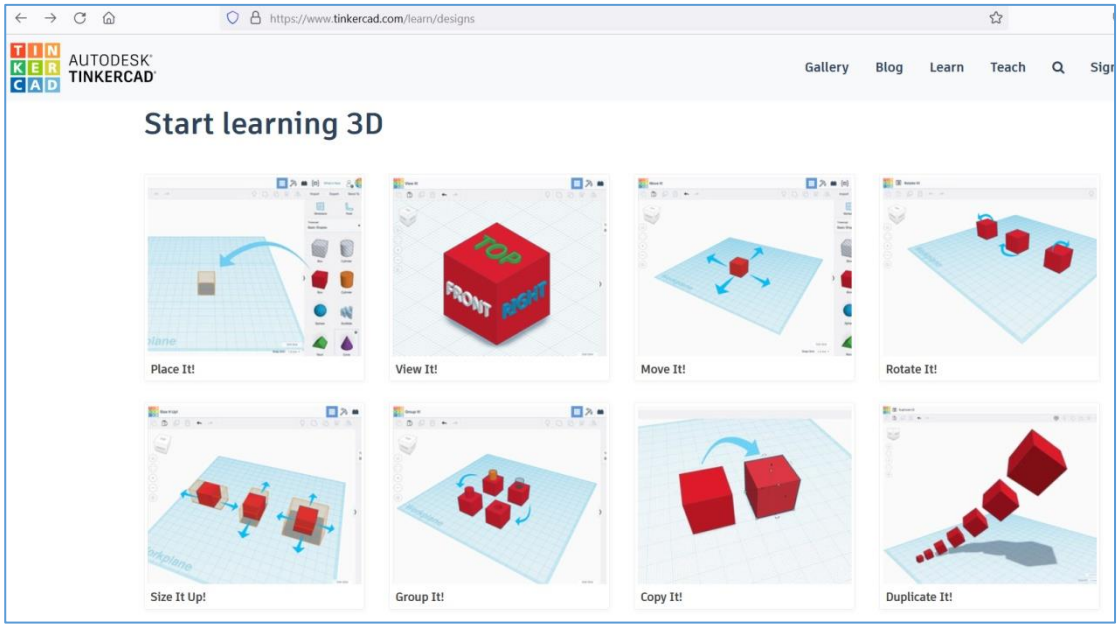
Assessment:
The training course aimed at 70 students (6 classes) from the 1st grade of the school. It was implemented in the School FabLab. The students used computers and laptops to have access to the presentations and design their 3D model with Tinkercad. They created the gcode by the "slicing" process with Cura Software installed at the lab server and printed with the Ultimaker 3D printer of the School FabLab. The students acquired the necessary skills for their autonomous use of 3D modeling software and created physical objects starting from original ideas. They used their imagination and creativity having positive feelings and enthusiasm during the lessons. The students expressed their satisfaction for the educational material and their joy that they were able to design and print their own object. No technical problems were observed.

Worksheet for the students

Task 1

Visit tinkercad.com website using the shared link with your class code. Type your nickname in order to have access to the platform. Click on Learn menu. You are going to learn how to create a 3D model. Place objects on the surface, create holes and learn how to move, rotate, resize and group them. You will also learn how to download your design and save it as a .stl file.

Link: https://www.tinkercad.com/learn/designs



Task 2

Design your 3D model using your imagination and creativity. The object can be a gadget like a keychain with your name. Download the .stl file.

Open it with Cura software and scale it 70%, see your printing options.

Save the file in .gcode extension and upload it.

Augmented Reality: Learning about Internet and World Wide Web with AR

Description:

The object of the lesson is the process of teaching Informatics through the use of augmented reality. The chosen subject is titled "Internet and World Wide Web". The students learn about the history and the structure of the Internet and the World Wide Web and study about the most popular internet services. The teacher creates AR experiences enriching the content of the school book. The teacher designs markers like QR codes, text and images from the school book. The markers provide animations, images, 3D models, videos available on youtube channels and other repositories including the FabLab Learning Repository. The teacher adds the resources on the AR Lectio web platform and installs AR Lectio app in the school mobile devices (tablets).

After the announcing of the theme of the lesson students are encouraged to watch videos, carry out experimental tasks with the help of devices and study discovering the AR content on the school book.

Learning Objectives:

to acquire knowledge about the history and the structure of the internet
to recognize the most popular internet services
to promote an active response with the content of the book
to motivate students to be engaged in the learning process

Expected results:

Using AR technology increases learning efficiency, facilitates students' training and cognitive activities, improves the quality of knowledge acquisition, provokes interest in a subject, promotes development of research skills. AR can add gamification to the learning process, grow student's motivation and positively influence their learning achievements.

Key issues:

Augmented Reality, Internet, World Wide Web

Technologies:

AR Technologies, Mobile Devices

Software:

AR Lectio

Age of students:

16

Number of students

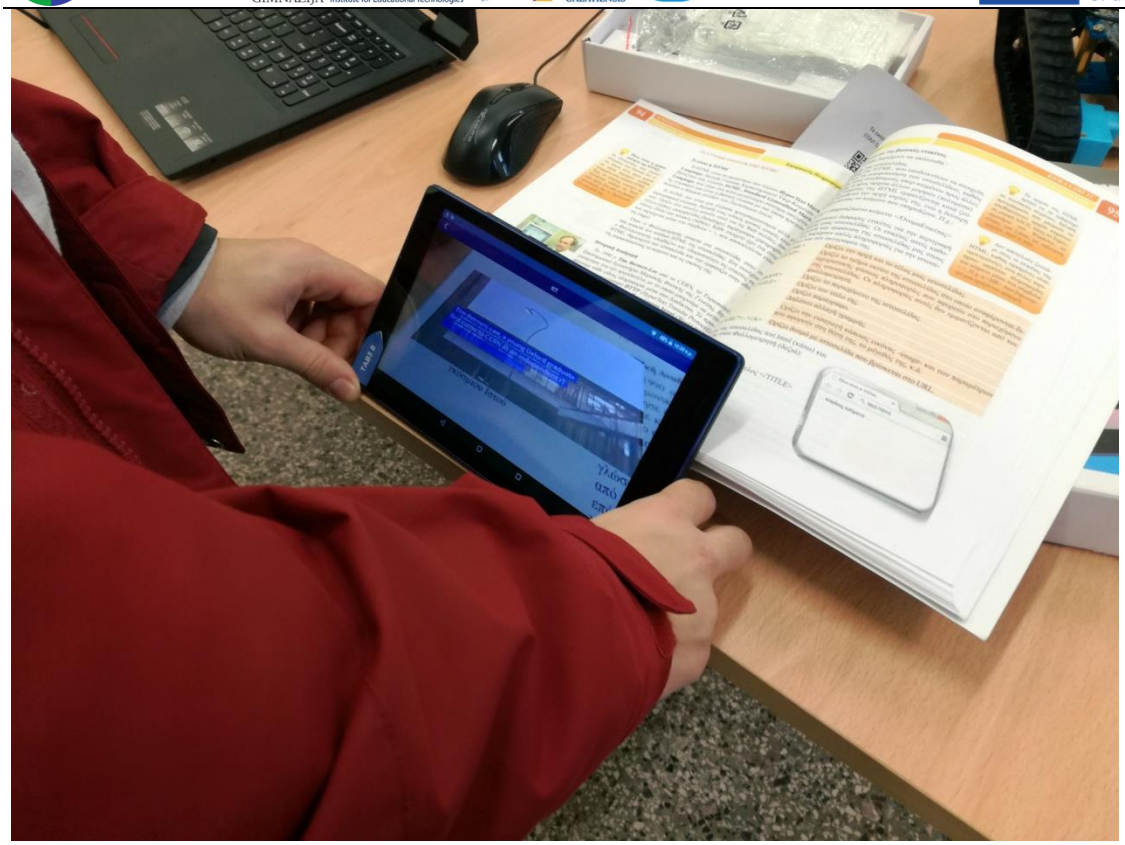
70 (6 classes)

Didactic Hours:

1 per class

Assessment:

The training course aimed at 70 students (6 classes) from the 1st grade of the school. It was implemented in the School FabLab. The students used the school tablets to have access to the enriched content of the school book. At the end of the study, it is revealed that the students have a positive attitude towards AR applications. They want to use this kind of application in other courses as well. They indicate satisfaction with the application. No technical problems were observed.



Integration of the Technologies. The Odyssey project

Description:

The concept of the lesson is based on the visualization of the story of Odysseus. The Odyssey is one of two major ancient Greek epic poems attributed to Homer. It follows the Greek hero Odysseus, king of Ithaca, and his journey home after the Trojan War. The aim of the game is to help Odysseus travel from Troy back to Ithaca avoiding all the obstacles and the difficulties that he faced during his journey. The students are instructed to create the board for the robotic contest, which includes the obstacles, the hero and his companions came across in their journey. The obstacles, which according to the tale of Homer were mythical creatures and forces of nature, will be created using traditional techniques, such as clay modeling, as well as new technologies such as 3D printing. For inspiration, the students will be shown short video animations and will be handed aiding prints with depictions of those mythical creatures. Each team is in charge of one robot to follow the robot mission. Robot mission is to move around the play-board helping the hero (Odysseus) to go from a starting point (Troy) to a destination area (Ithaca). The students have to construct and program the robots. In addition, the students have to design and print 3D models for the hero and the other objects on the board. AR technology is also used to create markers and AR experiences enriching the content of the board.

Learning Objectives:

to improve their knowledge and skills about the 3D, Robotics and Mobile technologies
to enhance their knowledge about Greek culture, history and mythology.

Expected results:

The implementation of the activities promotes creativity, imagination, communication, teamwork with different learning methods. The students have the chance to explore, discover, experiment, discuss, collaborate and solve problems. Artistic, IT and communication skills will be improved.

Key issues:

learning by doing, learning by making, project based learning

Technologies:

LEGO Robotics, 3D Technologies, Mobile Technologies

Software:

TinkerCad, Cura, Lego programming software, AR Lectio

Age of students:

16-18

Number of students

20

Didactic Hours:
10

Assessment:
The students enjoyed the implementation of the project using all the technologies offered by the FabLab. They expressed their satisfaction because they had the opportunity to work as a team and managed to solve different kind of problems. The students acquired the necessary skills for their autonomous use of the software and the hardware. No technical problems were reported.

YouTube Link:
Technology activities during the students meeting in Trikala, Greece - YouTube

Worksheet for the students
ART ACTIVITIES
Task 1
Design the board for the robotic game. Use paper, markers, pens etc to draw three starting points, three destination areas, three boxes for the moveable objects and six boxes for the forbidden areas (obstacles). In addition, use your imagination and design different routes for the robots to follow (easy, medium and hard level).

Task 2
Create the obstacles using clay and following the given instructions. Put the obstacles on the forbidden areas of the board.
Suggestion: The obstacles can be mythical creatures and forces of nature like Circe, Symplegades, Scylla, Cyclop, Laestrygonians, Aeolus etc

3D TECHNOLOGY ACTIVITIES
Task 3
Discover stl files for 3D printing about Odyssey on the World Wide Web. Some popular websites are Thingiverse (https://www.thingiverse.com/) and Myminifactory (https://www.myminifactory.com/).
Suggestion: Scan The World (https://www.myminifactory.com/users/Scan The World). Scan The World is an initiative introduced by MyMiniFactory that is creating a digital archive of sculptures, landmarks and monuments from around the world using 3D Scanning and Printing technology. You can find 3D scanned sculptures of Odysseus, Penelope, Cyclop etc.

Task 4
Design your own 3D models using the web app Tinkercad. Use your imagination and creativity in order to design the heros and the other objects for the board.
Suggestion: The castle of Troy, Ithaca Palace, the Odysseus' boat etc.

Task 5

Download your 3D models as stl files. Use Cura software for slicing and print your models at the 3D printer of the school's FabLab. Put the printed objects on the board.

Task 6

Scan your artworks using the 3D scanner of the school's FabLab.

ROBOTICS TECHNOLOGY ACTIVITIES

Task 7

Construct the Lego Mindstorms EV3 Robots of the school's FabLab considering the design of the board. The robots could include the following line sensor, obstacle detection sensor (ultrasonic), lateral deflection sensor (Giro sensor), optical or display signaling, acoustic alarms, or vocal commands using / sound sensor / microphone. In addition, a color sensor could be added for different other indications.

Task 8

Create programs using the Lego Programming Software in order to make the robots follow the appropriate route. The robot's mission is to move from the starting point, grab the hero from his area and take him to the destination area avoiding the forbidden areas with the obstacles.

MOBILE TECHNOLOGY ACTIVITIES

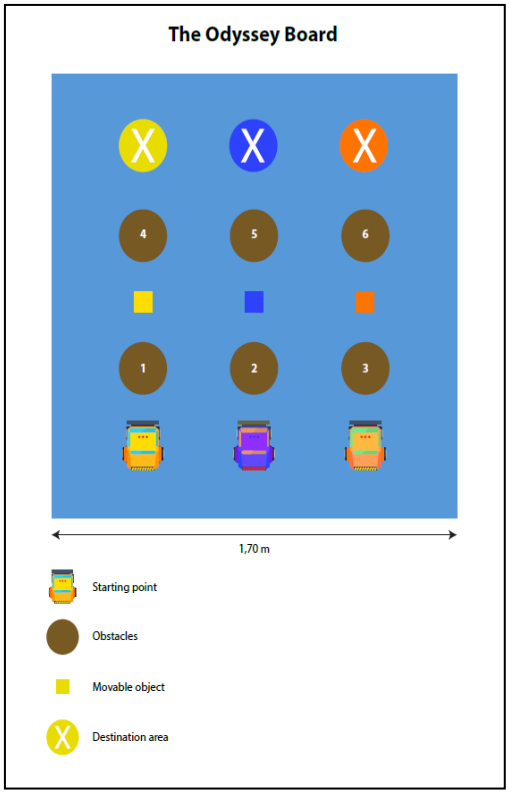
Task 9

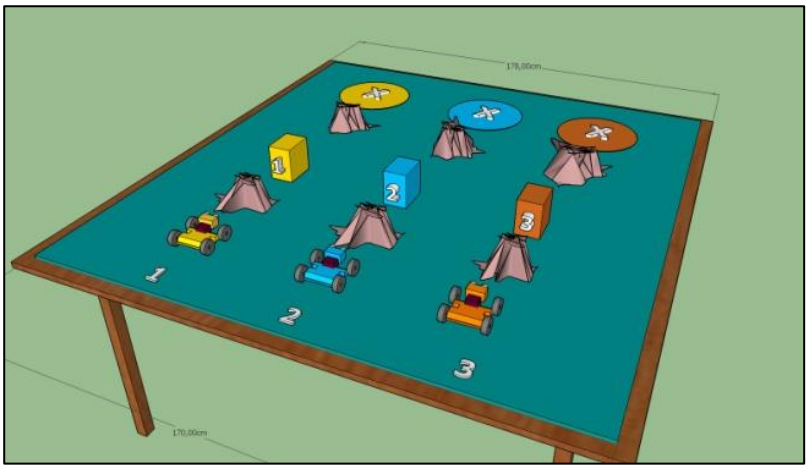
Enrich the board creating AR experiences. Design markers like QR codes, text etc and place them on the board. These markers can show the instructions for the routes, images, 3D models etc.

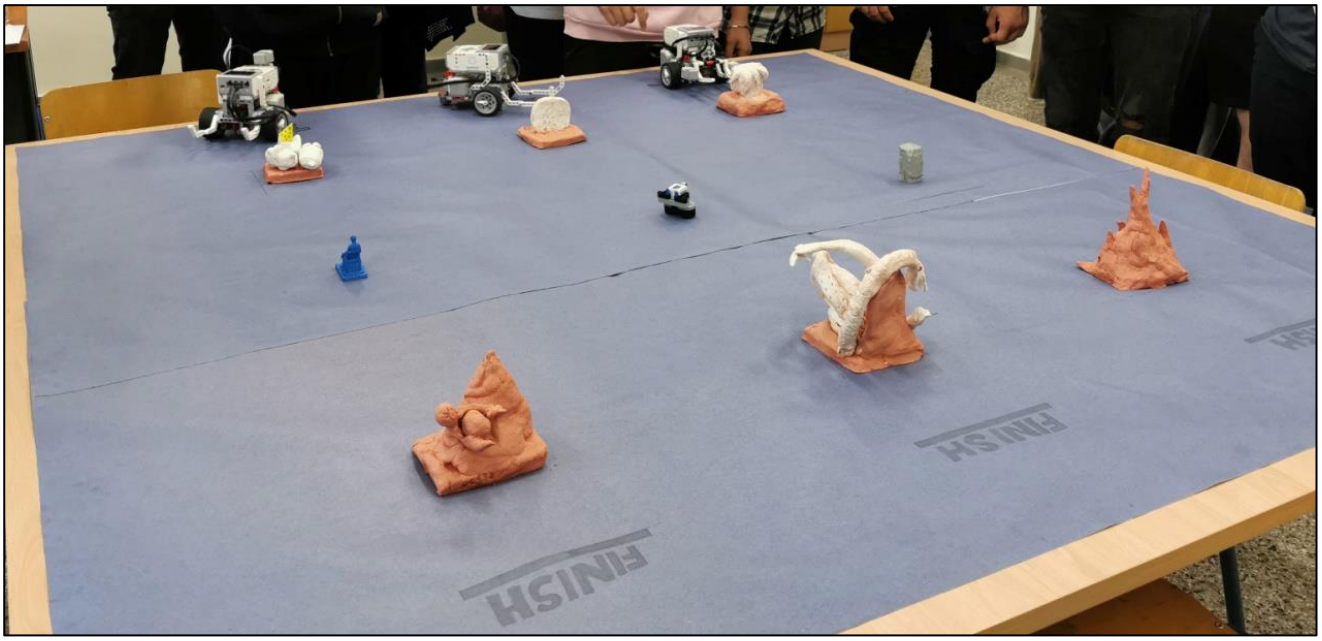
Task 10

Add the resources on the web platform (https://www.fablabschoolnet.eu/) and use the ARLectio app to see the AR content on the board.

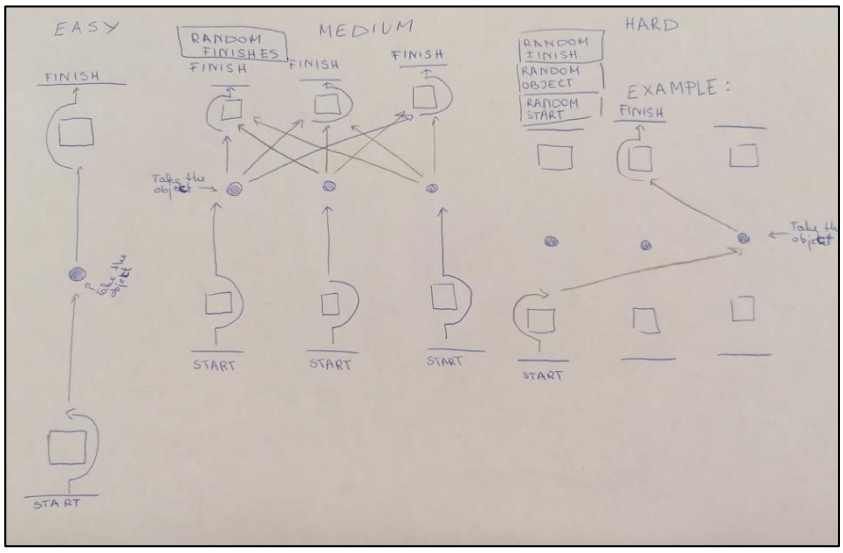
Sample board

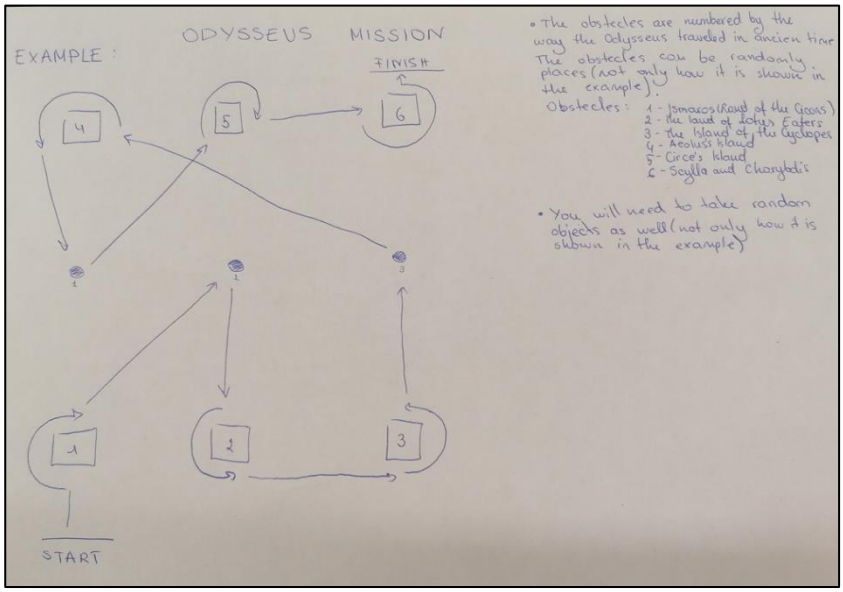




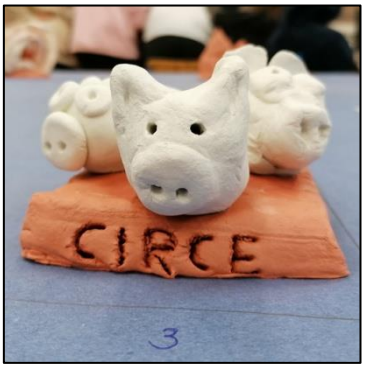


Sample routes

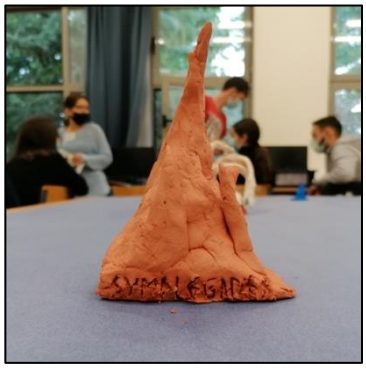




Sample artworks



Circe



Symplegades

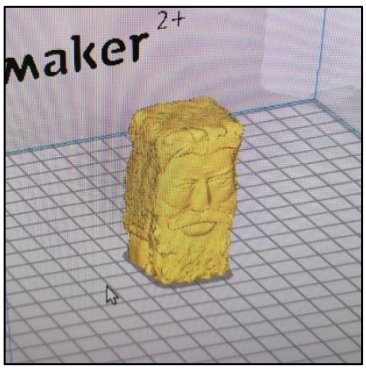
Scylla

Cyclop

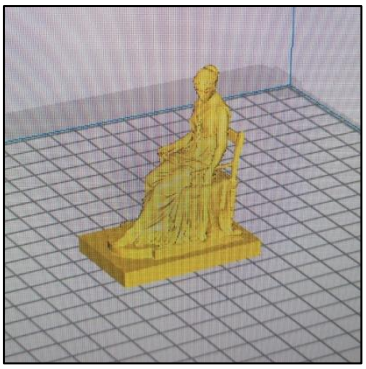
Laestrygonians

Aeolus

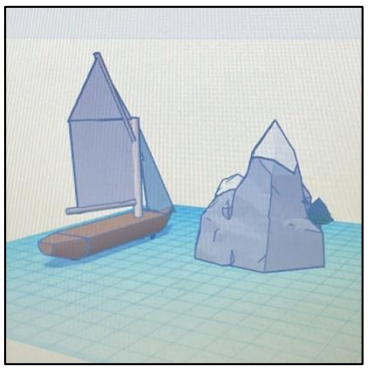
Sample 3D models and printings



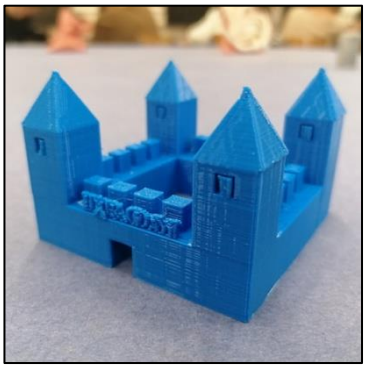
Odysseus



Penelope

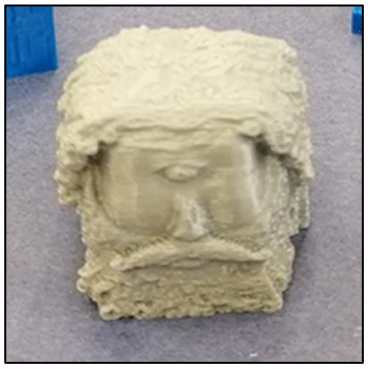


Odysseus' boat



Castle of Troy

Palace of Ithaca



Cyclop

Chapter 2. FABLAB SCHOOLNET IMPLEMENTATION IN LITHUANIA

Robotics. Long Term Plan

NAME, NAME, QUALIFICATION CATEGORY OF TEACHER WHO PREPARED THE PLAN:
……..

CLASS, COURSE: II-IV (optional)

NUMBER OF LESSONS DURING THE WEEK AND SCHOOL YEAR: 1 (35)

CLASS / GROUP CONTEXT (CHARACTERISTICS):

- **Classroom learning** (student achievement levels)
- **Self-directed learning** (students' ability to learn independently, to plan time, to choose learning methods, level of competence learning to learn)
- **Learning sociality** (students' ability to work in groups of different sizes, students' ability to help each other)
- **Classroom / group microclimate** (students' relationship, problem solving, adherence to agreements)

LEARNING OBJECTIVE: to grasp and understand the benefits of robotics in modern industries and services, to learn the basic programming commands and to control with them the robot.

TASKS:
1. To understand the principles of the robot construction, to be able to explain the use and benefits of the robot.
2. To learn the programing and controlling the robot.
3. To set a problem in the robot movement or control of the robot and perform new tasks.

INTEGRATION (underline the integrated program):

- ***Career education program;***

The gymnasium integrates relevant topics into the curriculum (underline the topic):

- ***Information literacy;***

EVALUATION (assessment system applied, cumulative points, total marks, ***monitoring of the student's individual progress***, recording, analysis)
;:
TEACHING TOOLS (textbooks, exercises, digital tutorials):

TRAINING CONTENT

PHASE (CYCLE), DIMENSIONS	**NUMBER OF HOURS**	**STUDENTS' SKILLS**	**EVALUATION** (diagnostic assessment: control work, oral assessment, laboratory work, test, etc., cumulative assessment: group work, test, public speaking, etc.)
1. Introduction. "Robotization in my world"	1	To be able to find information, organize and adapt topics for presentation with the help of search engines.	Individual work
2. Submission of information	1	Be able to convey orally prepared information on a given topic. Improve speaking skills.	Public speaking
3. Robot "body" design, construction, customization of measuring element and sensors a) Lego EV3 education kits (4 hours) b) mBot Ranger kits (4 hours) c) mBot Ultimate kits (4 hours)	12	Be able to read and understand construction instructions. Be able to adapt the required measuring elements and sensors, be able to connect them to the control unit.	Group work
4. Robots controlling and programming a) Lego EV3 education kits (4 hours) b) mBot Ranger kits (4 hours) c) mBot Ultimate kits (4 hours)	12	Be able to write the simplest program for robots controlling. Be able to understand and to know how to explain program commands to the others. To understand the operation control of engines.	Group work
5. Participation in various events (competitions, STEAM days, opened lessons, trainings)	4	Be able quickly and efficiently redesign a robot and adapt it for the other robot activities. Be able to modify the robot control program according to the needs of the task.	Individual work

6. Robot problem solutions	5	Be able to raise problems in robot control or movement. Be able to find solutions to a given situation.	
Total	35		

The resources for Robotics in Lithuanian language are available on:

- „Lego Mindstorms“ daugiau nei tik važiuojantis robotas - YouTube
- Robotikos konstruktoriai vaikams. Vaikų iki ir nuo 10 metų programavimo mokymas. - YouTube
- Virtuali edukacija "Pažintis su robotika ir programavimu“ - YouTube,
- Makeblock konstruktorius robotas mBot Ultimate 2.0 - YouTube

3D printing. Long Term Plan

20.. – 20.. school year

NAME, SURNAME, QUALIFICATION CATEGORY OF TEACHER WHO PREPARED THE PLAN:
……

CLASS, COURSE: II-IV (optional)

NUMBER OF LESSONS DURING THE WEEK AND SCHOOL YEAR: 1 (35)

CLASS / GROUP CONTEXT (CHARACTERISTICS):

- **Classroom learning** (student achievement levels)
- **Self-directed learning** (students' ability to learn independently, to plan time, to choose learning methods, level of competence learning to learn)
- **Learning sociality** (students' ability to work in groups of different sizes, students' ability to help each other)
- **Classroom / group microclimate** (students’ relationship, problem solving, adherence to agreements)

LEARNING OBJECTIVE: to grasp and understand the principles of creating, conveying and printing a 3D layout (object), to learn how to create the simplest 3D layouts by using various editing programs.

TASKS:
1. To understand the principles of printing a 3D object, to be able to explain how 3D layouts or objects appear.
2. To learn how to create the simplest objects, layouts by using various editing programs.
3. To learn how to use 3D printer.

INTEGRATION (underline the integrated program):

- ***Career education program;***

The gymnasium integrates relevant topics into the curriculum (underline the topic):

- ***Information literacy:***

EVALUATION (assessment system applied, cumulative points, total marks, ***monitoring of the student's individual progress,*** recording, analysis)

TEACHING TOOLS (textbooks, exercises, digital tutorials):

TRAINING CONTENT

PHASE (CYCLE), DIMENSIONS	**NUMBER OF HOURS**	**STUDENTS' SKILLS**	**EVALUATION** (diagnostic assessment: control work, oral assessment, laboratory work, test, etc., cumulative assessment: group work, test, public speaking, etc.)
1. Introduction. 3D in the palm of my hand	1	Be able to find a 3D object. Understand the variety of 3D object formats, be able to export and import 3D objects.	
2. Submission of information	1	Be able to convey orally prepared information on a given topic. Improve speaking skills.	Public speaking
3. Overview of 3D printers. Print settings, basic name and terms	3	Be able to tell the types of 3D printers. Be able to apply 3D object printer settings, fillings in.	Testing
4. Creating 3D objects of environments a) SugarCad (8 hours) b) Autodesk 123D design (8 hours) c) TinkerCad (8 hours)	24	Be able to create the simplest 3D object. Be able to work with various 3D development programs. Be able to create and present a 3D layout.	Group work
5. TinkerCad 3D programming	3	Be able to create a 3D object during programming.	Testing
6. Problematic solutions for replacing original parts or objects with 3D objects.	3	Be able to find solutions to a given situation.	Individual work
Total	35 hours		

Augmented Reality using Mobile Devices. Long Term Plan

20.. – 20.. school year

NAME, SURNAME, QUALIFICATION CATEGORY OF TEACHER WHO PREPARED THE PLAN:
……

CLASS, COURSE: II-IV (optional)

NUMBER OF LESSONS DURING THE WEEK AND SCHOOL YEAR: 1 (35)

CLASS / GROUP CONTEXT (CHARACTERISTICS):

- **Classroom learning** (student achievement levels)
- **Self-directed learning** (students' ability to learn independently, to plan time, to choose learning methods, level of competence learning to learn)
- **Learning sociality** (students' ability to work in groups of different sizes, students' ability to help each other)
- **Classroom / group microclimate** (students' relationship, problem solving, adherence to agreements)

LEARNING OBJECTIVE: to grasp and understand the principles of creating, transmitting and using AR, to learn how to create the simplest objects conveyed by AR technology, to learn to use different applications.

TASKS:
1. To understand the principles of displaying an AR object, to be able to explain how to convey visual information with the help of an AR object.
2. Learn to use various AR apps.
3. Development of AR on various platforms.

INTEGRATION (underline the integrated program):

- ***Career education program;***

The gymnasium integrates relevant topics into the curriculum (underline the topic):

- ***Information literacy;***

EVALUATION (assessment system applied, cumulative points, total marks, ***monitoring of the student's individual progress,*** recording, analysis)

TEACHING TOOLS (textbooks, exercises, digital tutorials):

TRAINING CONTENT

PHASE (CYCLE), DIMENSIONS	**NUMBER OF HOURS**	**STUDENTS' SKILLS**	**EVALUATION** (diagnostic assessment: control work, oral assessment, laboratory work, test, etc., cumulative assessment: group work, test, public speaking, etc.)
1. Introduction. VR or AR	1	Be able to understand the differences between VR and AR.	
2. Submission of information	1	Be able to convey orally prepared information on a given topic. Improve speaking skills.	Public speaking
3. AR technology. Requirements for using AR technilogies. AppStore and GooglePlay Gadget Review. Demos for using selected applications	2 p.m.	Be able to demonstrate AR applications with both video and video reviews. Be able to create video reviews.	Group project
5. QR code	1 p.m.	Be able to generate a QR code. To be able to use QR codes in different devices.	Individual work
4. AR objects on the phone, tablet. a) education and learning (10 hours) b) entertainment and living (4 hours) c) Information and communication areas (8 hours)	10 p.m.	Be able to use AR apps for various educational or learning purposes. Be able to use AR apps for entertainment or alternative purposes. Be able to create an AR information stand with the help of QR, photo, audio or video.	Individual work
6. Search and creation of school information places.	8 p.m.	Be able to create an information stand in the relay life of the school.	Group project
Total	35 hours		

Chapter 3. FABLAB SCHOOLNET IMPLEMENTATION IN BULGARIA

3D printing, Robotics and mobile technologies

Google Expeditions in action

Topic title
A journey through technological evolution
For the lesson
Class: 9c Subject Title: Information Technology Number of students: 12 Objectives: - Students to get acquainted with the practical / real part of the things we learn. - To provoke interest / curiosity. - To enrich their knowledge about supercomputers and the evolution of computers. - To acquire independent skills in research direction. Expected results: - Acquisition of new knowledge and skills in working with mobile phones. - Acquiring new knowledge on the topics of computer evolution. Guiding questions: Who created the first computer? We have studied two types of operating systems. What are they? Did you look at the lyrics on the scenes? The descriptive text in the scene? Glossary of new concepts / old forgotten concepts: - Cryptography - Main team - RAM and ROM
Preparation for the expedition
Choose an expedition: Evolution of Computers Choose an interesting point of the site: Interesting points of the expedition are: - Get acquainted with the information in the scenes: The information in the scenes is suitable for the purposes of the lesson. Due to the fact that we have only 40 minutes, 4 scenes have been selected to be considered. They have the most important points for the material and general information as well suitable for quiz type questions: Unrealized Ideas, 20th Century Revolution, True Modern Marvel: The Smartphones, The Evolution of Play.
Before the expedition
Activating students *(Step on the knowledge they already have to prepare them for the expedition):* You all know who created the first computer, right? You have almost witnessed the evolution of mobile phones, as well as some of the most famous games of this decade. This hour, playing a game divided into two teams, we will learn more interesting things on the topic, which is the evolution of computers. I'm checking how many people have downloaded the app - in this case they were all without the students,

who do not have Android.

I divide the students into teams of 6 students with 4 phones.

During the expedition

Answering questions is done by raising your hand. The first team to raise their hand and answer correctly receives a point. The winning team does not control the share
next week.

After the students look at Scene 1 - Unrealized Ideas:

1. Who created the Difference Engine? - Instant response.
2. In what directions was it used? - Additional stage research, attempts at learning.

Students view Scene 3 - 20th Century Revolution:

3. For what purpose was Alan Turing's machine used?
4. What do modern computers and the Apple Lisa computer have in common? 5. What was the name of the machine used during World War II? *Bonus point:* Do you know anything else with this name?

Students view Scene 5 - True Modern Marvel: The Smartphones:

6. What power does ENIAC require and what does iPhone 6 require?
7. What is the CPU speed of the iPhone 6?
8. What is the random access memory of Apollo Guidance Computer?

Students view Scene 6 - The Evolution of Play:

9. What is the name of the first publicly demonstrated video game?
10. What was the first game to become Mainsteam (discussion of the word mainstream)? Say her name in Bulgarian and English.

Bonus point: What is the name of the game that is on the point of The State of Modern Gaming?

After the expedition

The points of both teams should be 10-12. In the case of the class in question, there are 11. They managed to answer the last bonus question, but not the first. All the main issues were dealt with,
as some took more time.

The winning team received its prize: no control next week. The others, of course, were not happy.

I asked the students if they liked this type of class. Most of them liked it. Several students did not answer. For homework I gave them to study all the scenes from this expedition and
the next hour to share them with the class before the control.

Upgrading ideas

It took a while for everyone to open the right expedition. The lesson was very useful for practice. In the last 2 scenes there was more interest and awareness than the boys on the topic. Maybe next time, gender should be taken into account when dividing groups. Maybe the number of questions should be reduced - because part of the lesson continued in the break.

Method 2: Autodesk Fusion 360

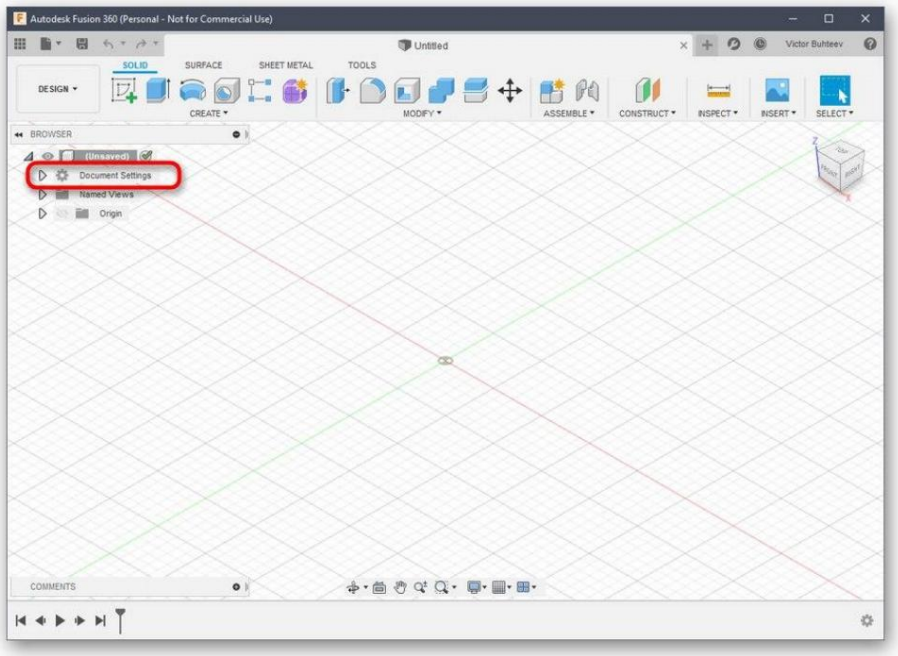
The next program, called Autodesk Fusion 360, is available for free private use for one year, so it's perfect for mastering and creating simple patterns to print on existing equipment in the future. We decided to make the principle of getting acquainted with this software the same as with Blender, so we created a phased separation.

Download Autodesk Fusion 360 from the official website

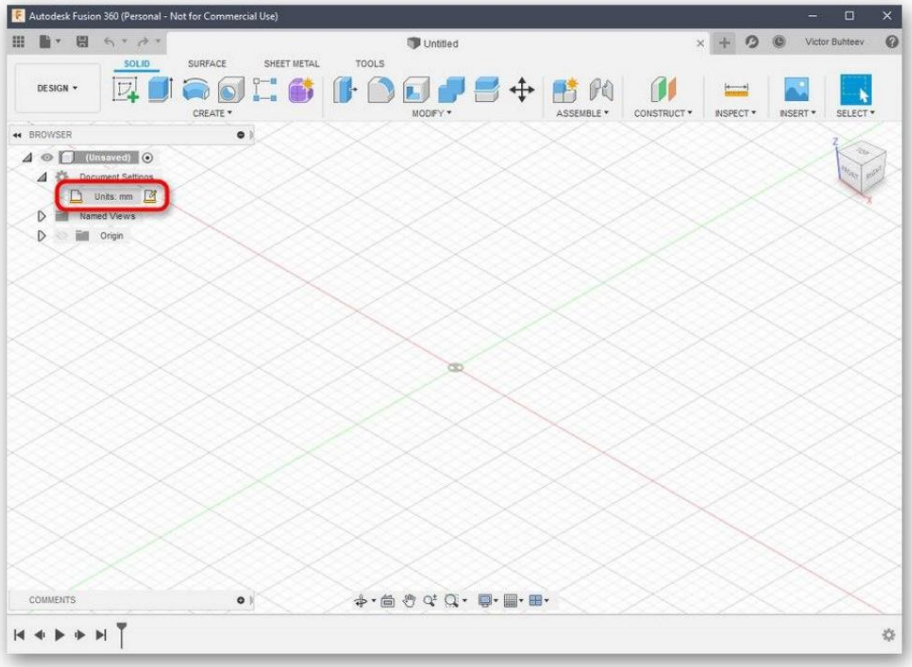
Step 1: preparatory steps

Autodesk Fusion 360 does not have to manually activate toolbars or select any unusual options. The user only needs to make sure that the project metrics are correct and, if necessary, change the properties of the sides of the view, which happens as follows:

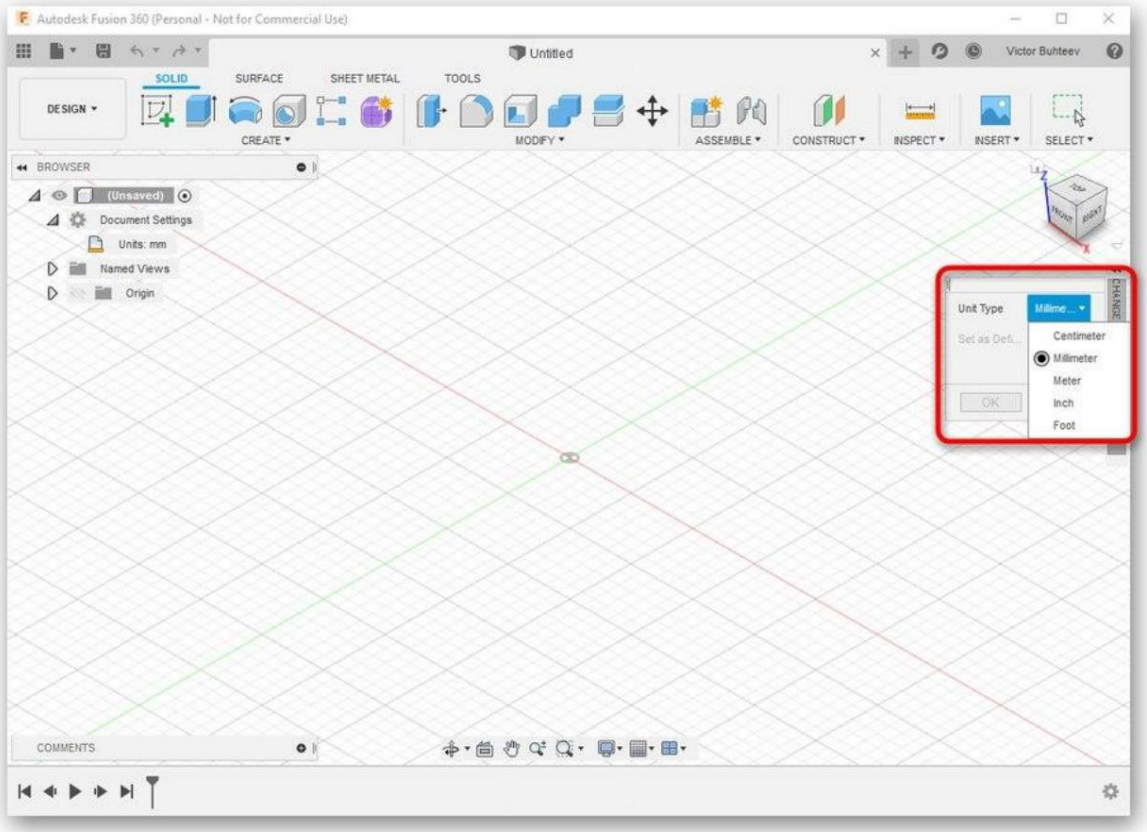
1. After downloading and installing Autodesk Fusion 360 from the official site, the first startup must occur. Start windows will not be displayed, so a new project will be created automatically. Pay attention to the *"Browser" section,* which is located on the left under the main panels. Select *"Document Settings"* here to expand this section.



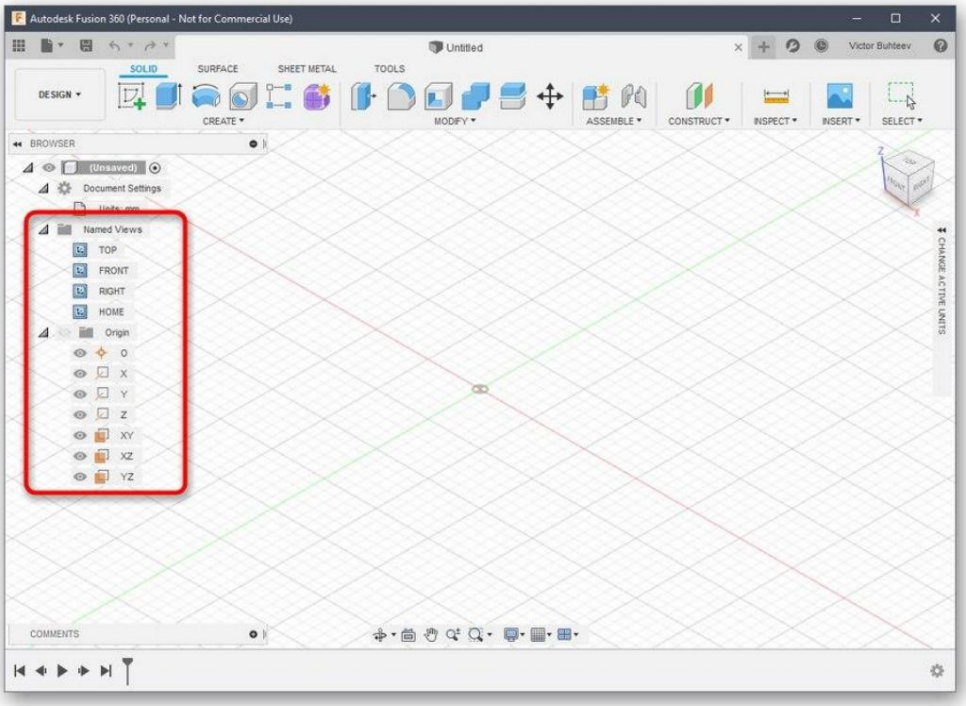
2. Go to Edit *Units files if* you are not happy with the value of millimeters by default.



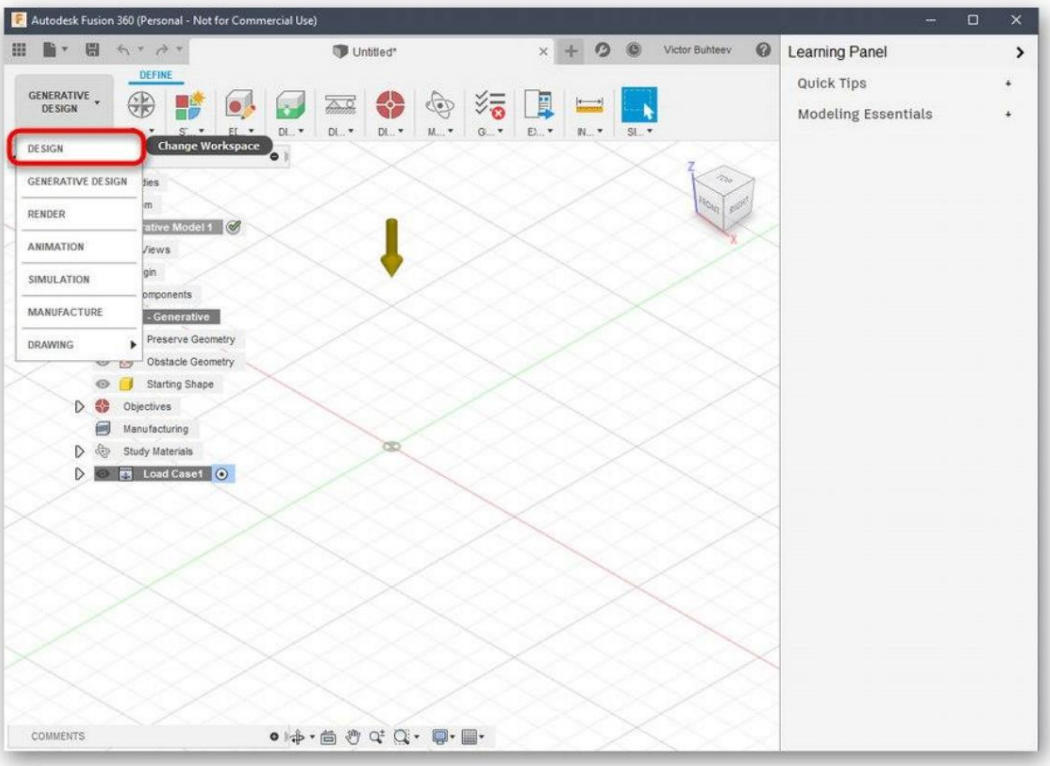
3. In the block shown on the right, select the optimal unit of measure you will need follow the whole time of interaction with the project.



4. Then read the section *Named Views* and *"Origin"* ... Here you can rename each country according to personal preferences and customize the display of axes in the workspace.



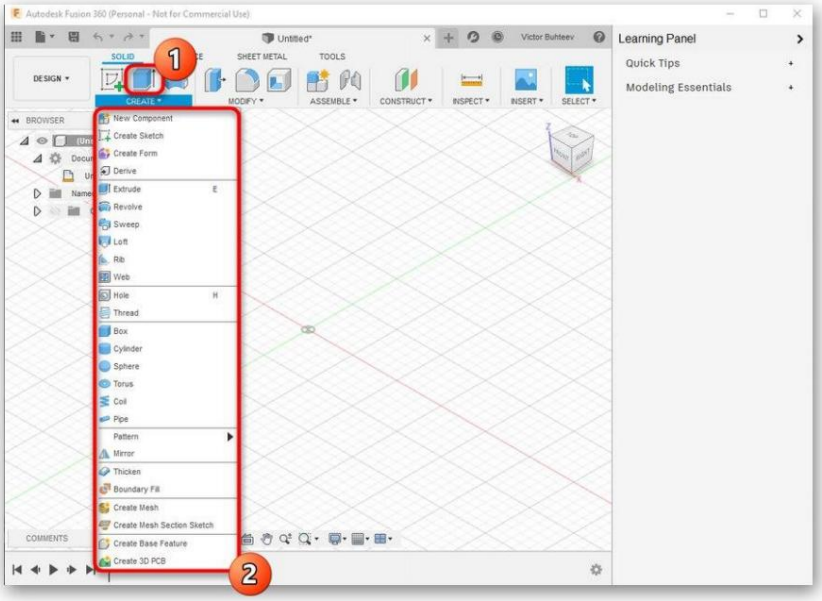
5. When completing the configuration, be sure to select *the "Design"* space , because that is where the primary creation of all objects takes place.



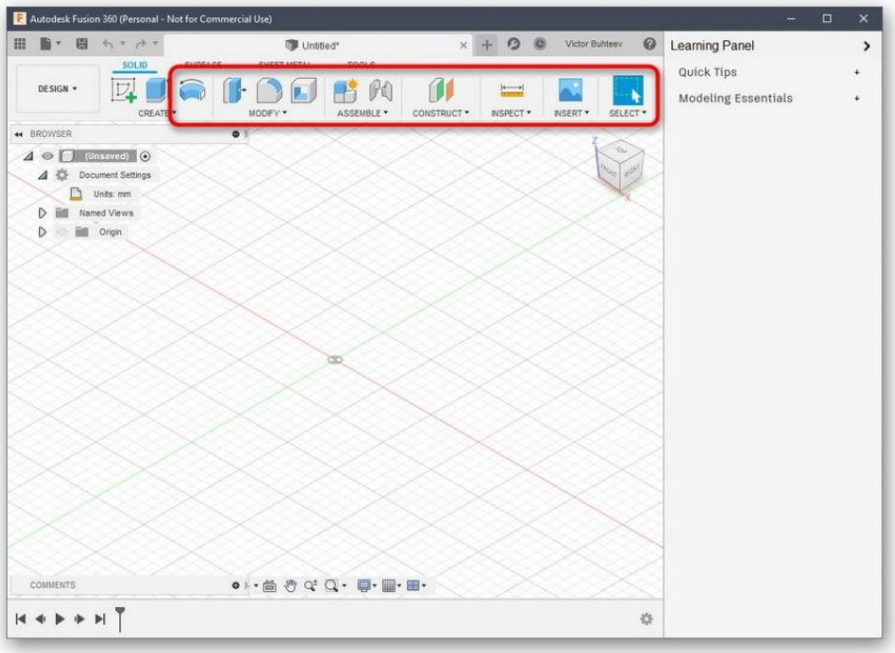
Step 2: Develop a print model

If you are faced with the need to manually design a model using Autodesk Fusion 360, you will have to study this program for a long time or at least get acquainted with the basics. Let's first look at a simple example of adding shapes and editing their size.

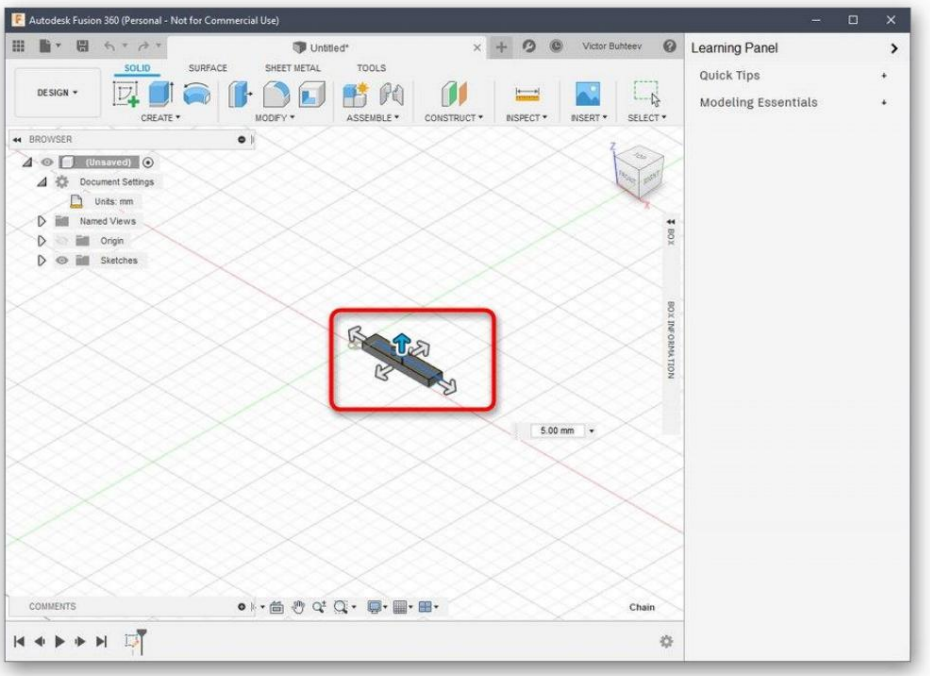
1. Open the *Create* list and check the available shapes and objects. As you can see all the basic forms are present here. Just click on one of them to go add.



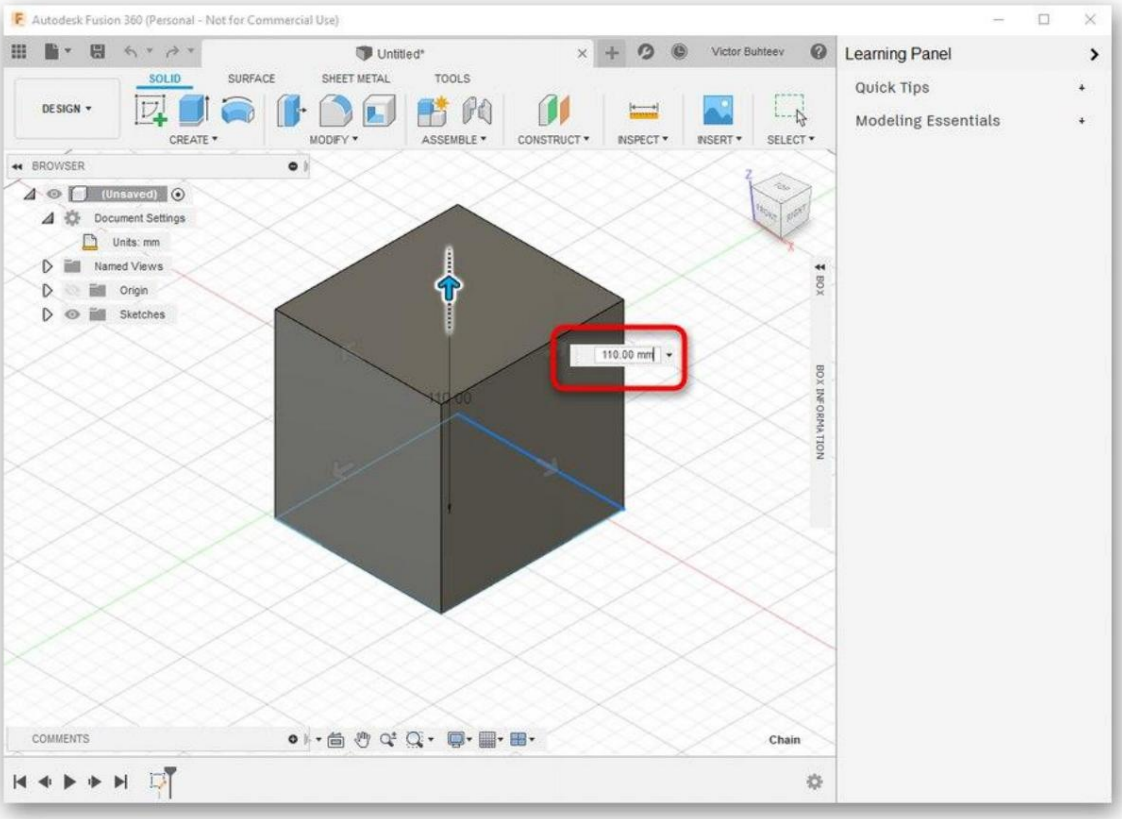
2. Also look at the other items in the top bar. The main thing space here is occupied by modifiers. The design of their icons makes it clear what they are responsible for. For example, the first modifier extrudes the sides, the second rounds them, and the third creates an indent.



3. After adding object shapes to the workspace, levers will appear, by moving which you adjust the size of each side.



4. When adjusting, look at the individual sizing field. You can do it edit yourself by setting the required values.



We've already talked about the key features to follow when reviewing Blender, so we won't dwell on that again. Instead, we suggest you do

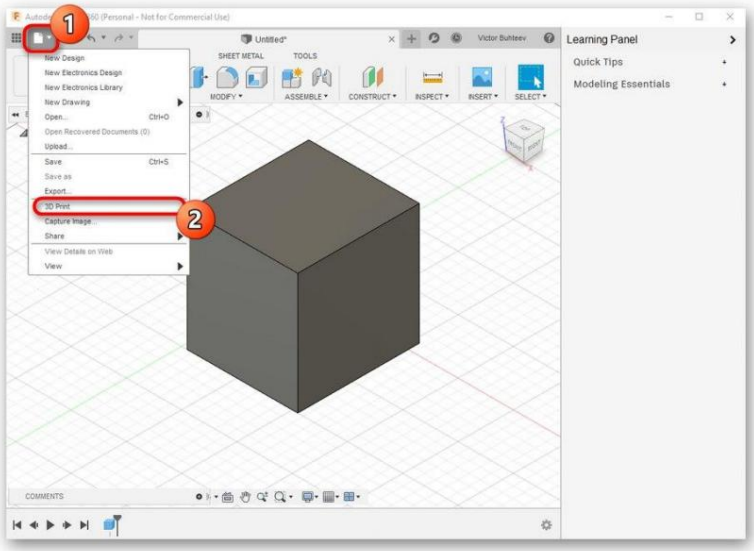
explore the rest of the interaction with Autodesk Fusion 360 by reading the official documentation on the site to master the creation of not only primitives, but also objects with a much higher level of complexity.

Continue reading the Autodesk Fusion 360 documentation

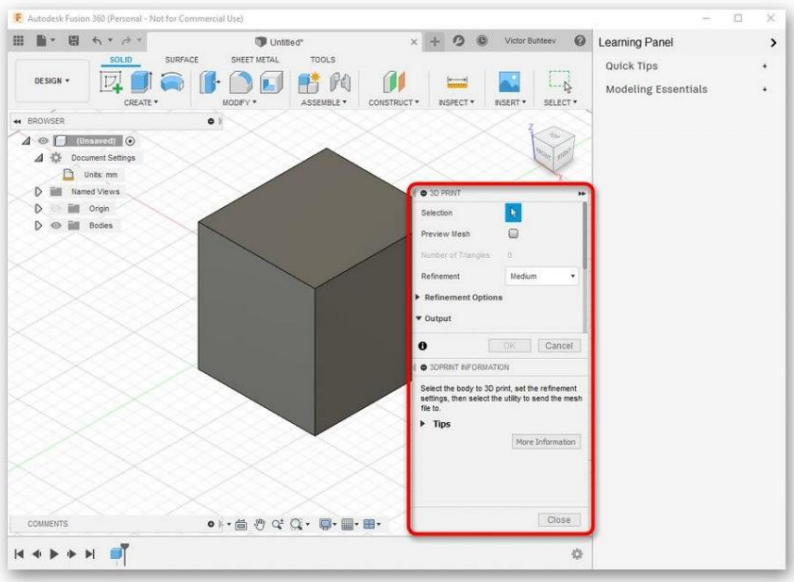
Step 3: Prepare to print / save the document

As part of this stage, we will talk about two different actions that are directly related to 3D printing. The first is to send the work immediately using the software you are using. This option is only suitable in situations where the printer itself can be connected to a computer and communicate with such software.

1. In *the "File"* menu, activate the *"3D Print" item.*

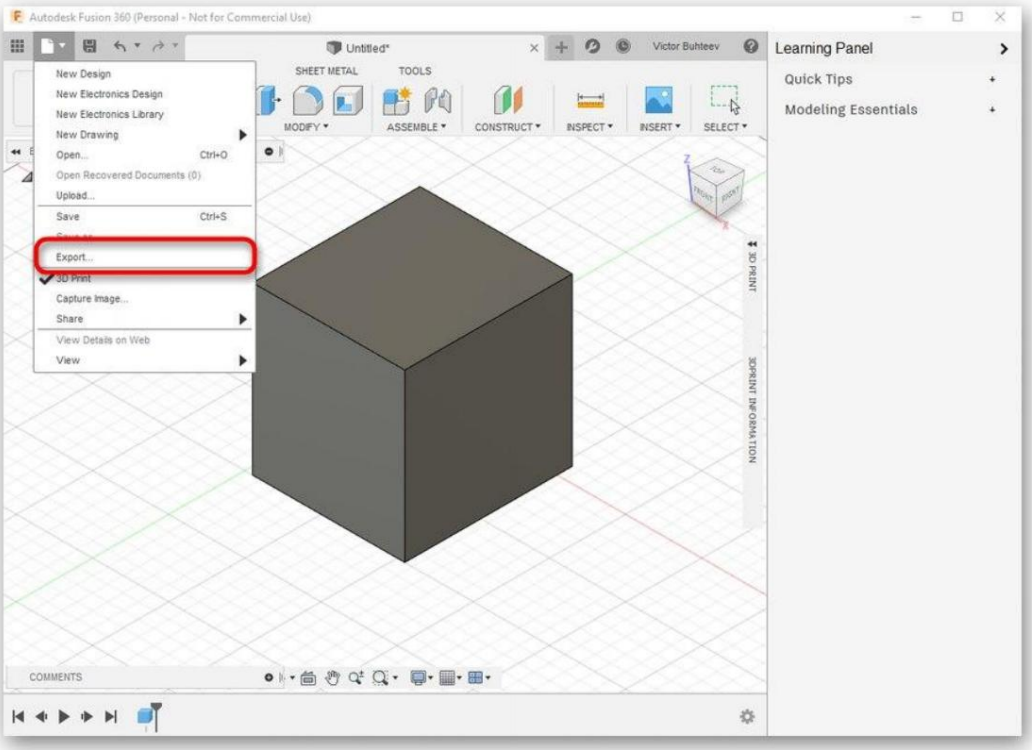


2. A settings block will appear on the right. Here you just need to select the output device itself, if necessary - to activate the visualization and start the task.

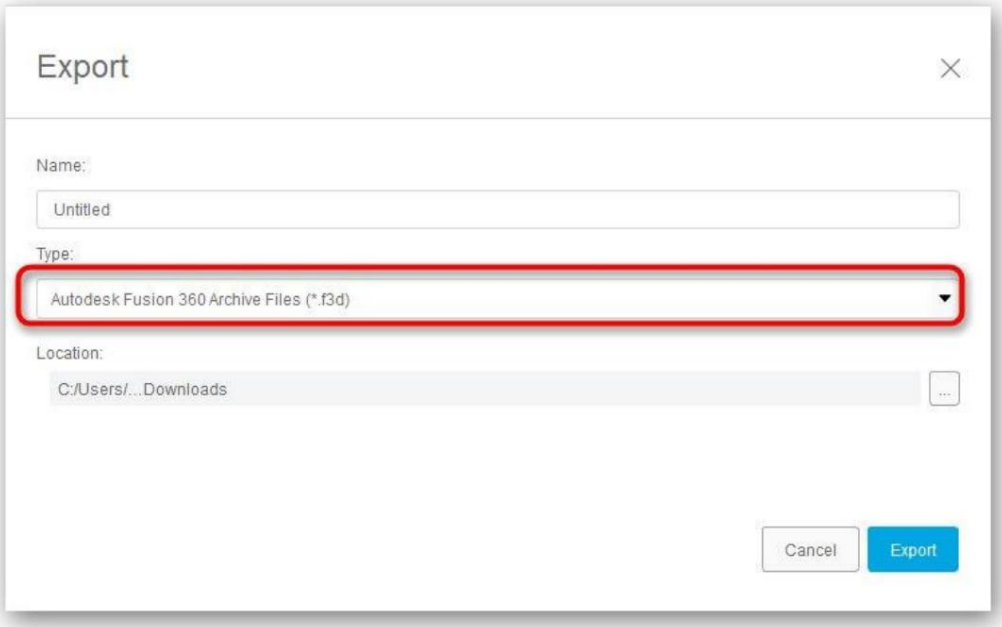


However, now most standard printers still only support USB memory or work exclusively with patented software, so the need to save an object arises much more often. This is done as follows:

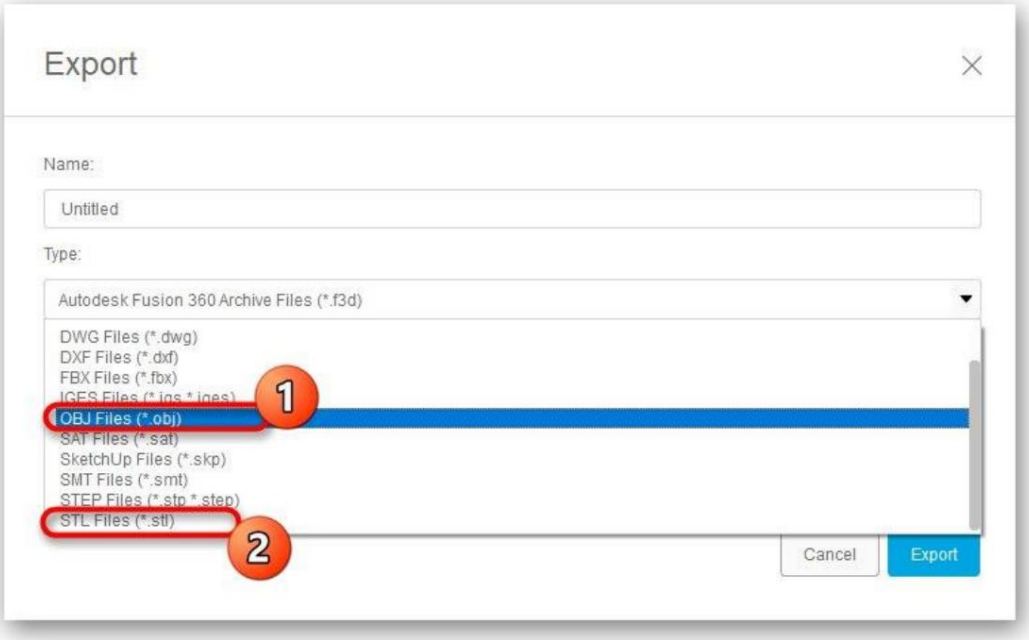
1. In the same *File* pop-up menu, click the *Export button.*



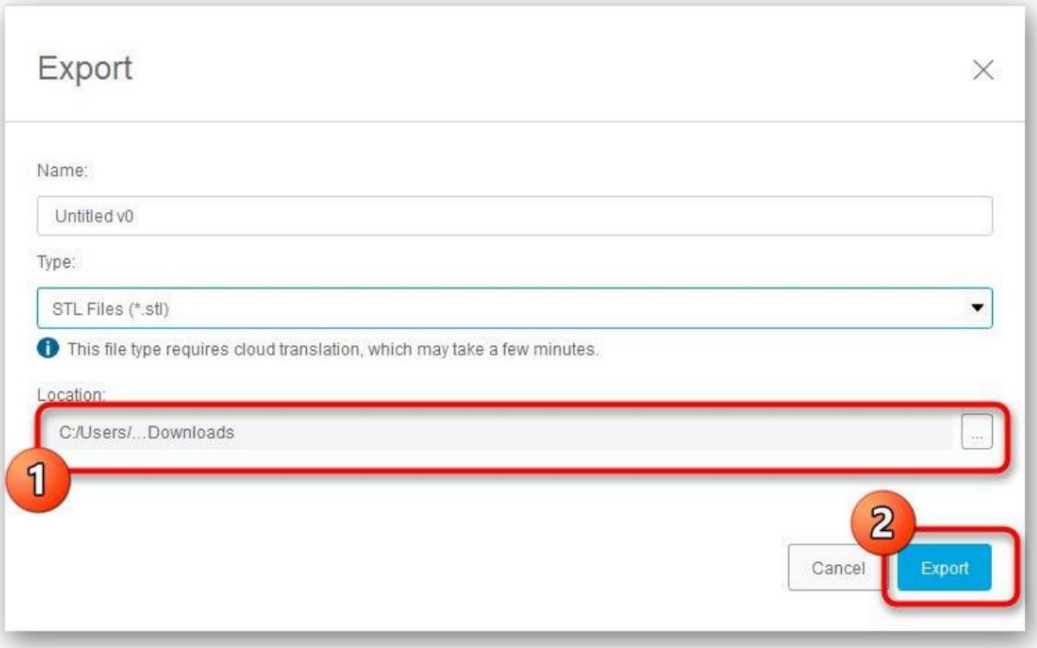
2. Expand the *Type list.*



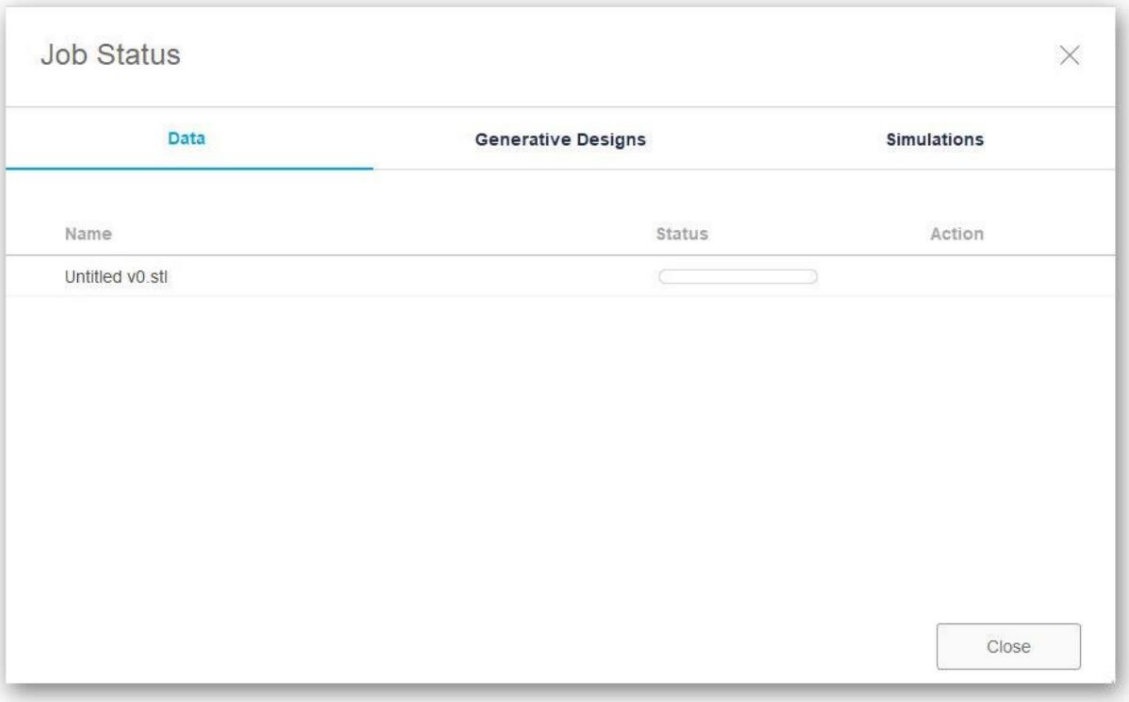
3. Select the subject *"OBJ files (* obj)"* or *"STL files (* .stl)"*.



4. Then set your save location and click the blue *Export button.*



5. Wait for the recording to finish. This process will only take a few minutes.



If this export fails, you will need to save the project again. To do this, click on the special button or use the standard key combination *Ctrl* + S.

Method 3: SketchUp

Many users know SketchUp as a home modeling tool, but the functionality of this software is much broader, so it can be used as a modeling tool in preparation for 3D printing. SketchUp is on our list today due to the easy import of ready-made free models for editing and additional saving in the desired format. Let's take turns to look at all aspects of managing this software.

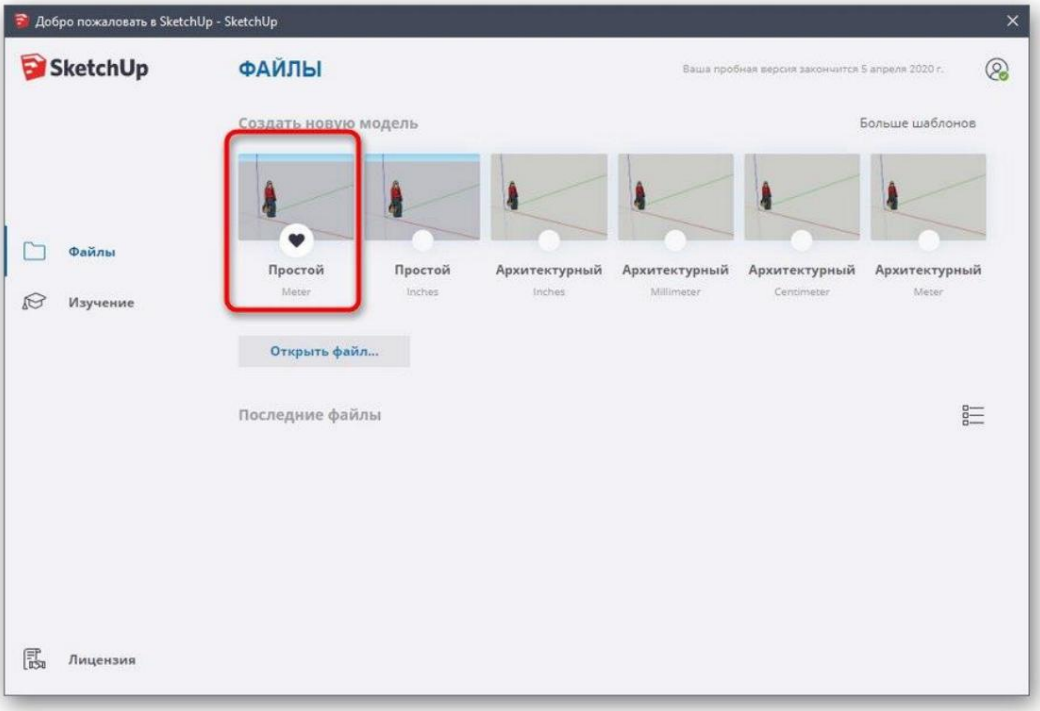
Download SketchUp

Step 1: First start and work with models

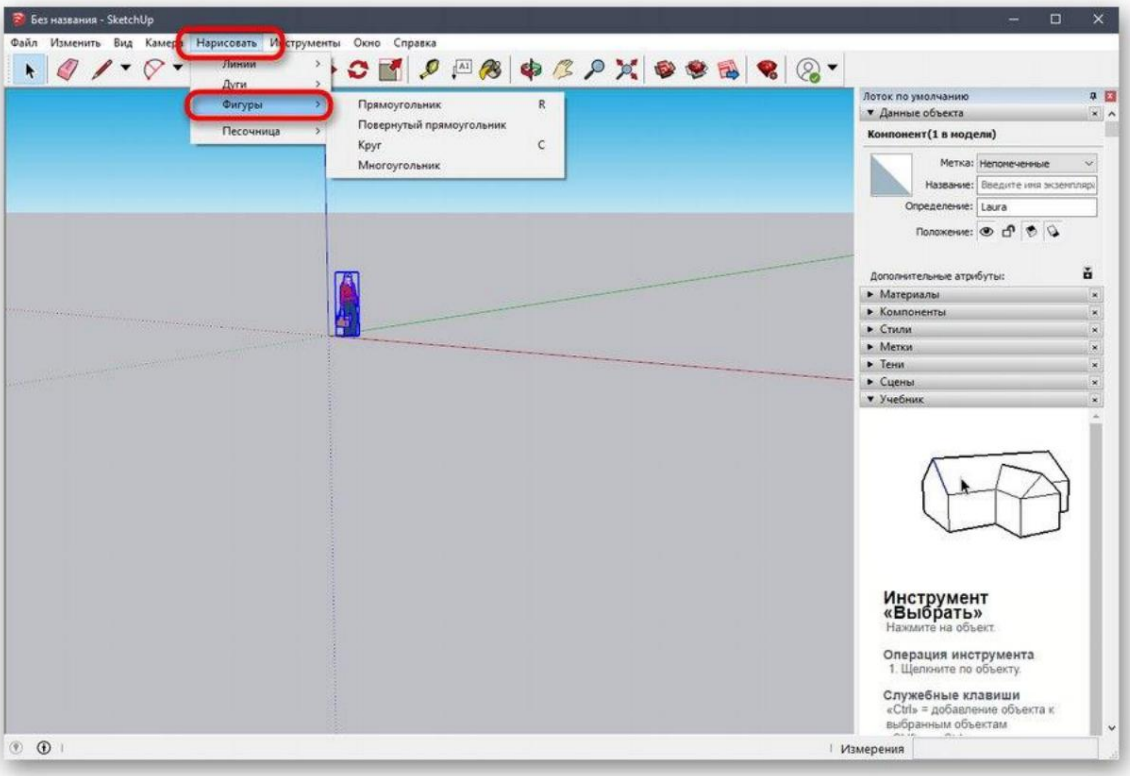
First we offer you to get acquainted with the basic principle of interaction with SketchUp to find out exactly how to add and manage models. We will also leave a link to the lessons if you want to explore this solution in more detail.

1. After installing and running SketchUp, you need to click the " Login" button to link a user account. If you have started to get acquainted with the trial period, then from this moment the countdown to the end of it begins.

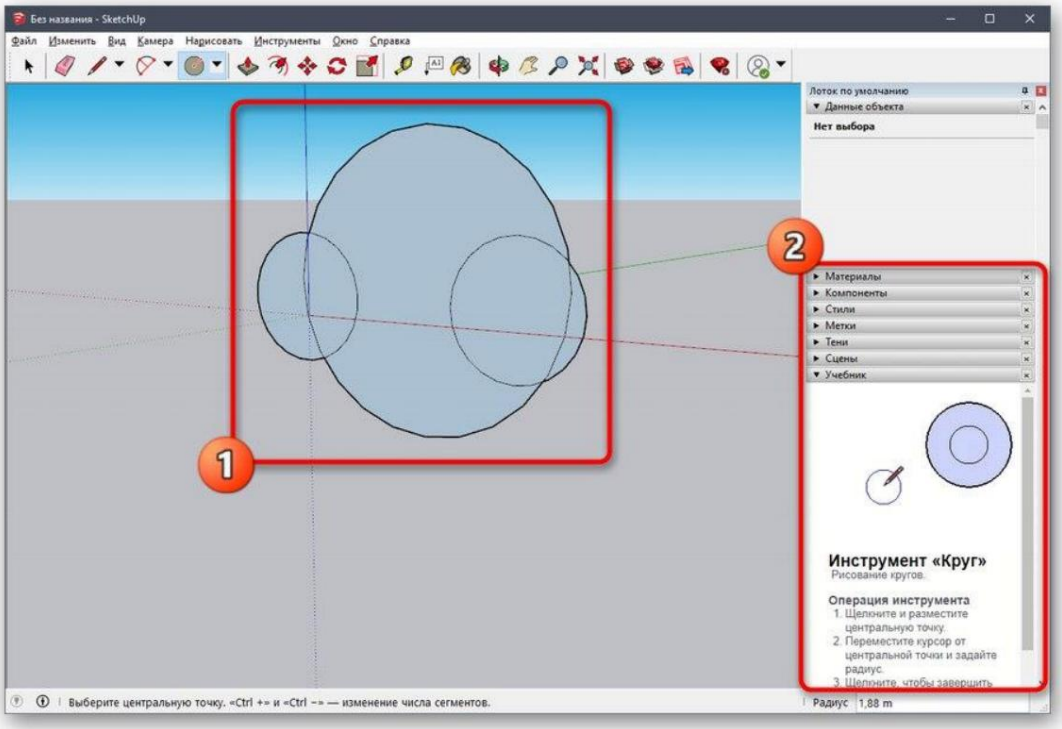
2. When the *SketchUp Welcome* window appears , click *Plain* to go to the workspace.



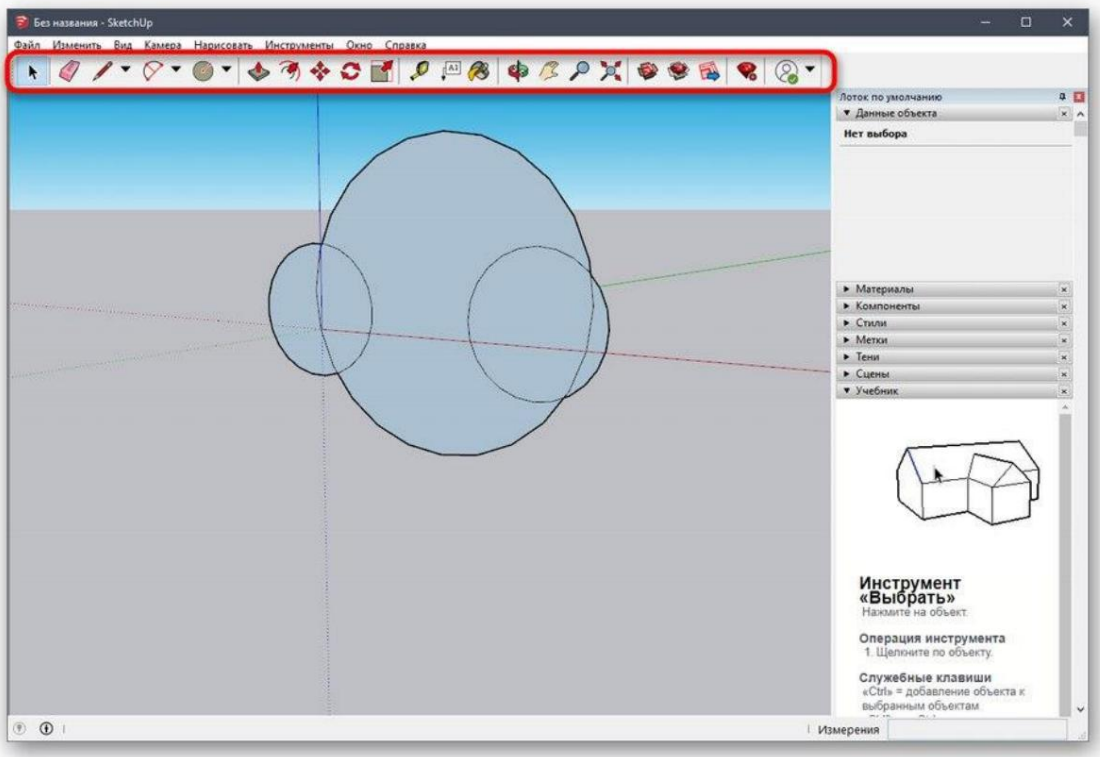
3. Drawing figures in this program is done in the same way as in others similar solutions. Hover over the *"Draw"* tab and select a custom shape.



4. It is then placed on the workspace and edited at the same time.



5. The other buttons on the top panels perform modification options and are responsible for other actions.



As we said earlier, the developers of SketchUp provide a wide variety of tutorials on how to interact with this application, not only in text format, but also

as a YouTube video. You can find out all about it on the official website using the link below.

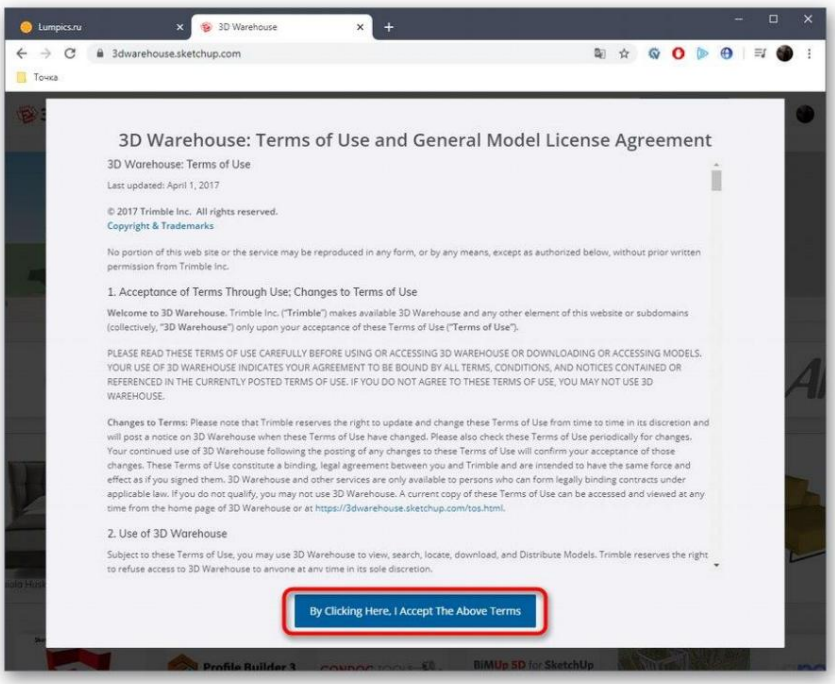
Continue reading the SketchUp documentation

Step 2: Load the finished model

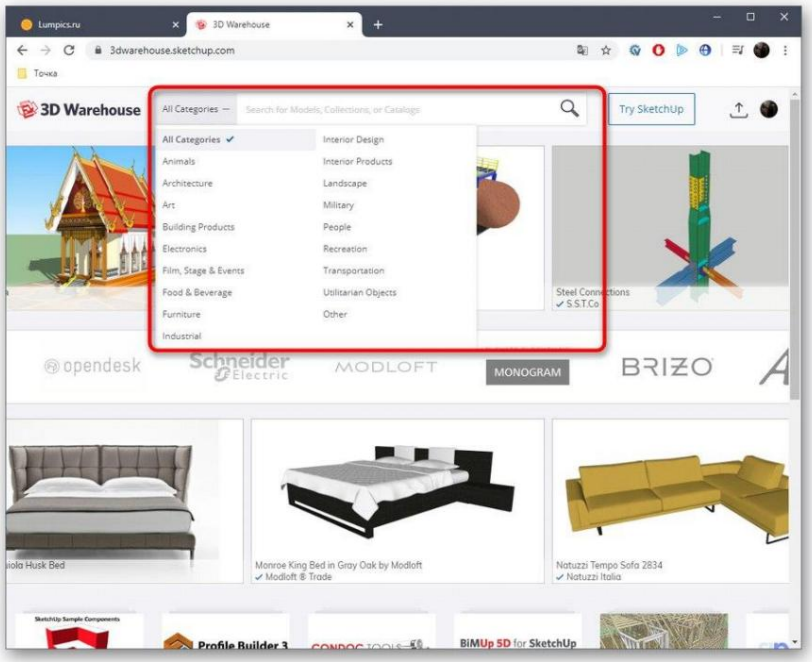
Not all users want to create their own models that will be sent for printing in the future. In such cases, you can upload a completed project, edit it, and only then export it in an appropriate format. For this, the official resource from the developers of SketchUp is used.

Go download models for SketchUp

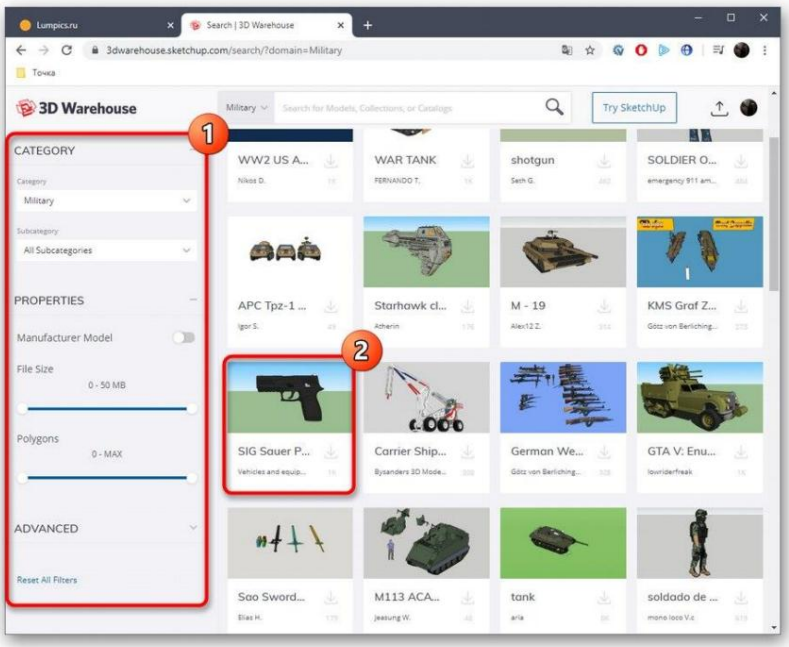
1. Use the link above to go to the main page of the site to find models. There, confirm the license agreement to get started use.



2. Then we suggest you use the built-in category search function to quickly find a suitable model.



3. Look for an option in the list, and also pay attention to additional ones filters.



4. Once you have selected a model, all you have to do is click on *"Download".*

5. Run the resulting file via SketchUp.

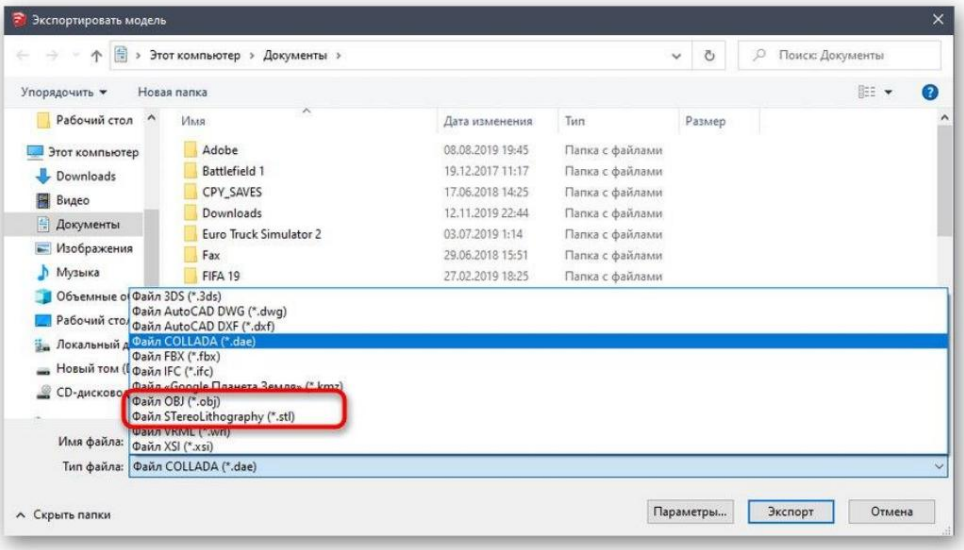
6. Review the model and edit it if necessary.

Step 3: Export the finished project

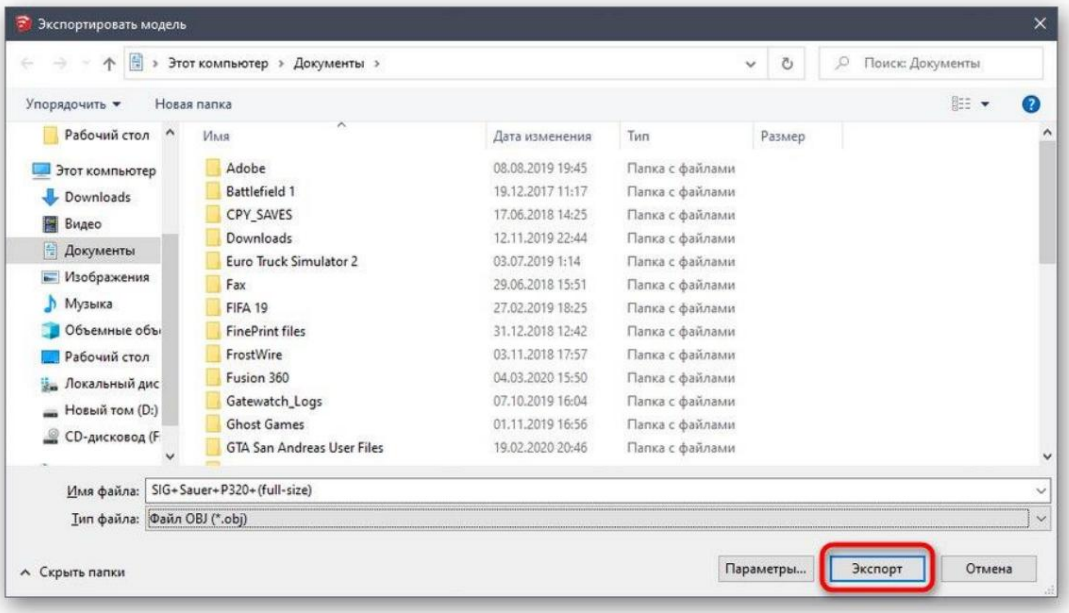
In the end, all you have to do is export the finished project for further printing on the existing device. You already know what format you should save the file to, but this is done as follows:

1. Hover over the *"File" - "Export" tab* and select *"3D Model".*

2. In the Explorer window that appears, you are interested in the OBJ or STL format.

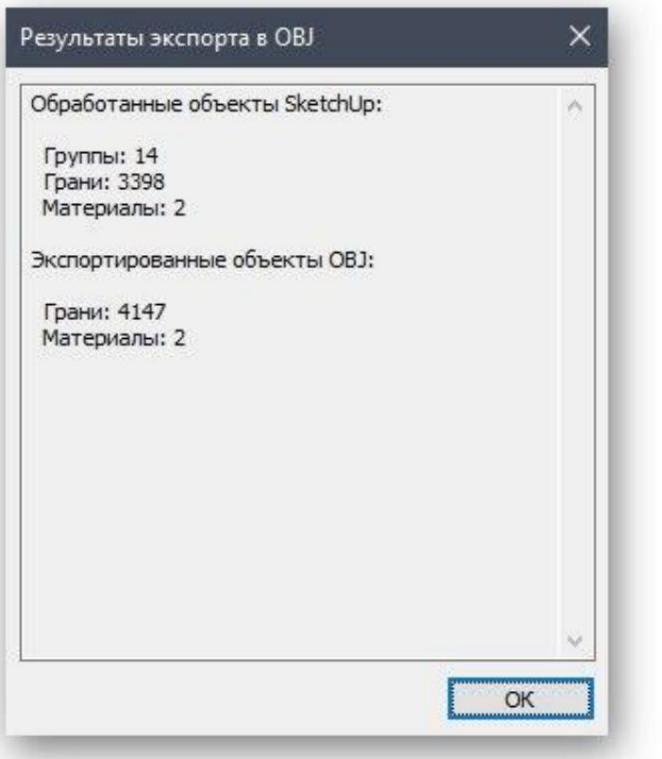


3. Once you've selected your location and format, all you have to do is click *Export.*



4. The export operation starts and you can monitor the status yourself.

5. You will receive information about the results of the procedure and you can proceed to the print job.



You have just learned about three different 3D modeling programs that are suitable for creating any job for printing on a 3D printer. There are other similar solutions that allow you to save files in STL or OBJ format. We recommend that you familiarize yourself with their list in those situations where the solutions described above do not suit you for some reason.

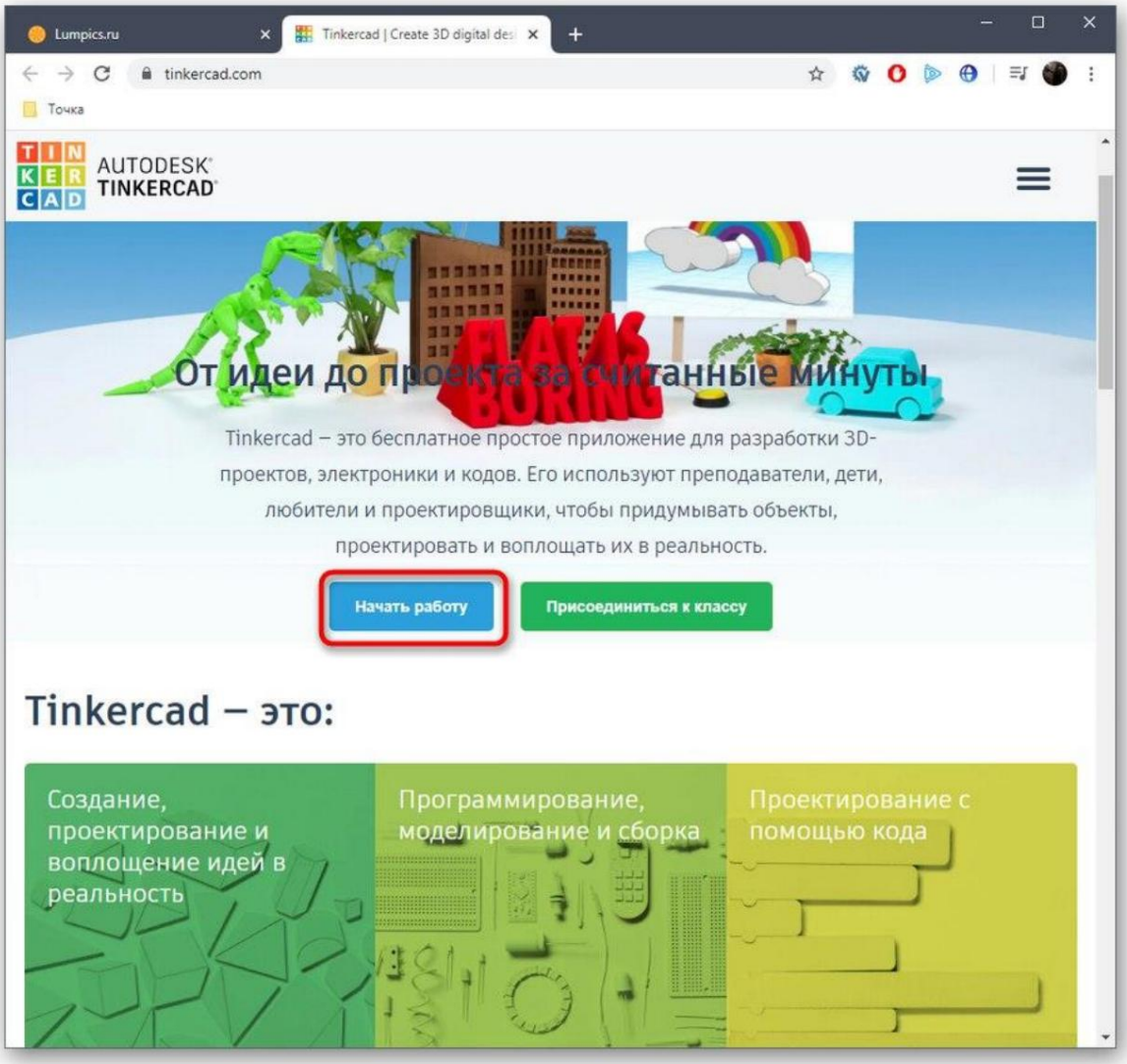
More information: 3D modeling software

Method 4: Online services

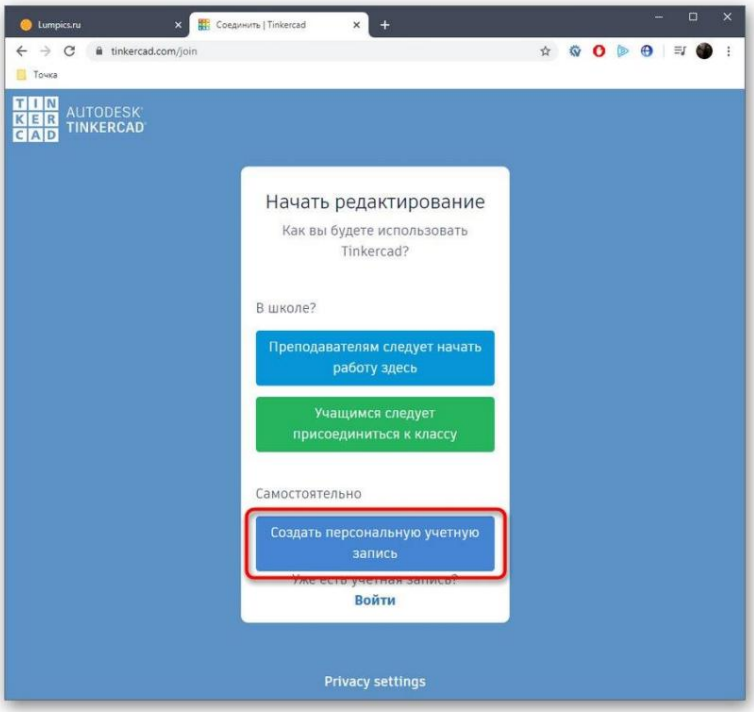
You can't ignore specialized online sites that allow you to create a 3D model without downloading an application to your computer, saving it in the desired format, or sending it to print immediately. The functionality of such web services is significantly inferior to full-fledged software, so they are suitable only for novice users. Let's look at an example of how such a site works.

Go to the Tinkercad website.

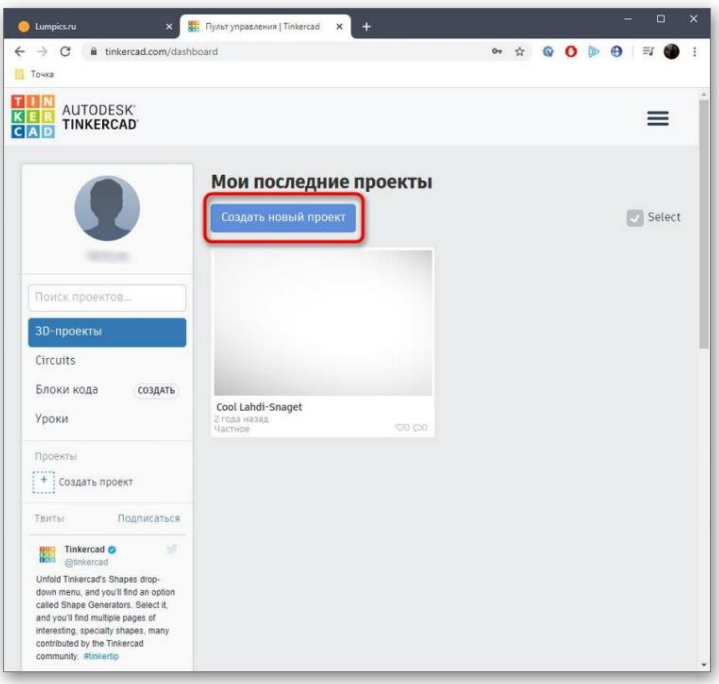
1. We have chosen Tinkercad as an example. Click on the link above to go to the site where you click the *Getting Started button*.



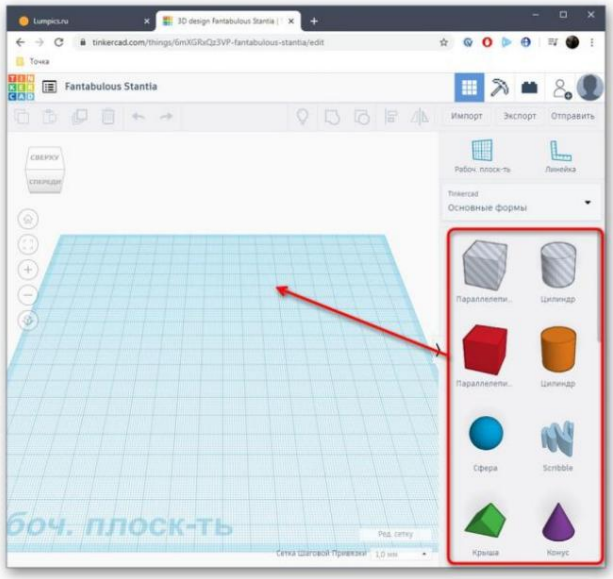
2. If you do not have an Autodesk account, you will need to create one to access your personal account.



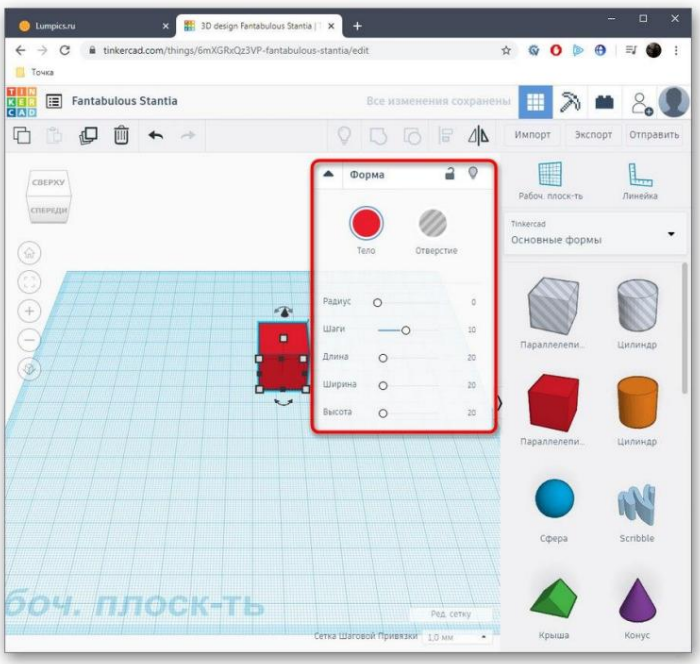
3. Then start creating a new project.



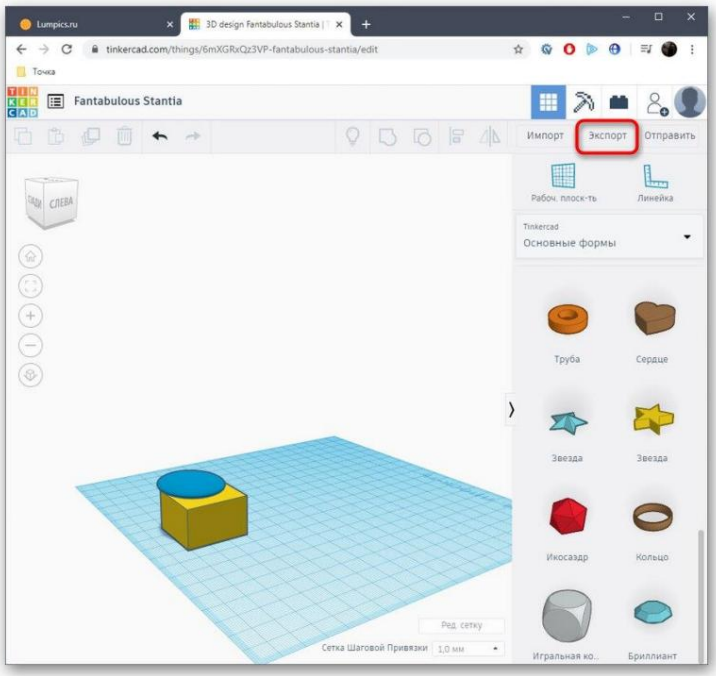
4. On the right side of the workspace you see the available shapes and forms. They are added to the plane by dragging



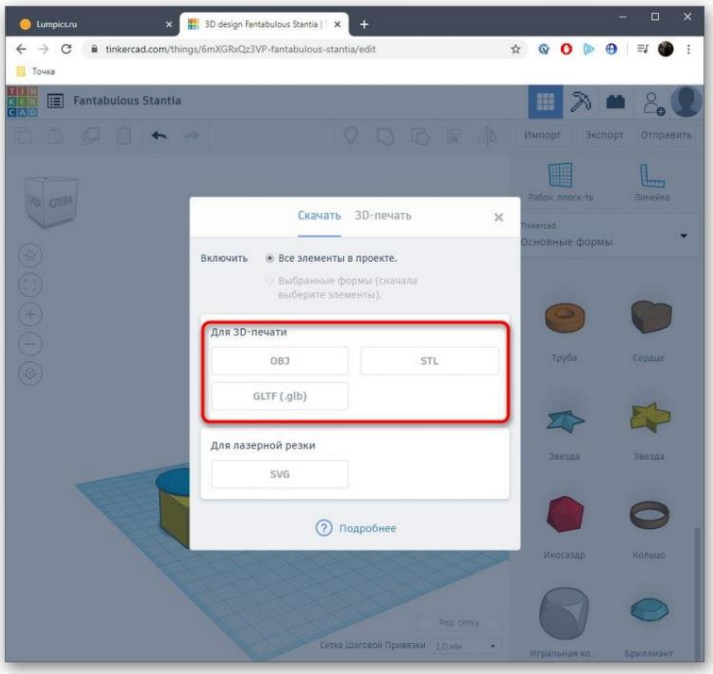
5. Then the size of the body and the hole is edited according to the requirements of the user



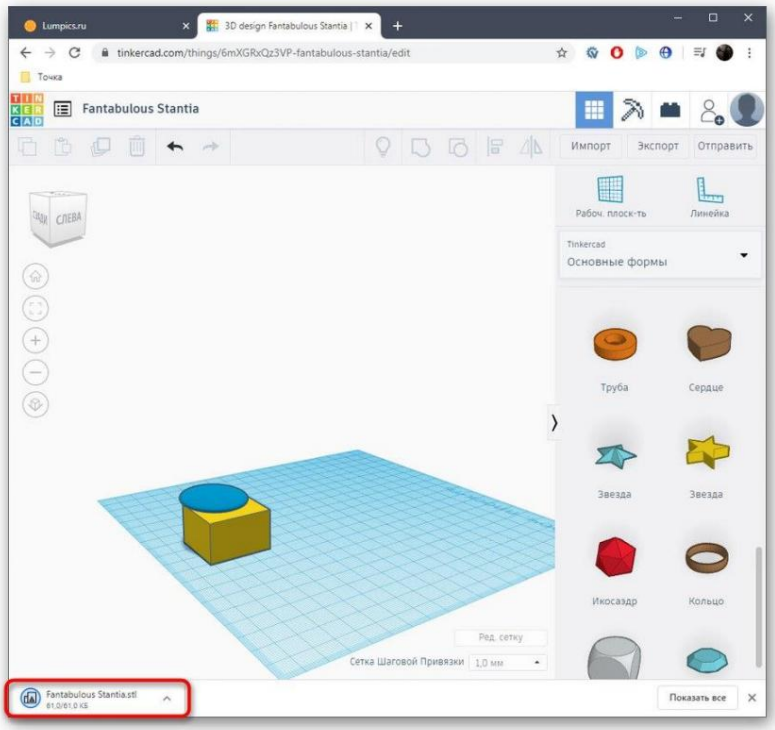
6. When you have finished the project, click *"Export"*.



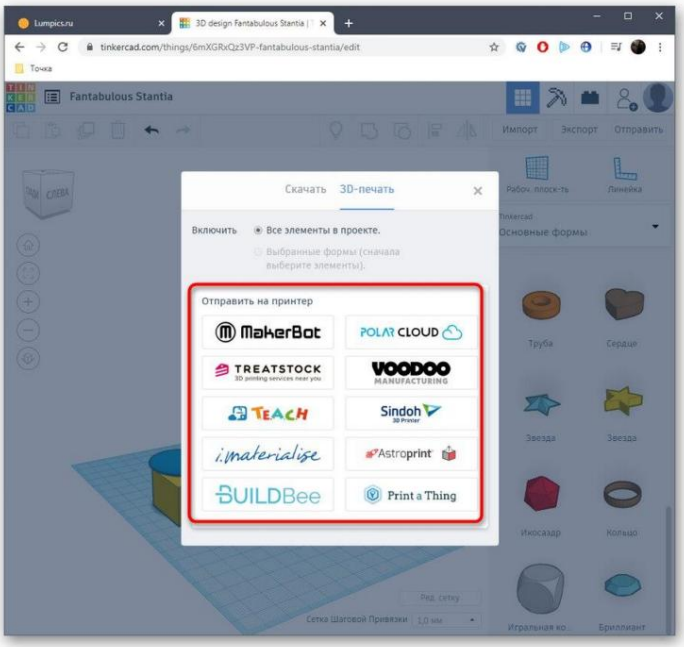
7. Available 3D printing formats will be displayed in a separate window.



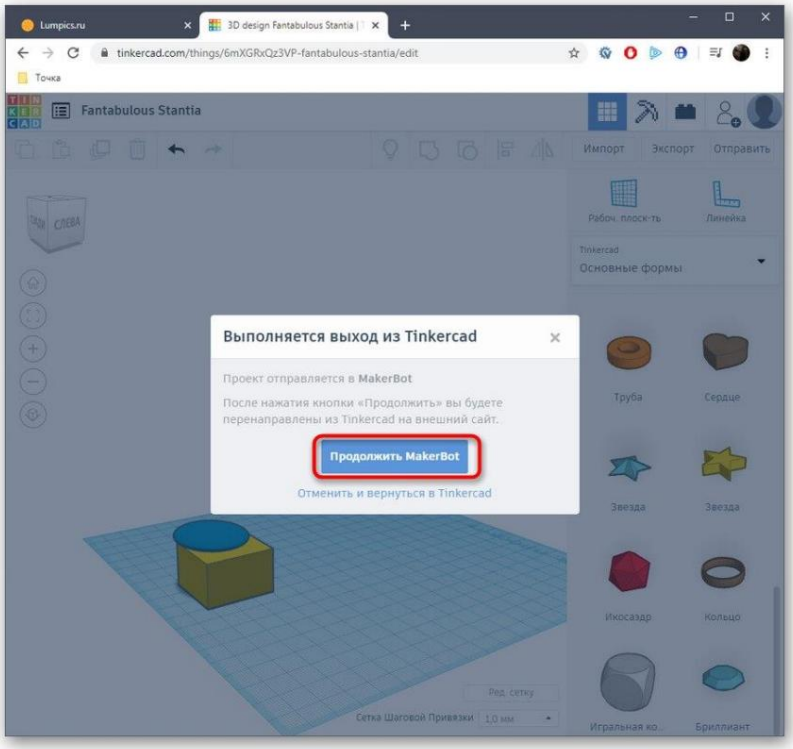
8. Once selected, an automatic download will begin.



9. If you do not want to download the file and you can send the print job immediately, go to the *"3D Printing" tab* and select the printer there.



10. There will be a transition to an external source and then the preparation process will begin and task performance.

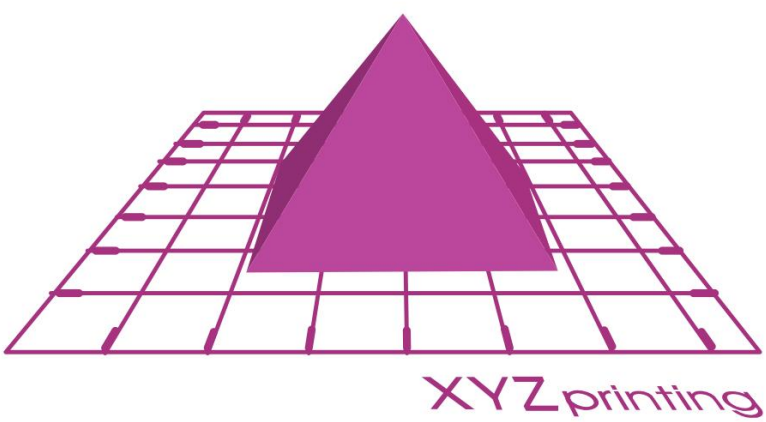


We can not look at absolutely all popular web services for 3D modeling, so we mentioned only one of the best and optimized for 3D printing. If you are interested in this method, just search sites through a browser to find the best option for yourself.

This was all the information about creating a model for printing on a 3D printer, which we wanted to share in one lesson. Then you just need to load the object file into the job preparation software, connect the printer, and start printing.

CHAPTER 4

Working with XYZmaker



4-1 Keychain with name

- Difficulty ★☆☆☆☆

4-2 Wrench

- Difficulty ★★★☆☆

4-3 Penguin

- Difficulty ★★★★☆

CHAPTER 4-1

Keychain with a name short guide

- Difficulty

- Objects used (from the toolbar on the left)

1. Geometric figures

Geometric Figure

Cube

Tube

2. Text tool

Tools

Text Generator

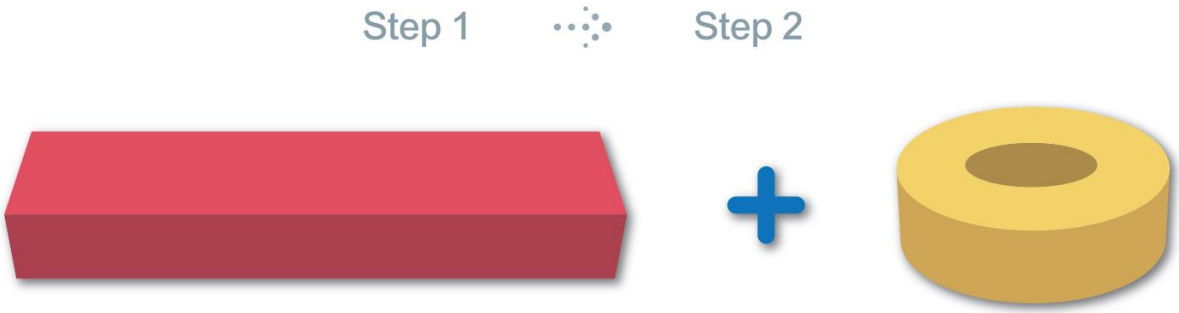
- Site processing
  (from the toolbar on the left)

Land

- Object commands (pop-up window)

1. Command of the object position parameter

2. Object dimension parameter command (scale unlocked)



Step 1

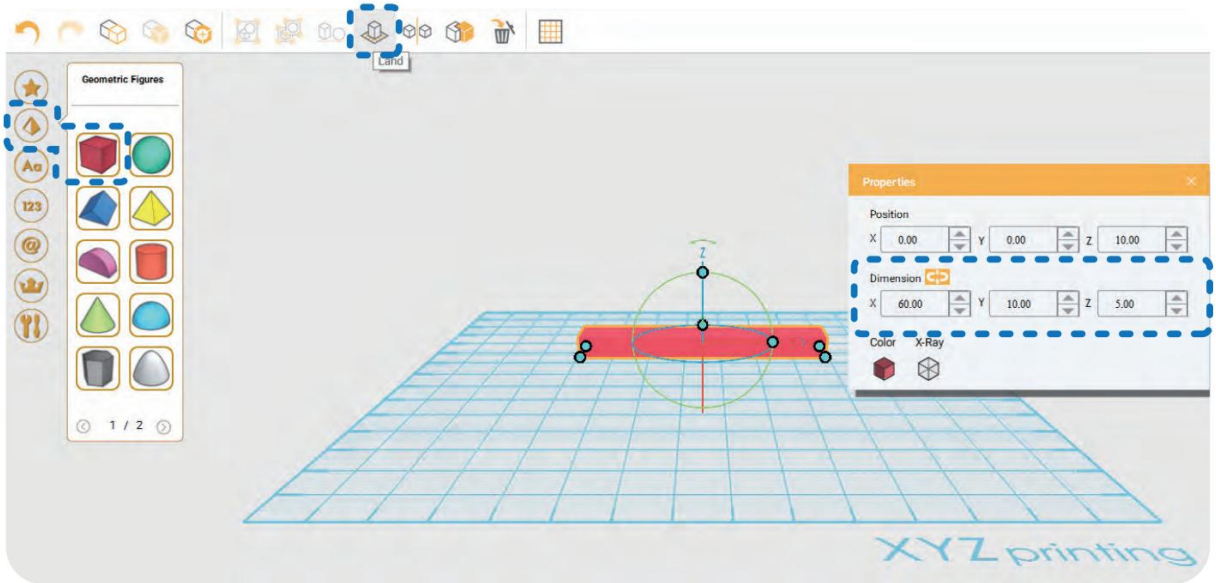
Click  geometric figure from the left menu

double click on a cube.  After the cube came down,

adjust the length, width and height in

size box (X: 60, Y: 10, Z: 5)

Click  on top and place on the desktop

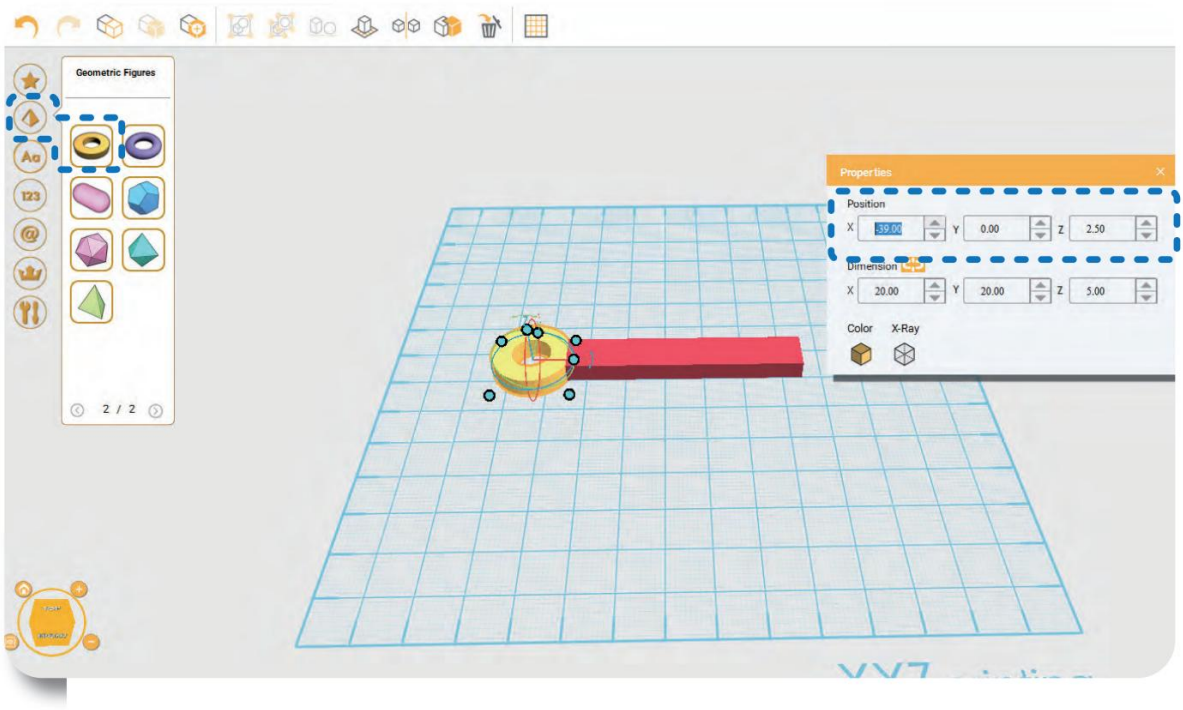


Step 2

Click  geometric figure from the left menu

double click on a figure  Once it appears,

set to position (X: -39, Y: 0, Z: 0).



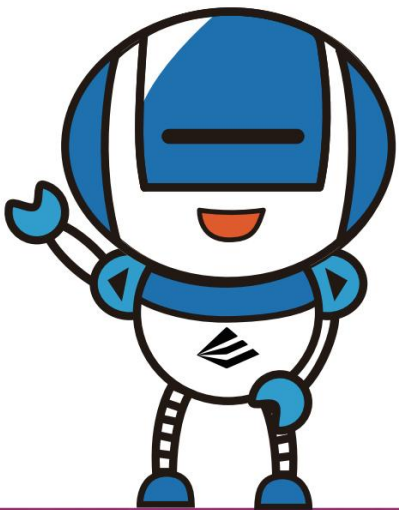
TIP:

If the object does not appear on the desktop

desktop, select both objects at once

and kick 

to appear on the desktop ~

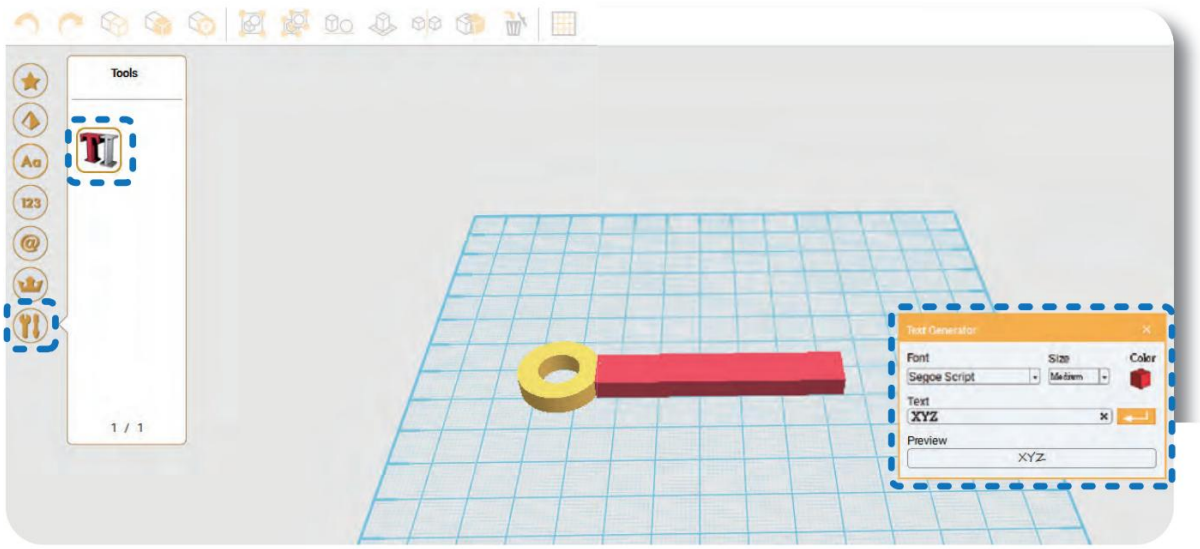


Step 3 ··· Step 5

Step 3

Click  To from the left menu and click on 

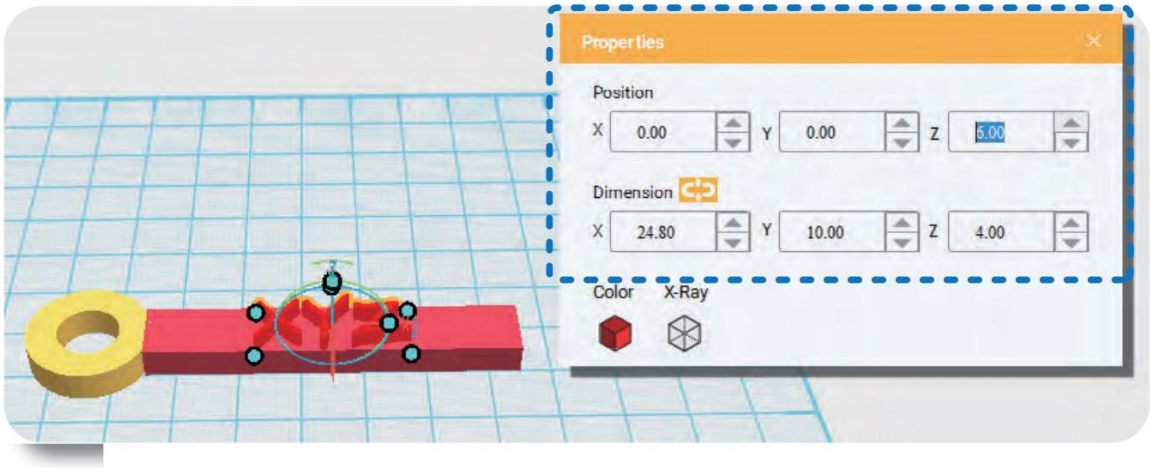
text generator. Once a small one appeared window, select the desired font in the Font field, select Medium as Size and enter the desired text in the Text column. After confirmation click the arrow Enter.



Step 4

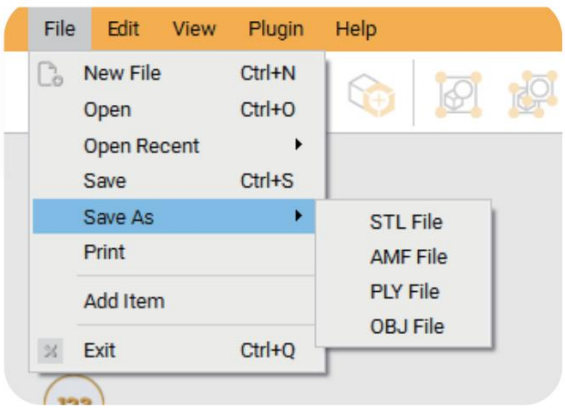
When the text appears, a small one will open window. Set on

position (X: 0, Y: 0, Z: 6). (More than 4 are allowed. The size of the text may vary depending on the font. See for reference

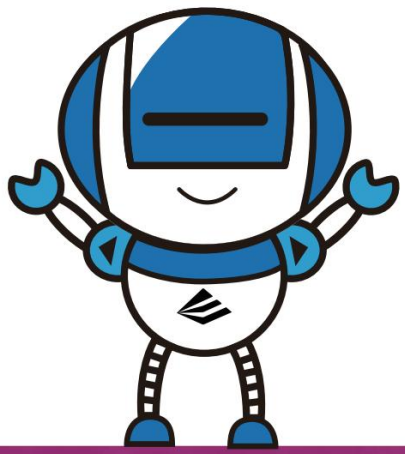


Step 5

Once your keychain is complete, click File in the upper left corner. After clicking Save, select the desired file format to finish the modeling step.



Ready!!

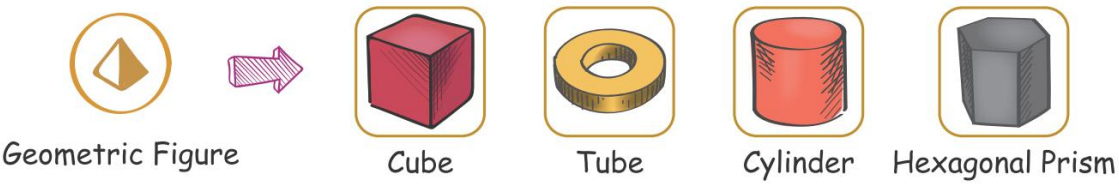


CHAPTER 4-2

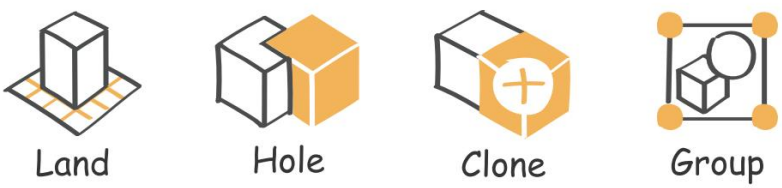
Wrench modeling

- Difficulty 

- used figures - geometric objects
  (from the bar on the left)



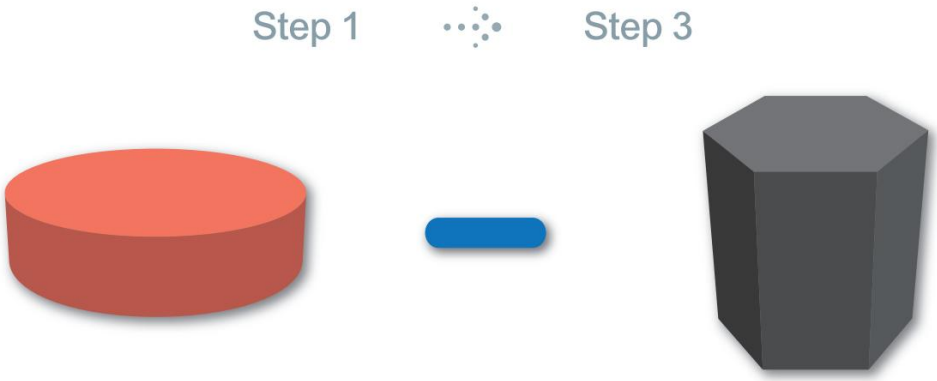
- Processing of objects
  (from the bar on the left)



- Object commands (pop-up window)

1. The command of the parameter of the position of the object

2. Object dimension parameter command (scale unlocked)



Step 1

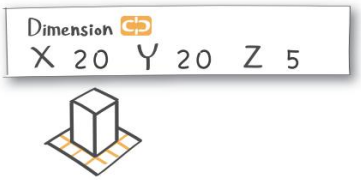
Geometric figures on the left

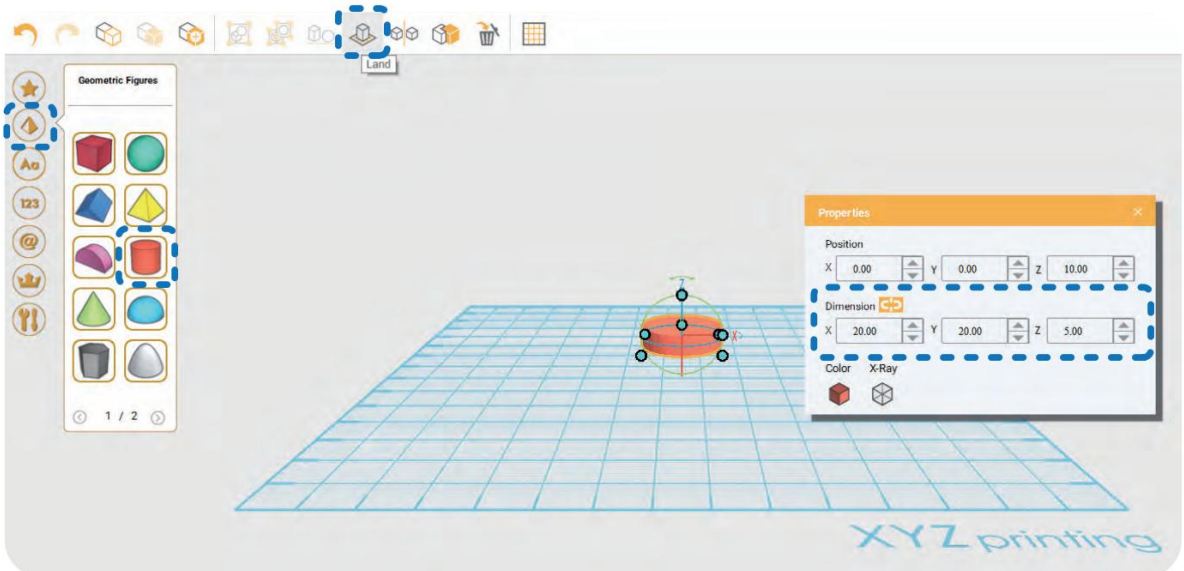
Cylinder. Then

adjust length, width and height

size box (X: 20, Y: 20, Z: 5). Click

on top and place on the desktop.



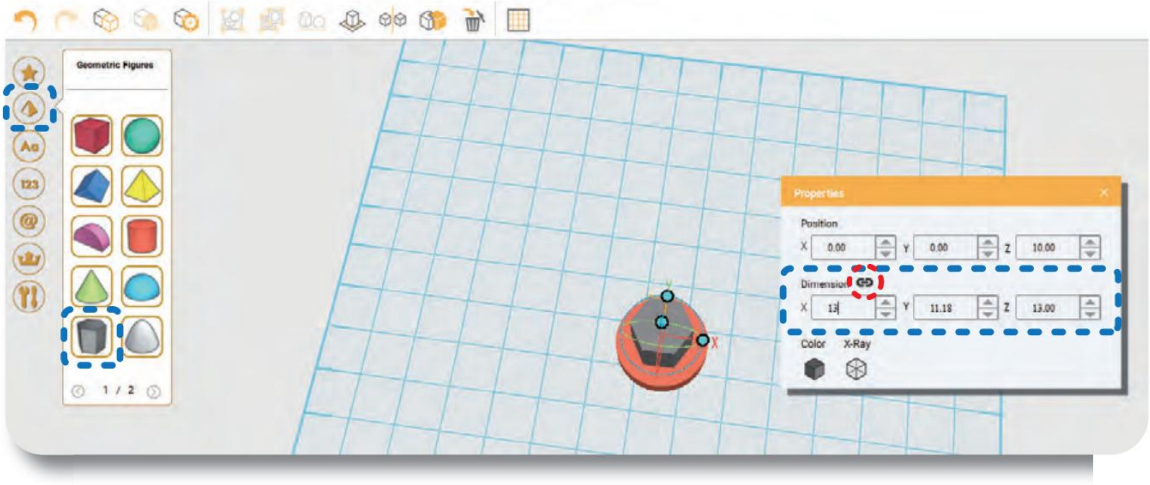


Step 2

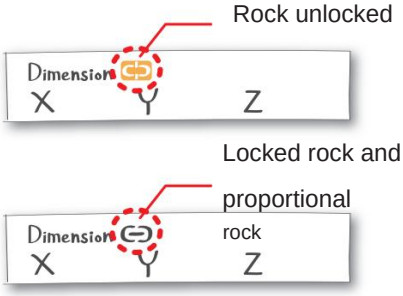
Click  left and select Fi  prism.Then,

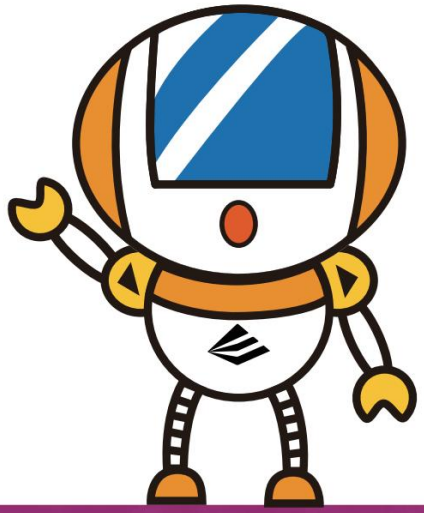
Lock the scale in 

Dimensions (X: 13; Y and Z are scale proportionally). Unlock.



ADVICE:



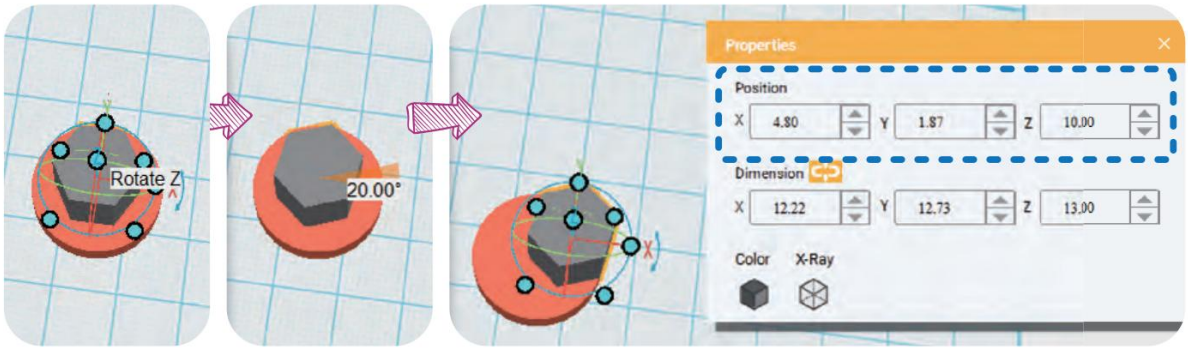


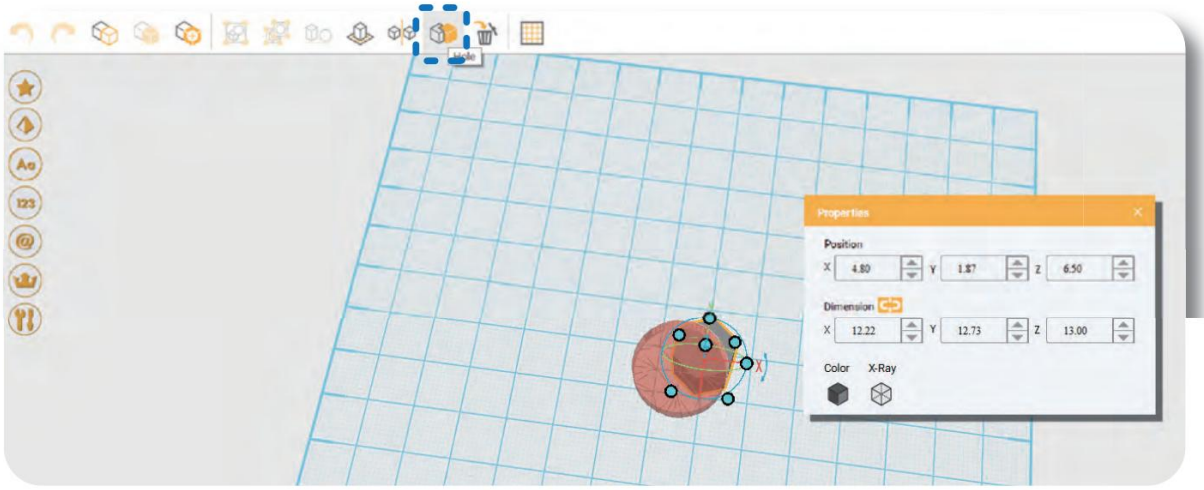
Step 3

Rotate the hexagonal prism around the Z axis by 20 degrees and place in the correct position

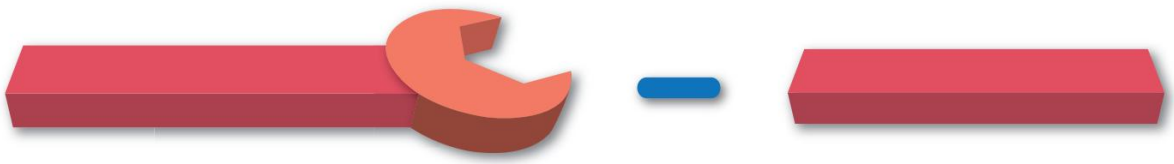
Position
X 4.8 Y1.87 Z6.5

(Position X: 4.8, Y: 1.87, Z: 6.5). Click  cylinder and select the hole. Select the prism and click again  hole to complete the cut.





Step 4 ··:·· Step 6



Step 4

Click  Geometric figures on the left

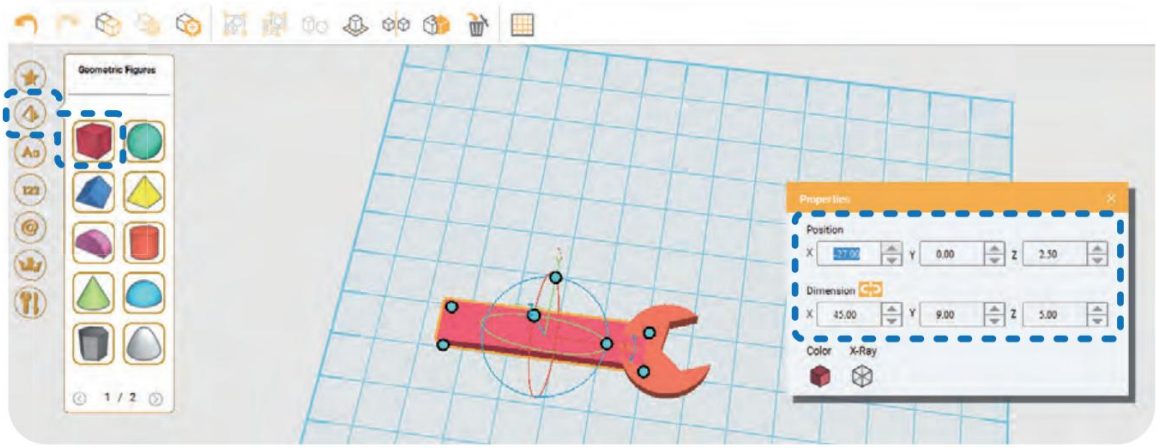
double click on 

adjust the length, width and

height (dimensions X: 45, Y: 9, Z: 5).

Put in the right position

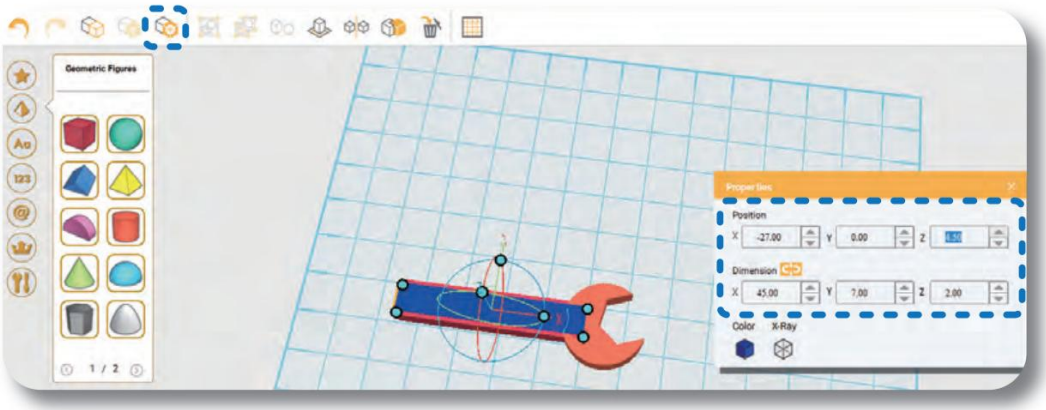
(Position X: -27, Y: 0, Z: 2.5).



Step 5

Click  Geometric figures on the left double click on

(Dimensions X: 45, Y: 7, Z: 2). Move to the correct position (Heading Z: 4.5).



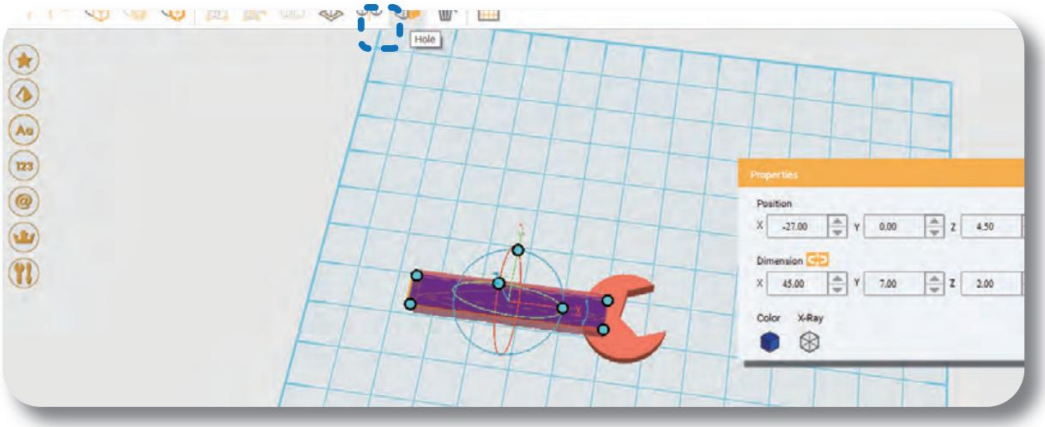
Step 6

Click on the first cube and click on  hole.

Then click on the second cube, 

click the hole again to finish

the operation.

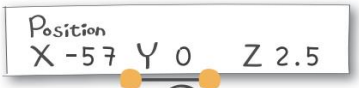


Step 7

Step 7

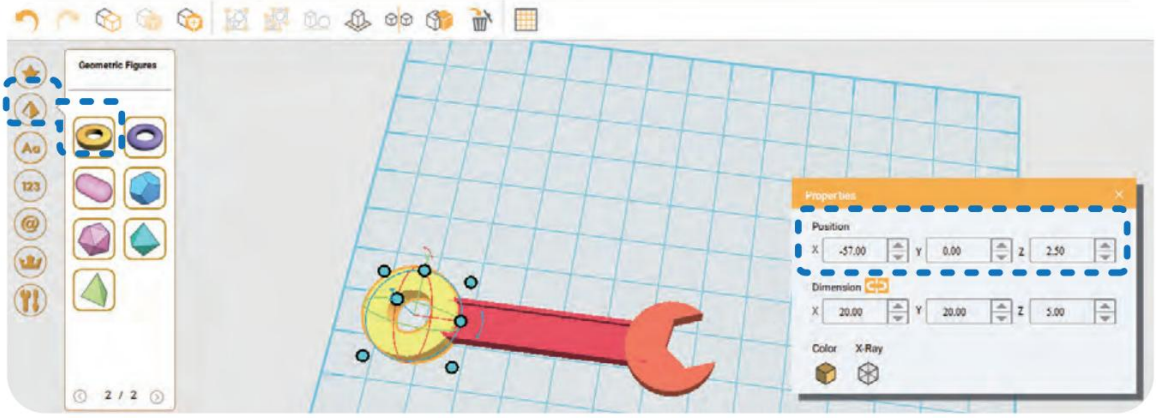
Click  geometric figures on the left

double click on  pipe. When it appears,

put in position  (position X: -57).

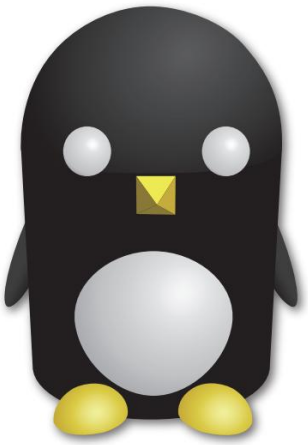
Select all objects and use.  Group all objects

for more convenient editing. Wrench modeling is complete.



CHAPTER 4-3

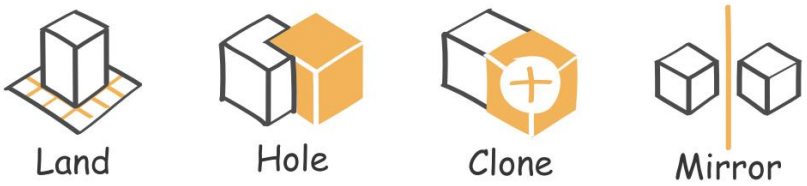
Penguin modeling



- Difficulty 

- Used objects - geometric shapes
(from the left menu)

- Site processing
(from the toolbar on the left)



- Object commands (pop-up window)

1. Command
of the object
position parameter

2. Object dimension
parameter command
(scale unlocked)

Step 1 ··· Step 2

Step 1

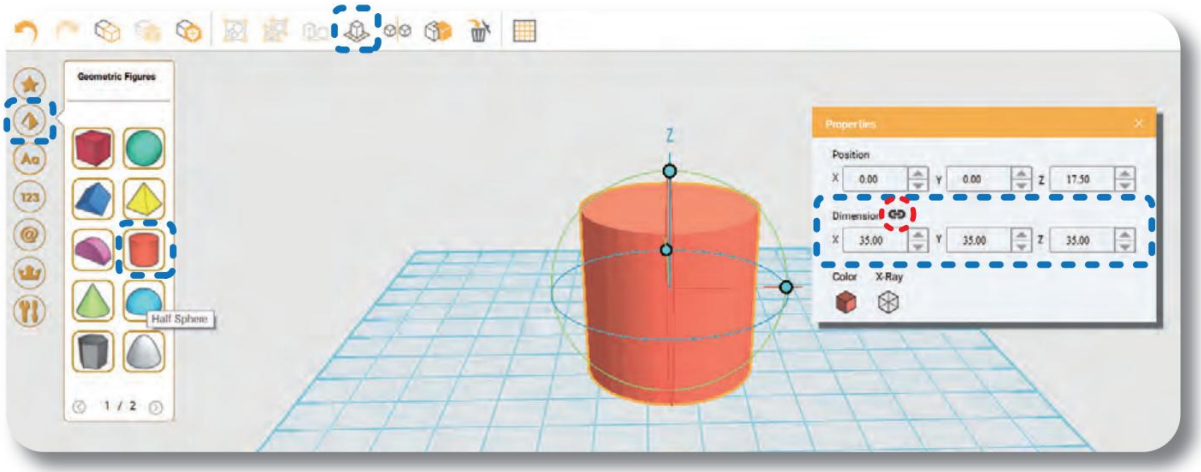
Click  geometric figures on the left

double click  cylinder. When a cylinder appears,

Lock the rock

dimensions (X: 35, Y: 35, Z: 35).

Click  on top and place on the desktop.



Step 2

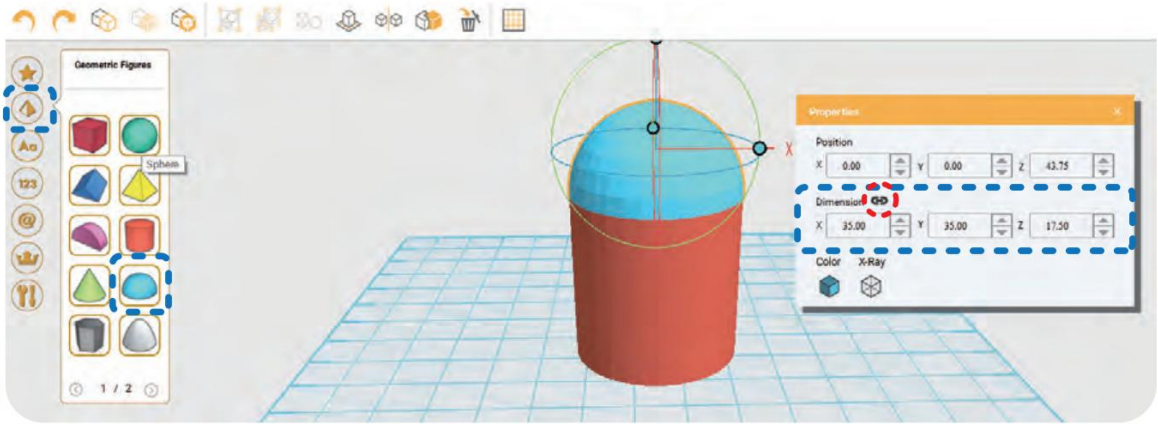
Click  geometric figures on the left

double click  Hemisphere. When it appears,

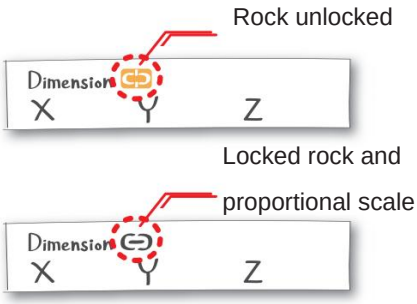
lock the scale of 

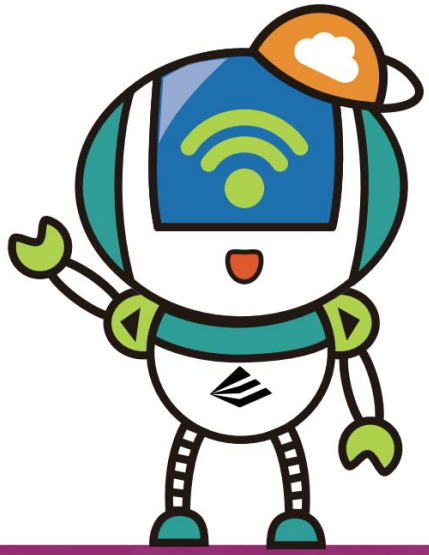
dimensions (X: 35; Y and Z are scaled proportionally). Move to position:

(Position X: 0, Y: 0, Z: 43.75).

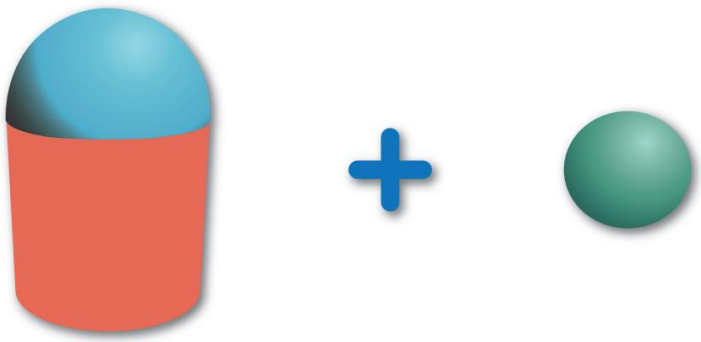


ADVICE:





Step 3



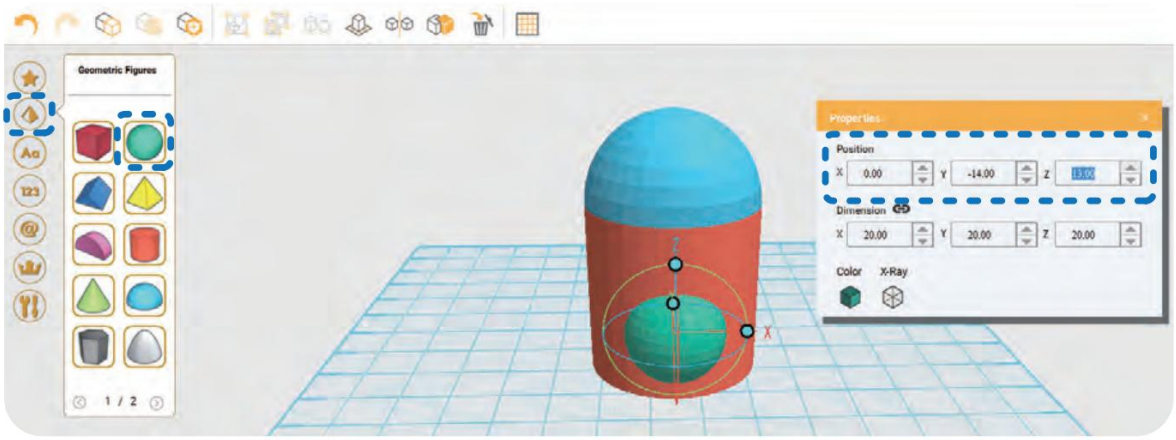
Step 3

Click  geometric figures on the left

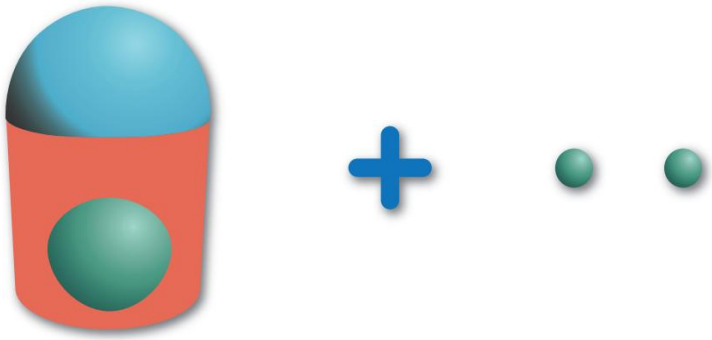
double click on  sphere. When it appears,

move to position

(Position X: 0, Y: -14, Z: 13).



Step 4 ··· Step 5

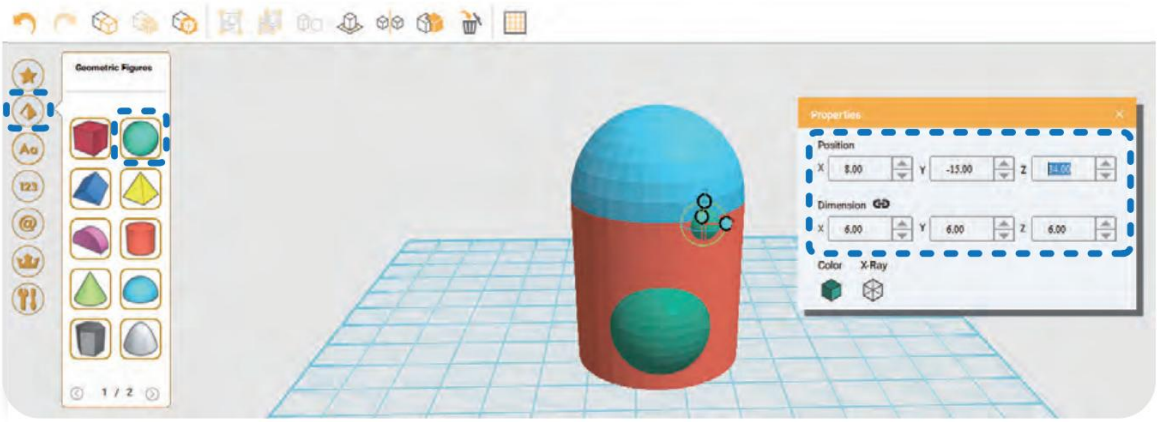


Step 4

Click  geometric figures on the left double click on  sphere. When it appears, Lock the rock 

in sizes (X: 6; Y and Z are scaled proportionally). Then move to position 

(Position X: 8, Y: -15, Z: 34).

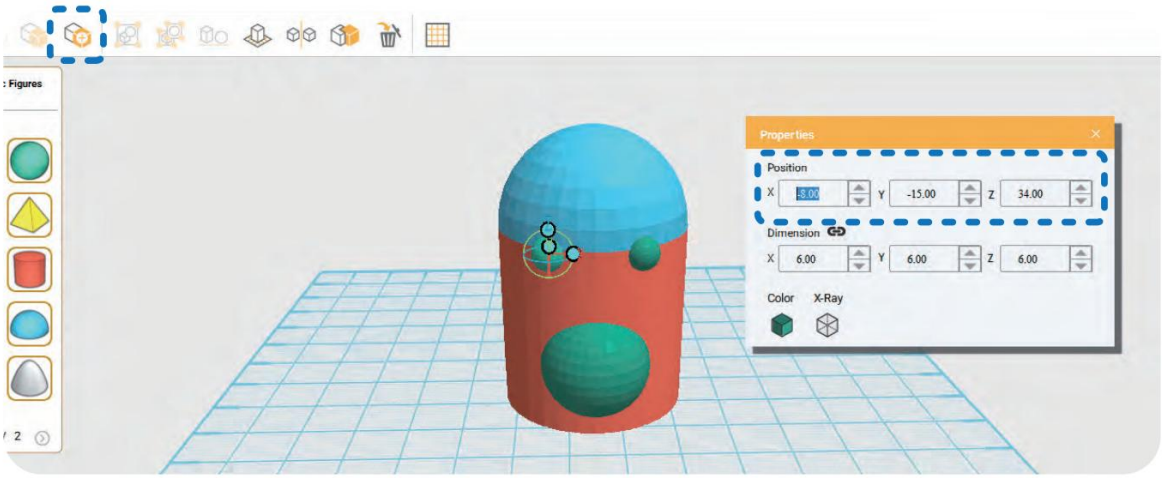


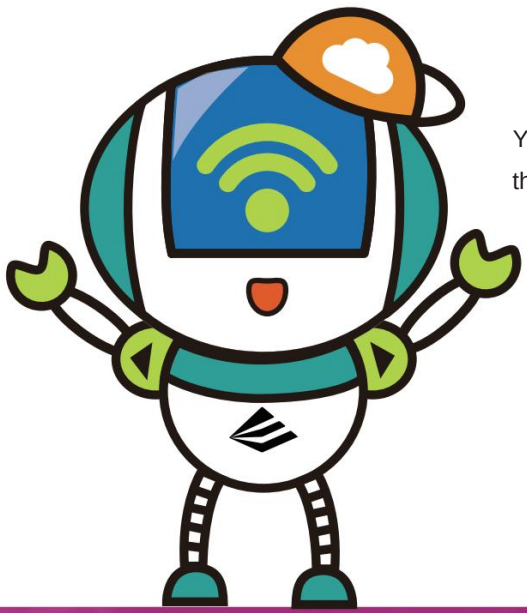
Step 5

Click  Clone in the top row to clone another small sphere in the same place

and move to the correct position

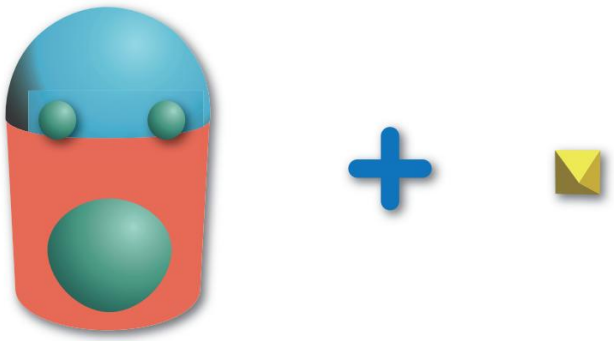
(Position X: -8, Y: -15, Z: 34).





You created half of the adorable penguin ~

Step 6



Step 6

Click  geometric figures on the left

double click on  Pyramid. When it appears, lock

scale with dimensions 

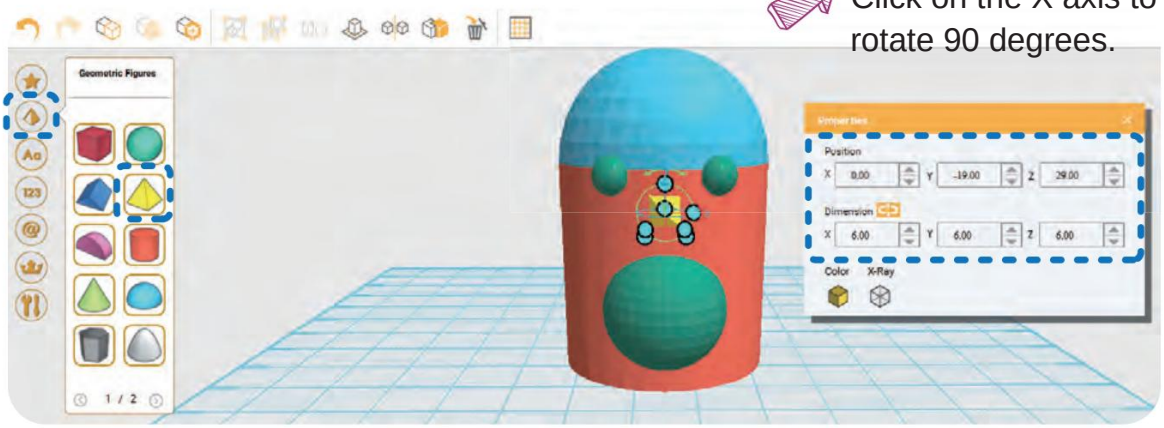
(X: 6; Y and Z are

scale proportionally. Unlock. Rotate about the X axis of

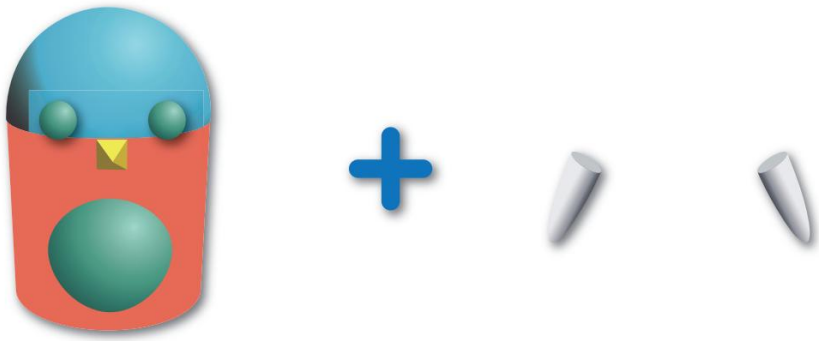
90 degrees and move to position

(Position X: 0, Y: -19, Z: 29).

 Click on the X axis to rotate 90 degrees.



Step 7 ··· Step 8

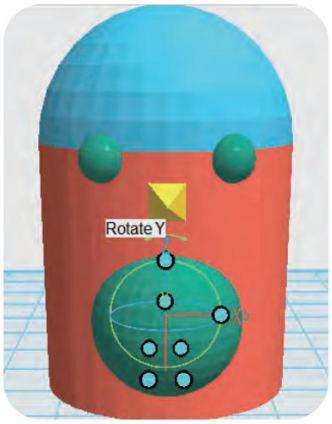


Step 7

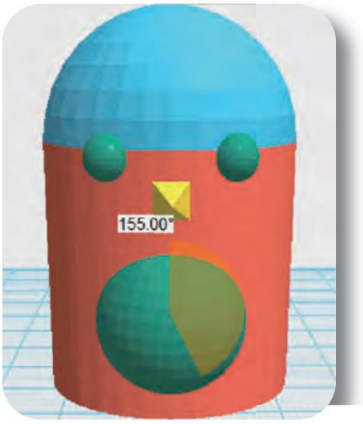
Click double click on  geometric figures on the left Paraboloid. When it appears, adjust length, width and height in

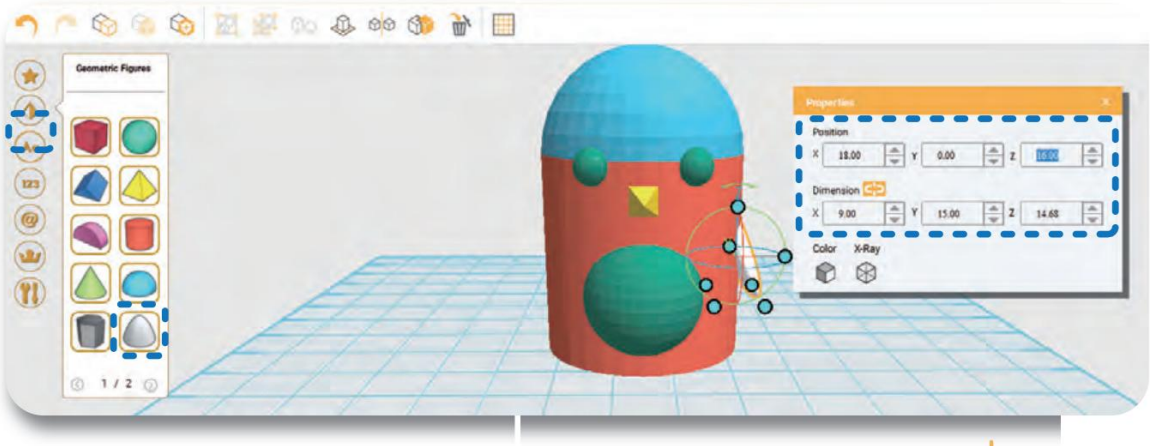
(Dimensions X: 5, Y: 15, Z: 15). Rotate around the Y axis at 155 degrees and move to the correct position

(Position X: 18, Y: 0, Z: 16).



Click on the Y axis to rotate 155 degrees



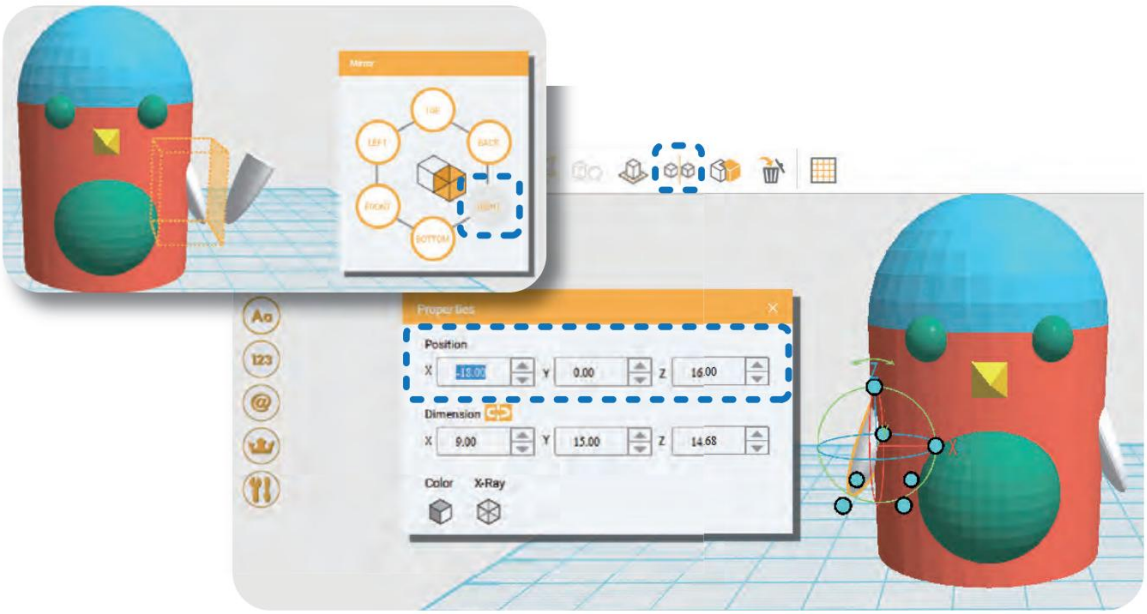


Step 8

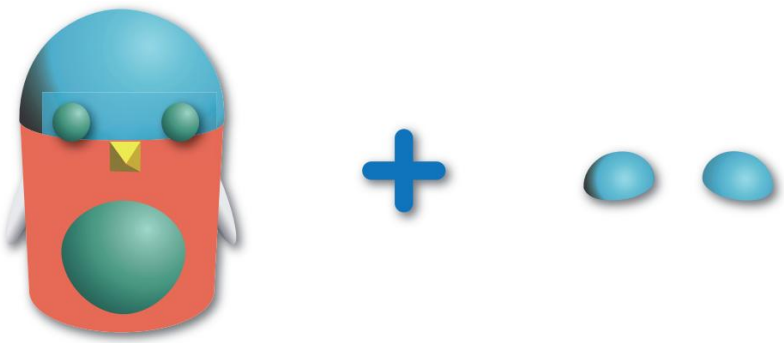
When the paraboloid is on the right, use  Mirror (if left, you will not be able to select it).

Then move the duplicate mirror paraboloid to position

(Change parameter X to -18).



Step 9 ··:·· Step 10



Step 9

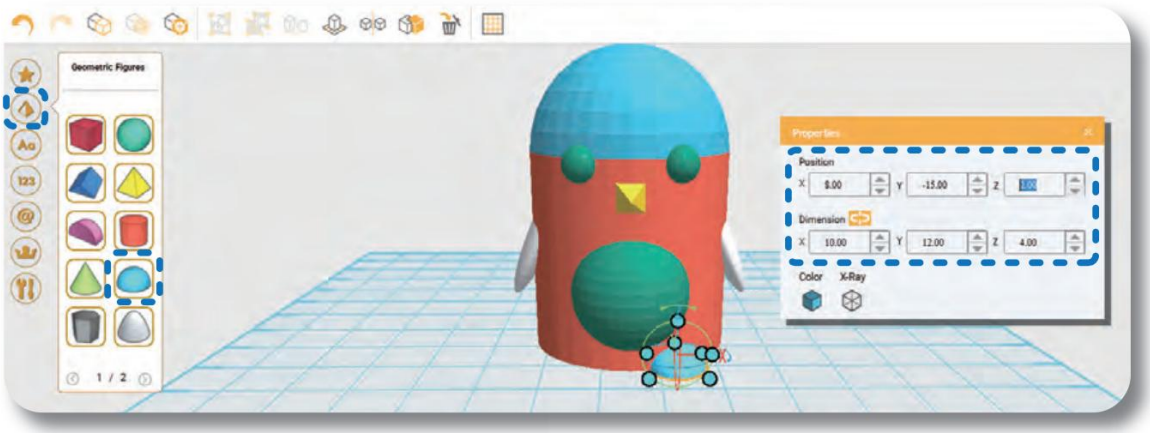
Click  geometric figures on the left double click on  hemisphere. When it appears, adjust length, width and height:

Dimension
X 10 Y 12 Z 4

(dimensions X: 10, Y: 12, Z: 4), and move in position

Position
X 8 Y -15 Z 2

(position X: 8, Y: -15, Z: 2).

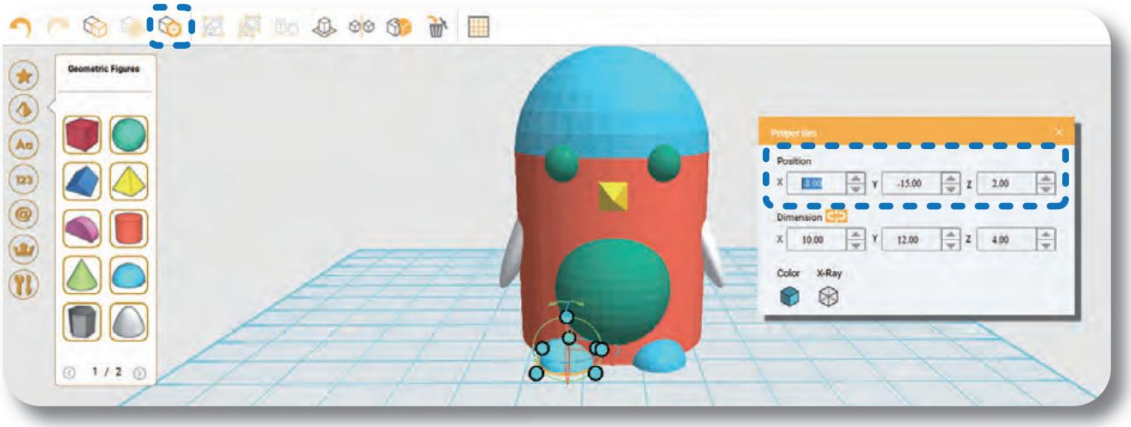


Step10

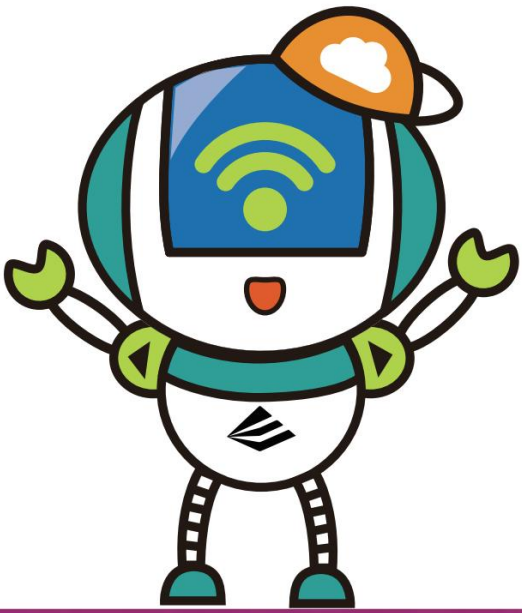
Click  clone in the top row to clone hemisphere and move to position.

Position
X -8 Y 0 Z 0

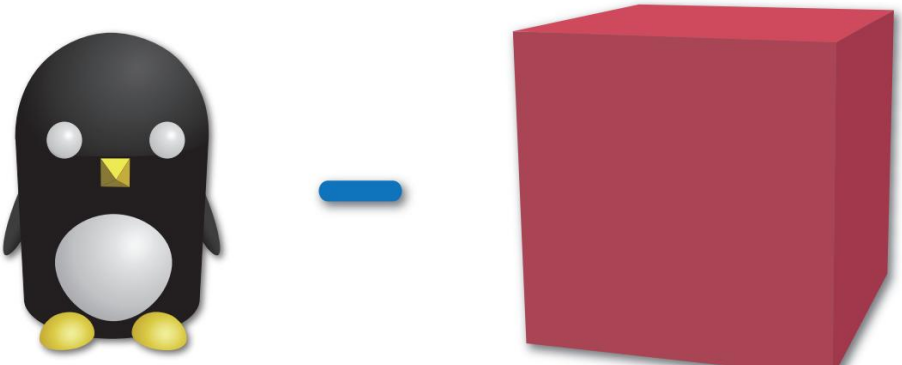
(Change parameter X to -8).



Your lovely penguin is ready !!
You can color it in different colors. In the extended instructions you can create a custom ornament ~



Step 11 Step 12



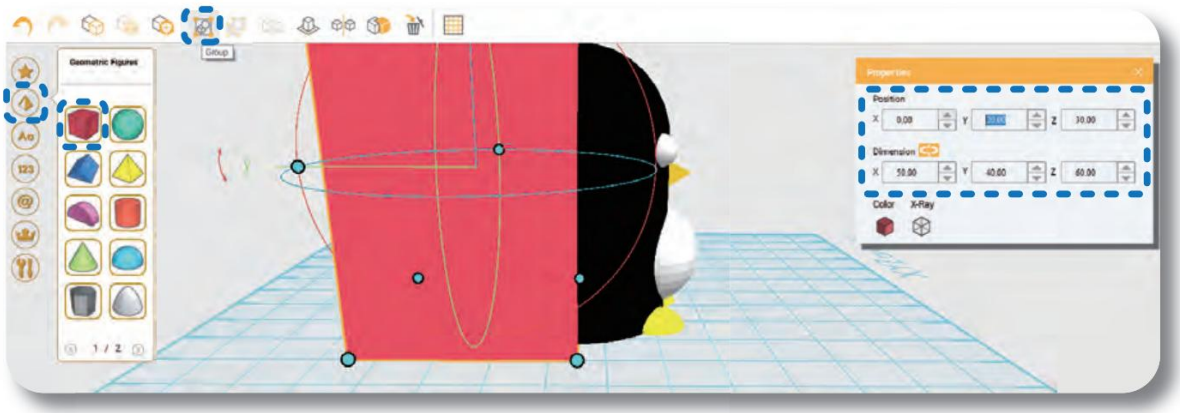
Step11

Group all the penguin objects. Then click on 

 Geometric shapes on the left and double click on

 Cube. When it appears, adjust the length, width and height on  (Dimensions X: 50, Y: 40, Z: 60).

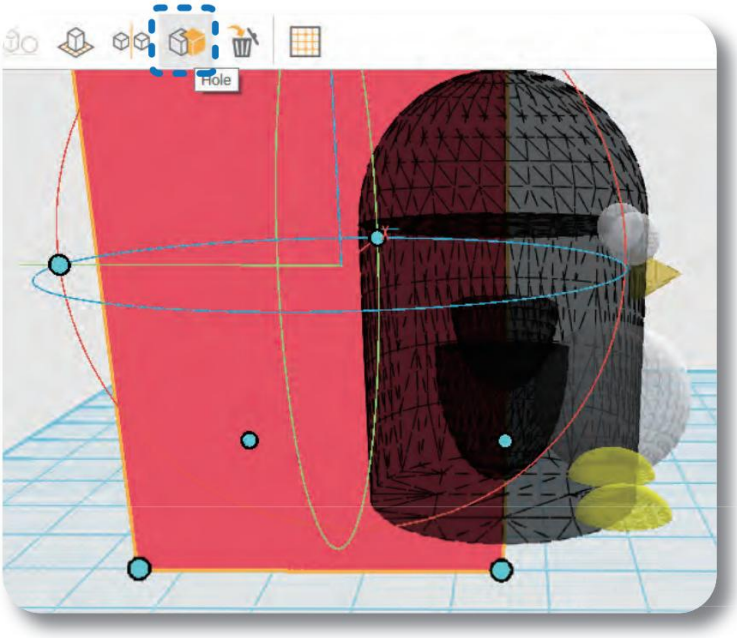
Position:  (Position X: 0, Y: 20, Z: 30).

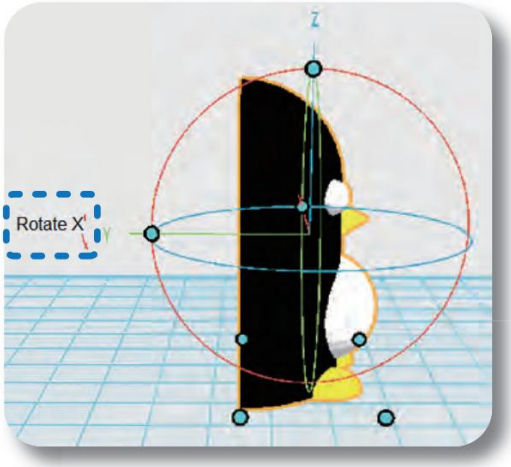


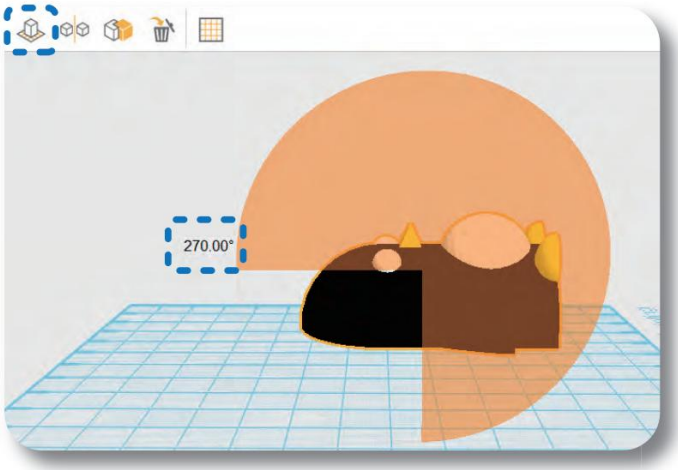
Step12

Click on the penguin and select  hole.

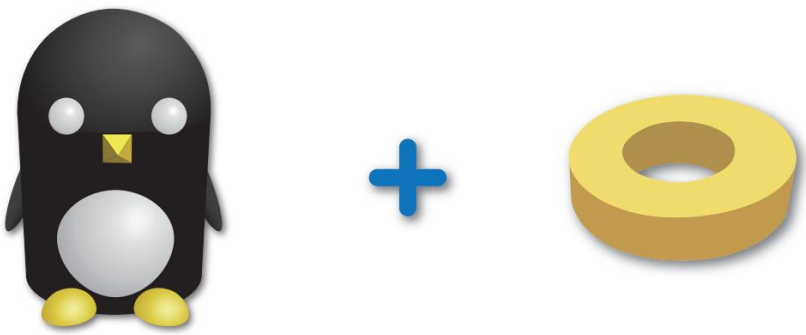
Then click on the cube and click  again on the hole to complete the cut. Rotate the cut penguin around the X axis to 270 degrees and click  to put it on the desktop.





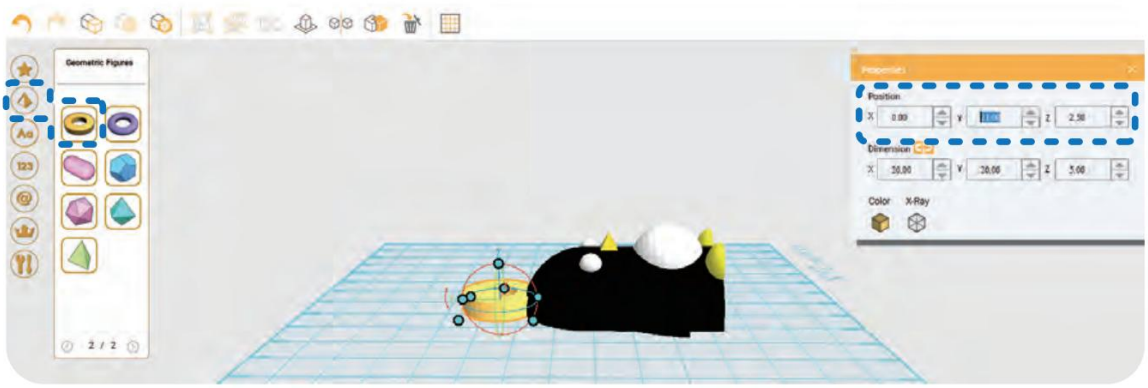


Step 13

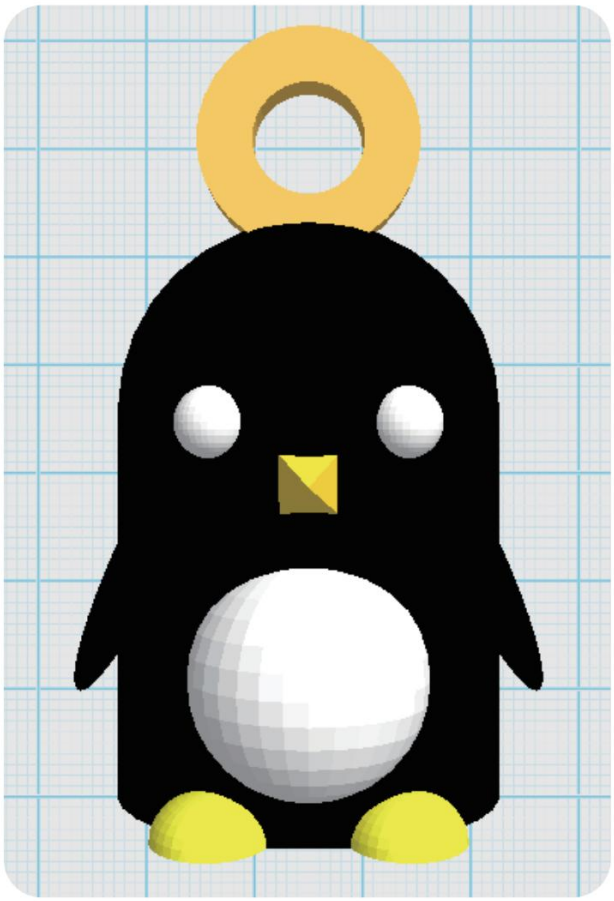


Step13 Click  geometric shapes on the left, double click on a pipe.  When it appears, position: 

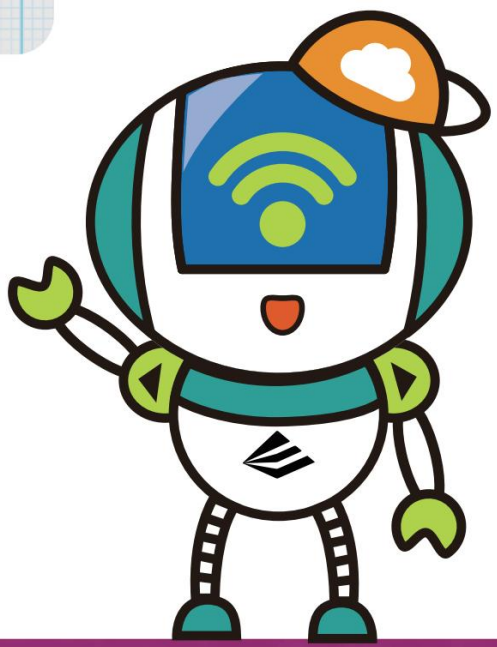
(Position X: 0, Y: 21, Z: 2.5) to complete the simulation of the penguin.



Final ~~~



With a few easy steps,
can become a
wonderful ornament !!



CHAPTER 5

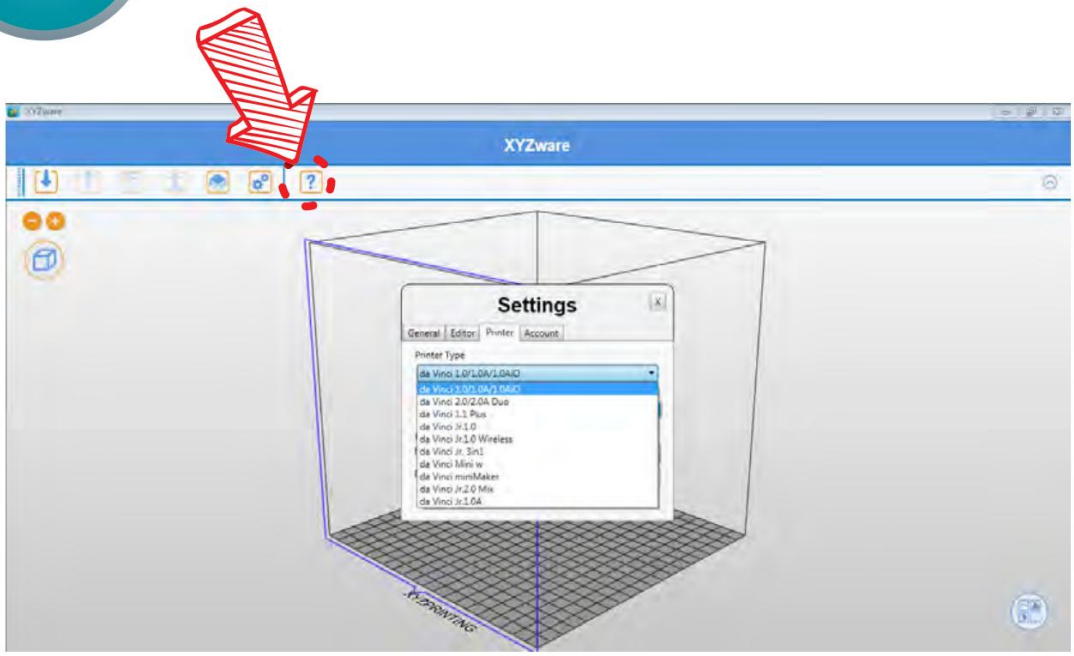
Printing

Step 1

Launch the [XYZware] software.

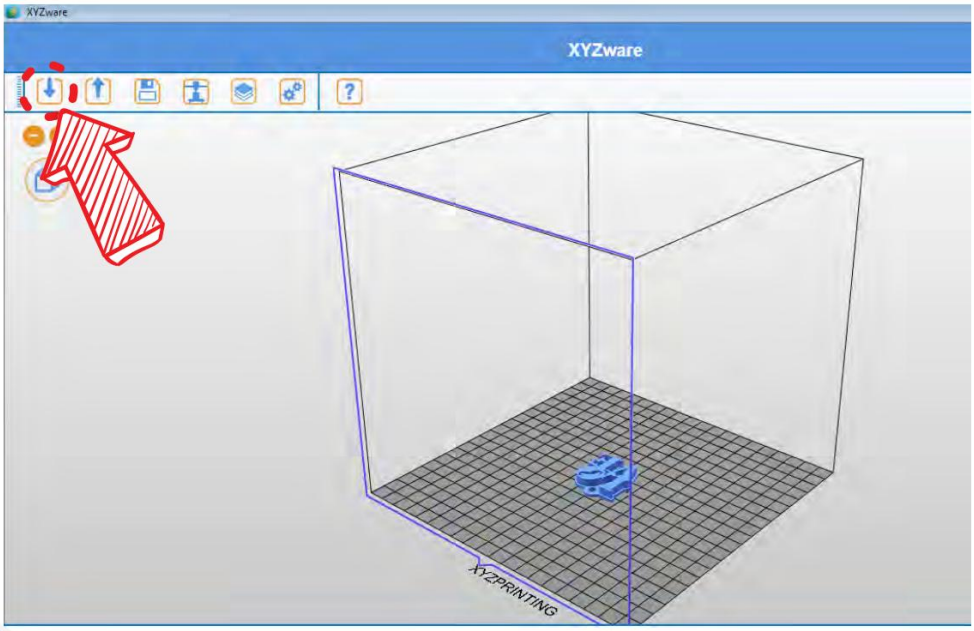
Step 2

Select “Settings” and select the printer model / in the example: da Vinci Mini w /.



Step 3

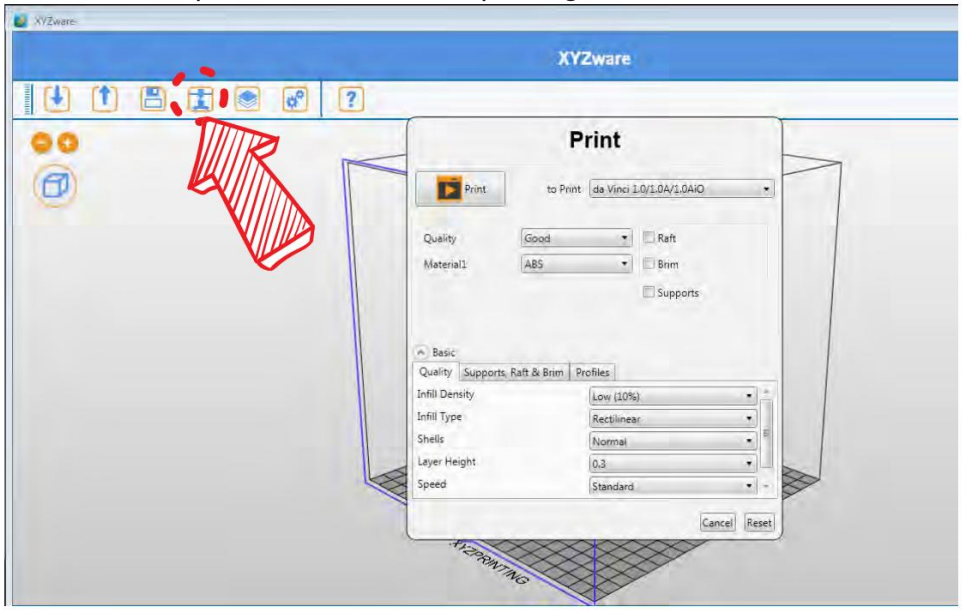
Click “Import File” and select the model you want to print or drag the STL file in the window.



Step 4

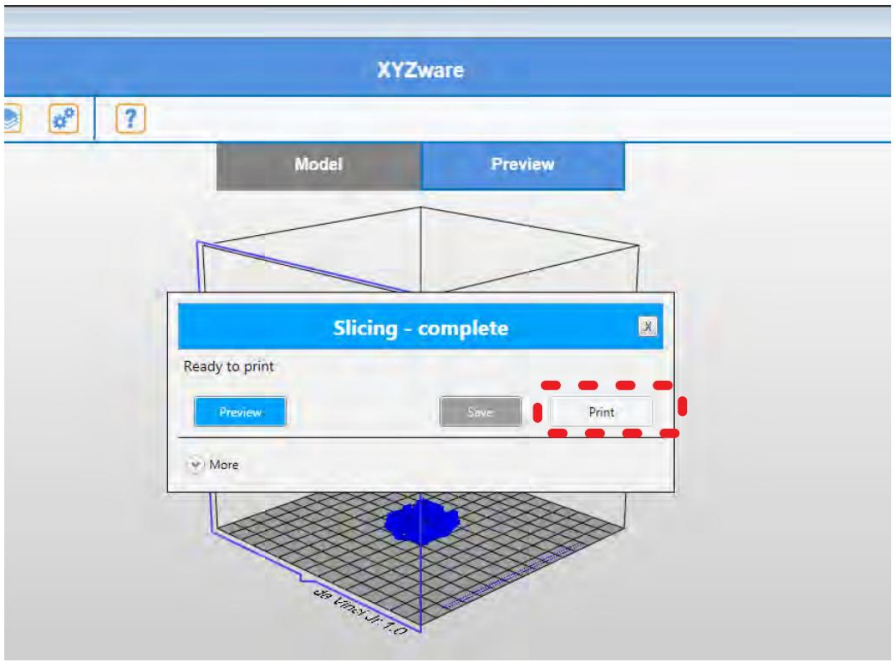
Click "Print". You can make custom settings in the window. Click "Advanced" to

set more parameters and start printing.



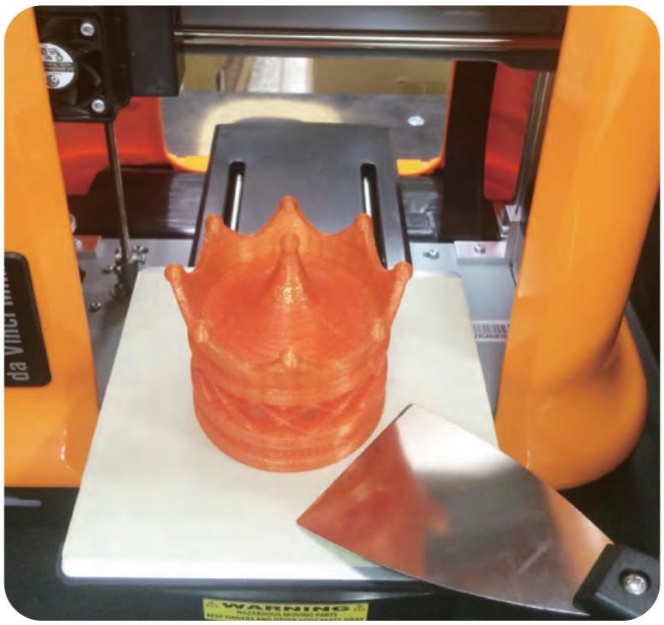
Step 5

When the cutting process is complete, click Print, for
to print the model.



Step 6

When printing is complete, you can pick up the model from the platform by carefully peeling it off
using the spatula provided with the printer.



As with setting up a 3D printer, we'll start with a few definitions.

Slicer - is a program that prepares a selected 3D model for printing. In fact, the whole instruction below will be about one of the variants of this program. The task of the slicer is to process the 3D model in such a way as to present it as a set of layers. In addition, it is the slicer that generates a file that is readable and understandable from the 3D printer. All generated 3D printing parameters are set in this generated file, but you need to set up the printer before starting to prepare a model.

Layer height / thickness - the program described above, as we understand, presents a 3D model as a set of layers. Usually the thickness of the layer varies between 0.04 mm to 0.2 mm. The detail of the printed model depends on the height of the layer. But it is important to remember that one of the factors that affects the speed of printing is the height of the layer. That is, the smaller the layer height, the longer the model will be printed. The reason is trivial: more layers - more levels - more printhead movements - more time.

Props - Props are 3D structures that are generated by the slicer. They are needed in places of the model where the model is potentially "hanging". This is due to the nature of FDM printers. This will be described in detail below.
Fill - Fill is a structure generated by the slicer, which determines to what extent and with what structure the model will be filled. When the model is not expected to experience high kinetic loads, the filling may be either minimal or non-existent. The cases (when and what types of fillings to use) are described below. Filling also affects the speed of printing, so it is used only when needed, thus saving time and filament.

Wall thickness - wall thickness is one of the determining factors for the strength of the model. Similarly, the thicker the wall of the model, the longer it will take to print. Just as the filling changes only when needed, we still save time and filament.

3d model (suitable for printing) - 3D models, suitable for printing, are a file of a specific format ".stl". This file contains the geometry of the 3D model, but it is not yet ready for printing, and as described above, it needs to be processed by Slicer. The slicer will generate an ".gcode" file that contains print parameters and the model, broken down into layers and converted into instructions for the 3D printer.

Slicer
We will use a CURA slicer for the instruction . This program is open source and you can use it for free. Download and install the program on your personal computer.

Preparation of the program.
Once the program is installed, we need to set it up for your printer. In the examples below, the settings are for the specific two models we offer, if your printer is different, then the procedure is similar.

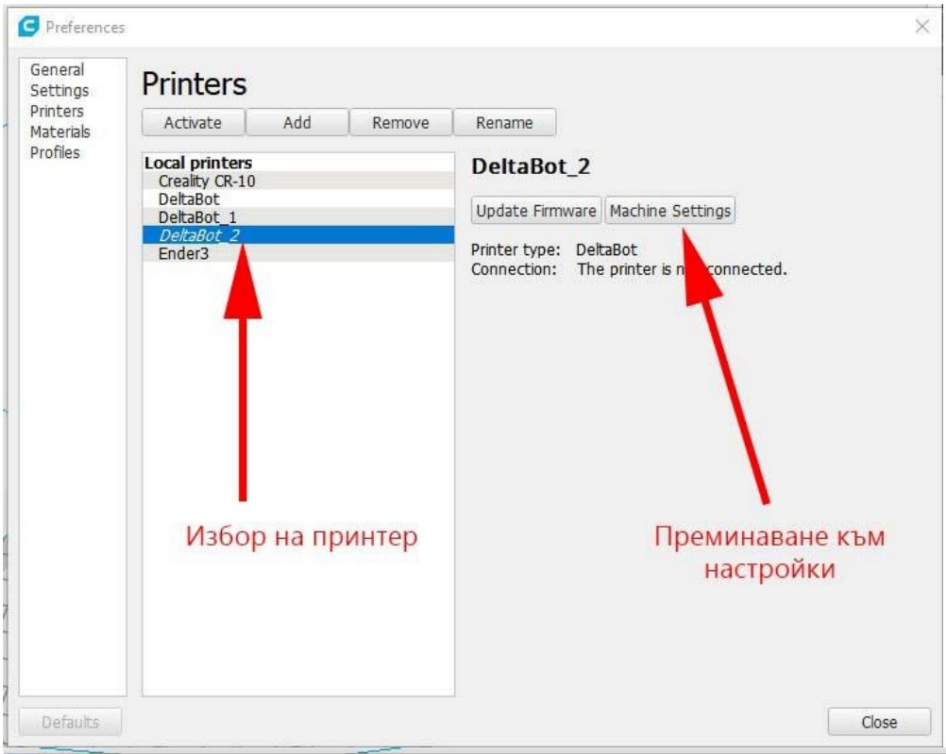
Open the program and follow the steps:

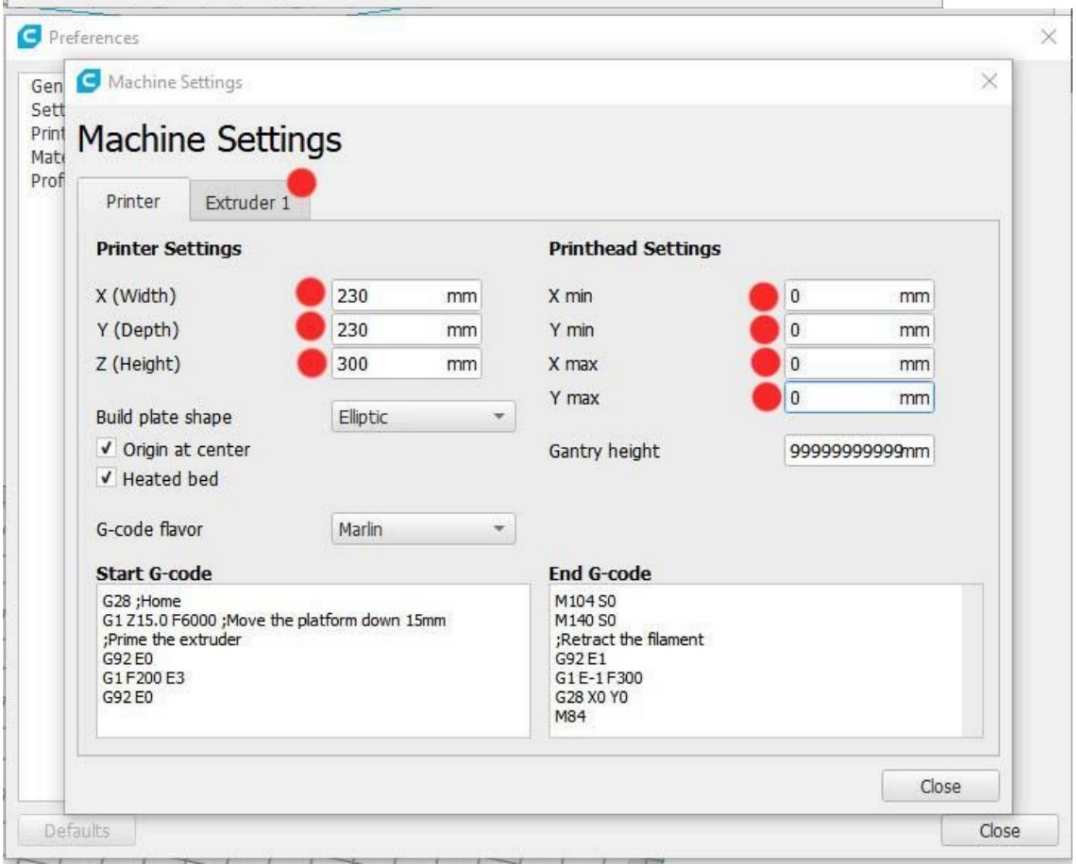
Ender 3

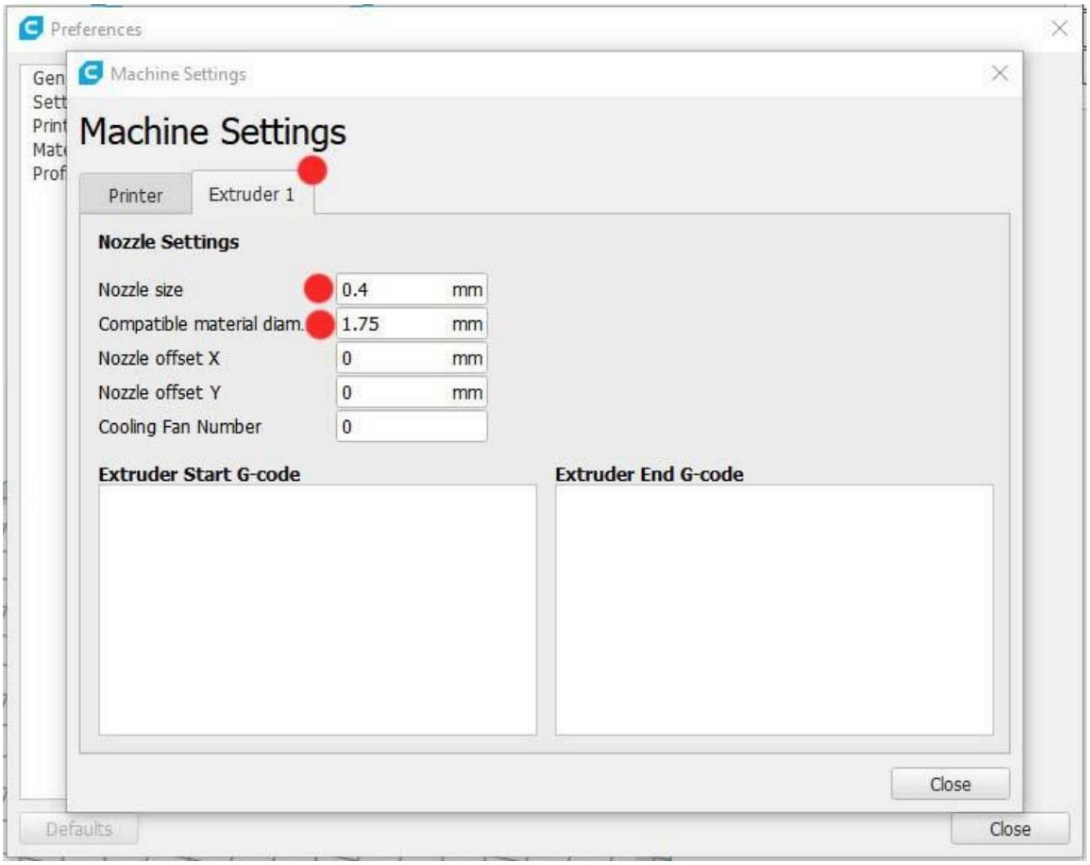
1. Go **to Settings -> Printer -> Add Printer…;**
2. Select the Other drop-down menu ;
3. Select **Creality Ender-3** from the list and click the **Add printer** button (bottom right) the profile is preset;

Anycubic Kossel Linear Plus

1. Go **to Settings -> Printer -> Add Printer…;**
2. Select the Other drop-down menu ;
3. Select **DeltaBot** from the list and click the **Add printer** button (bottom right);
4. Once the printer is added, set to go, go to:
   Settings -> Printer -> Manage Printers. Select the newly added printer and select **Machine Settings;**
5. In the menu that appears, fill in all the fields as shown in the images below down


Preferences
General Settings
Printers
Materials
Profiles
Printers
Activate
Add
Remove
Rename
Local printers
Creality CR-10
DeltaBot
DeltaBot_1
DeltaBot_2
Ender3
DeltaBot_2
Update Firmware
Machine Settings
Printer type: DeltaBot
Connection: The printer is n connected.
Избор на принтер
Преминаване към настройки
Defaults
Close

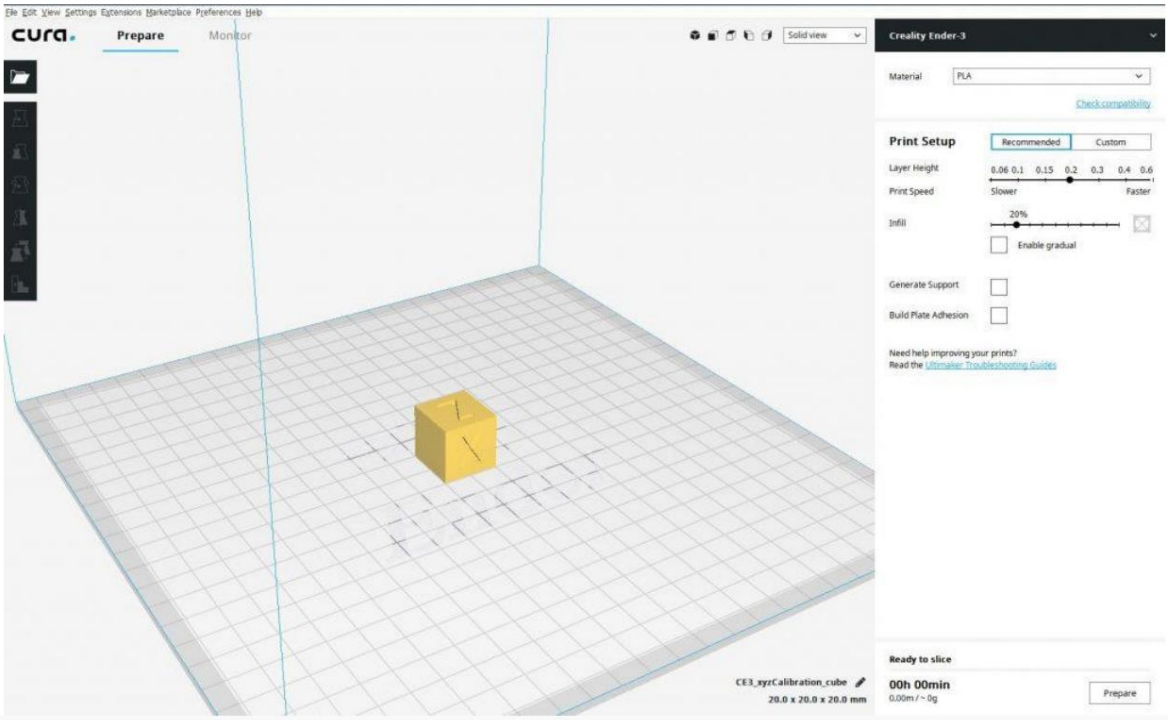

Preferences
Machine Settings
Machine Settings
Printer
Extruder 1
Printer Settings
X (Width) 230 mm
Y (Depth) 230 mm
Z (Height) 300 mm
Build plate shape Elliptic
Origin at center
Heated bed
G-code flavor Marlin
Printhead Settings
X min 0 mm
Y min 0 mm
X max 0 mm
Y max 0 mm
Gantry height 99999999999mm
Start G-code
G28 ;Home
G1 Z15.0 F6000 ;Move the platform down 15mm
;Prime the extruder
G92 E0
G1 F200 E3
G92 E0
End G-code
M104 S0
M140 S0
;Retract the filament
G92 E1
G1 E-1 F300
G28 X0 Y0
M84
Close
Defaults
Close



Acquaintance with the slicer and preparation of the first seal.
Before we start getting acquainted with the slicer, it is necessary to have a 3D model with which to visualize the process.

There are several free resources from which you can search and download models. The models we provide are already sliced and can be released directly for printing.

You can use one of the most popular sites to download models: https://www.thingiverse.com/.
For test printing we will use a test cube, which can be downloaded here. After downloading the model, unzip the model, go to the files folder, drag the file from this folder to the prepared and configured for your printer.
You will get this:



Once the model is loaded, you can press the **"Prepare"** button and the program will process and prepare it for printing with the default settings. Instead of the **"Prepare"** button, **"Save"** will appear , with which you can save the finished file to a selected medium. You can copy the exported file to the printer's **SD** card and start printing.
But let's get acquainted with the basic settings of the slicer. Here it is important to remember that the settings will largely depend on the capabilities of your printer. The example below is without exotic or complicated settings. They will be explained in a separate tutorial.

View
You can manipulate with the mouse: rotate to view the object you loaded.

By pressing the scroll wheel, you can translate the view, and with the right button - to rotate the view.

The program also has a set of preset views, which are located at the top right of the view.

By clicking on each of the preset views, the view will rotate as specified on the button.

Next to the preset view set, you'll find a drop-down menu containing options on how to present the loaded model.

- **Solid view** - the model is presented as an object with solid and opaque walls;
- **X-ray view** - the model is presented with translucent walls;
- **Layer view** - the model is presented as a set of layers (in the beginning, when we described what is a slicer, this is explained);

The most commonly used drop-down menu option is Layer View. Select it to consider the following options.

If you bring the loaded model closer, you will notice that the program already visualizes it as a set of layers. If you use the sliders that appear, you can see how each of the layers will be built from the inside.

In the future, we will often turn to this view of the model to analyze potential printing problems.

Manipulations with 3D model

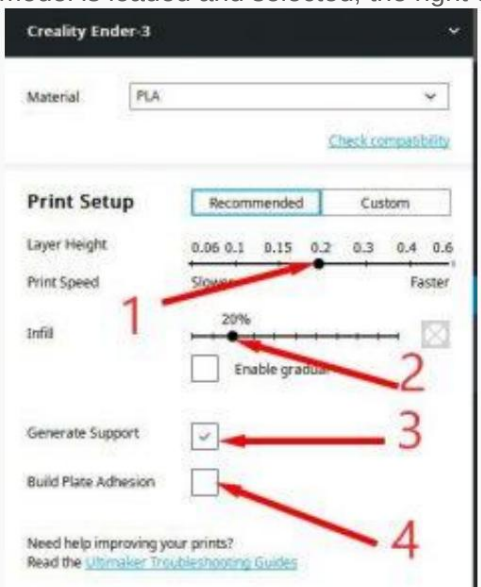
We will often need to change the position or other parameters of the model - namely the model, not its printing parameters. Cura offers several basic tools for this. To view them, select the model by clicking on it with the mouse (it can be made from any view). You will then have access to the buttons on the object manipulation panel (located on the left).

As follows from top to bottom:

- **Move** - move the model
- **Scale** - resize the model, as this option has two ticks that can be selected:
  - Snap Scaling - allows you to increase / decrease the size of equal parts;
  - Uniform Scaling - allows the model to be changed on all three proportional measurements;
- **Rotate** - rotation of the model on the three axes. You will often need this option to position the model in such a way that it does not come off or need support;
- **Mirror** - mirror reflection of the model relative to the selected plane;
- **Support Mesh** - designed to use individual models as supports (for now we will not use it because it is for advanced);
- **Support Blocker** - it is designed for configuration of supports (for now we will not use it because it is for advanced);

Printing parameters.

When the model is loaded and selected, the right one will

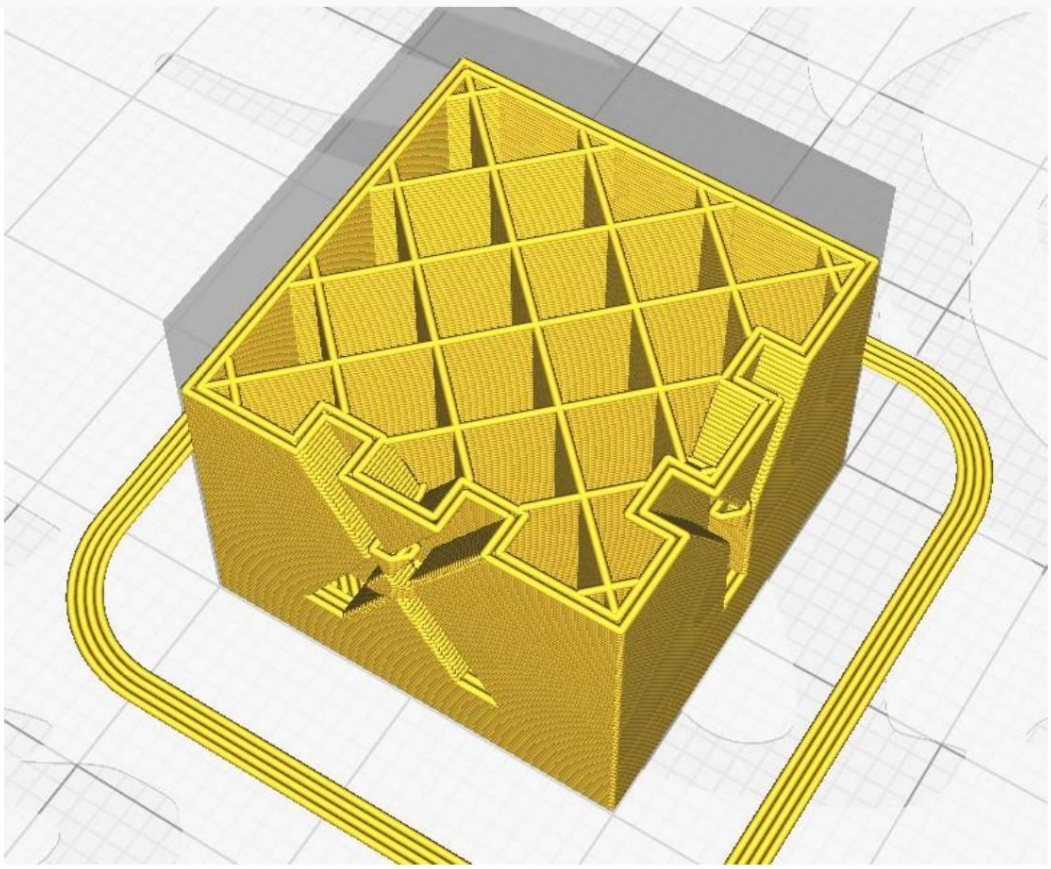


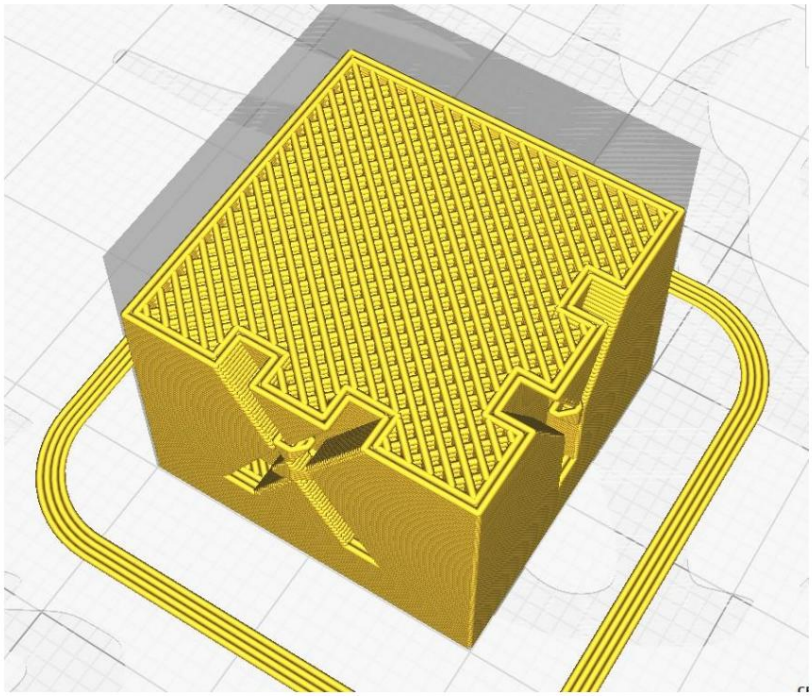
printer will build your model. Switch to **Layer View** settings responsible for how you find the

the changes.

1.Layer Height - layer height, responsible for how many layers the model will be built; try changing this setting by pressing prepare each time. You will notice that the detail of the model will change, but the printing time will also change. The lower the height of the layer, the slower but more detailed it will be printed.

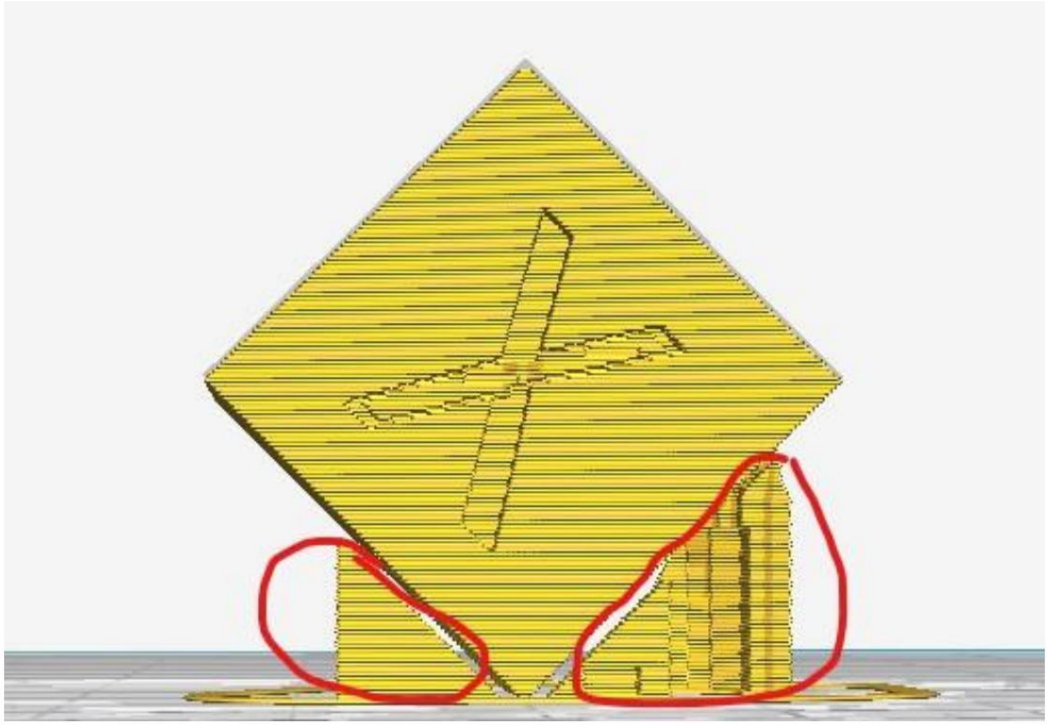
2. Infill - fill, determines what percentage will be filled model. By changing this option and pressing Prepare with the layer slider you can see how the fill in the model changes.

- 

- 

4. **Generate Support** - in this case with the cube model you will not notice changes if you select this option. But if you rotate the cube, you will notice that the program generates additional structures that support the model. This happens if the model or some part of it hangs in the air (the molten filament is always superimposed in layers and if there is no layer on which to lie, a support is made).

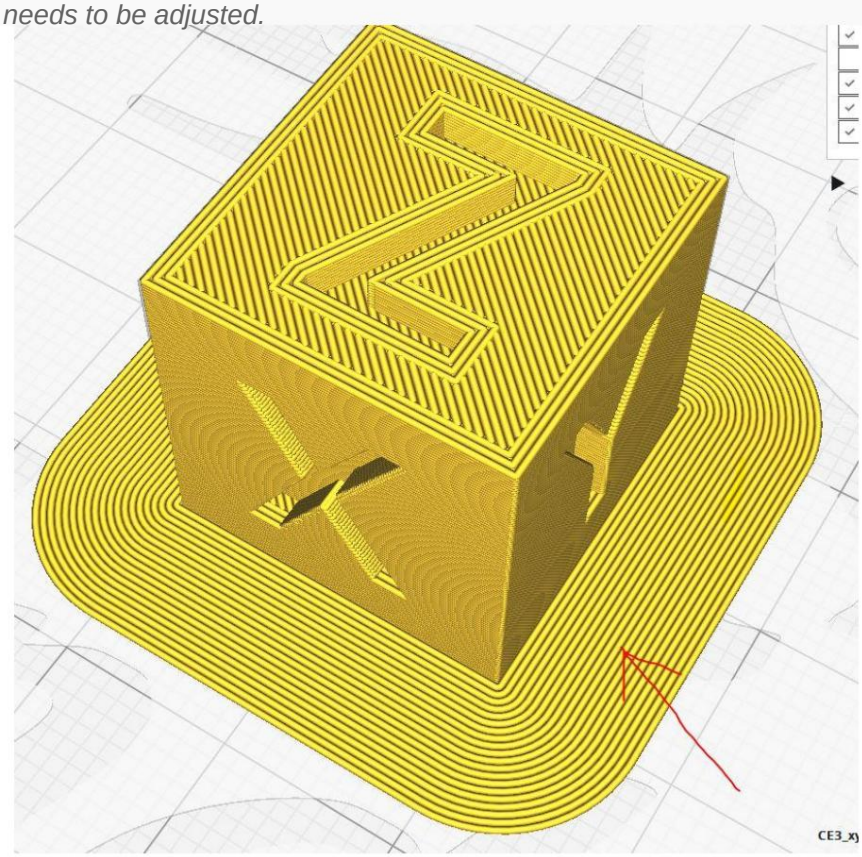
- 

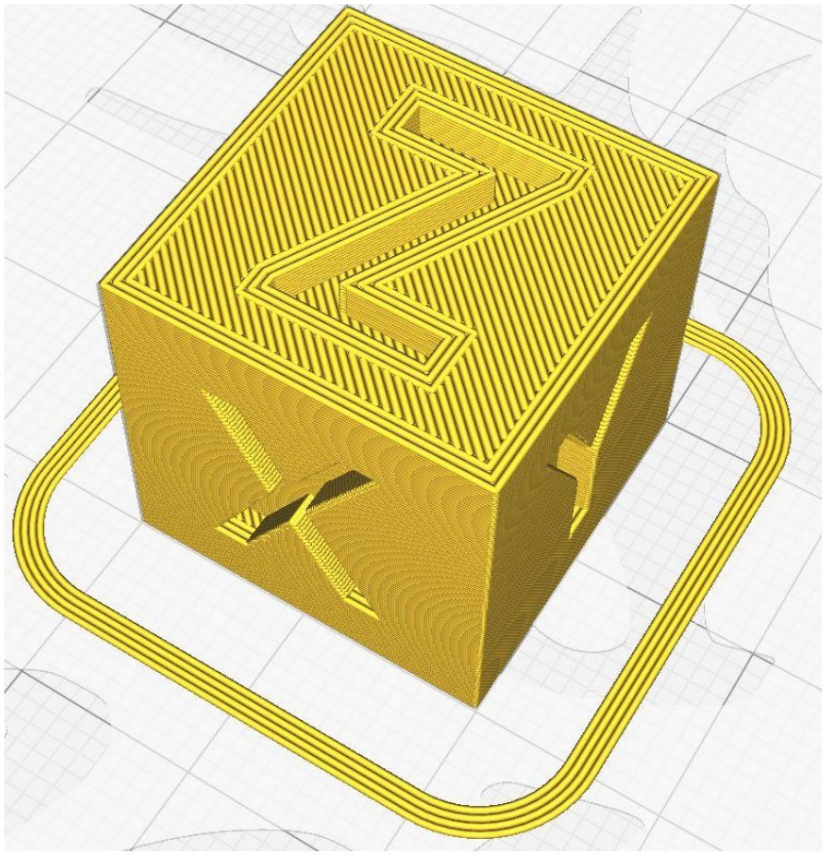
- 

** In the future, be careful and check if the model requires supports. In more detail: c*
the tutorial for advanced.

4. Build Plate adhesion - this option around the first layer of the model builds several additional lines. This is necessary to increase the area of the first layer. The larger the area, the better the adhesion to the base and the lower the risk of the model peeling off during printing. This option is extremely important, as it sometimes determines whether the model will be printed successfully at all.

** If this is the first print after calibration, Build Plate Adhesion is recommended*
enabled so you can see if the printer is calibrated correctly and what
still needs to be adjusted.

- 

- 

GEOGEBRA 3D Calculation + Augmented Reality

A lesson

Class: 11

Subject Title: Mathematics

Number of students: 26

Objectives: Visualization of an angle between crosses makes it difficult for students to get it present in space, and also because of the difficult definition of the concept of angle between crossed lines, spatial imagination, motivation, feeling successful.

Expected results: Professional demonstration of a mathematical model that comes to life in the classroom, enters and looks from the inside, inspires, brings satisfaction and a sense of being useful and leaves for use by others by publishing

Key issues: Challenge, creativity, ingenuity, application discovery, tools

Guiding questions:

What are you doing AH and MN?

What do we call the angle between crossed lines?

What additional construction is needed to find the place on the corner?

In which triangle is the required angle?

What is the shape of the triangle AHC?

Finding the measure of the required angle>

Glossary of new concepts: crossed lines, angle between crossed lines, parallel lines, lines lying in the bottom plane, an equilateral triangle

Assignment of a task:

A team of two students was given a math task: A cube ABCDEFGH was given. Point M is the midpoint of ABPoint N is the midpoint of BC. Find the angle between the lines MN and AH. To introduce myself a stereometric demonstration of determining the angle between these two lines and how we will find it.

Team sub-tasks (Discussion questions):

1. Download the application Geogebra 3D Calculation AR, registration in Geogebra 3D in order to publish the product
2. Use the Chrombook Tab 10 and account for demonstration purposes.
3. Recall the equation of rights in 2 points, homothety, work with parameters to have animation (can be manual or automatic animation)
4. Recall the definition of the angle between crossed lines and the algorithm for its construction.
5. Link to the product https://www.geogebra.org/classic/dt55t8jf
   we send video format mp4: http://bit.ly/2ORRVxx

In the process of work:

1. One student started working on a desktop computer using the account for

work in the teacher's G suite, an account in GEOGEBRA 3D and save the task in two ways, one of which is sharing. He moves extremely professionally
inverted on all sides of the model and it is seen that the lines do not lie in one plane, so they are crossed, dynamically (with animation) are made the necessary constructions to determine the location of the desired angle. Additional online recording on the screen with apowersoft in mp4 video format

2. **The second student works with the tablet, opening the already created model in Geogebra3D already with augmented reality using the AR button and demonstrating to students. They are in**
delight as heard in the video.

Conclusion:

I demonstrated to my colleagues in mathematics, who exclaimed in amazement at the open opportunity to continue to make models together with students, so they experience knowledge and it becomes lasting, they also present hypotheses based on this demonstration and then prove. The classroom lives with the model among ourselves. Forms critical thinking and design thinking.

Ideas:

We ask teams to develop tasks on this method with augmented reality as the product (Video, animation with AR we provide for study by students instead of traditional homework), the next hour we comment, ie students learn new knowledge at home and at school the next hour we are commenting, practicing, something like an inverted classroom model to overlap the globe, then we will move the model and we will have a task to calculate the length of the equator and radius. The other task with a cube for the pool and the number of tiles to cover the walls and floor. And the other tasks in Jamboard and scratch.

Method 1: Blender

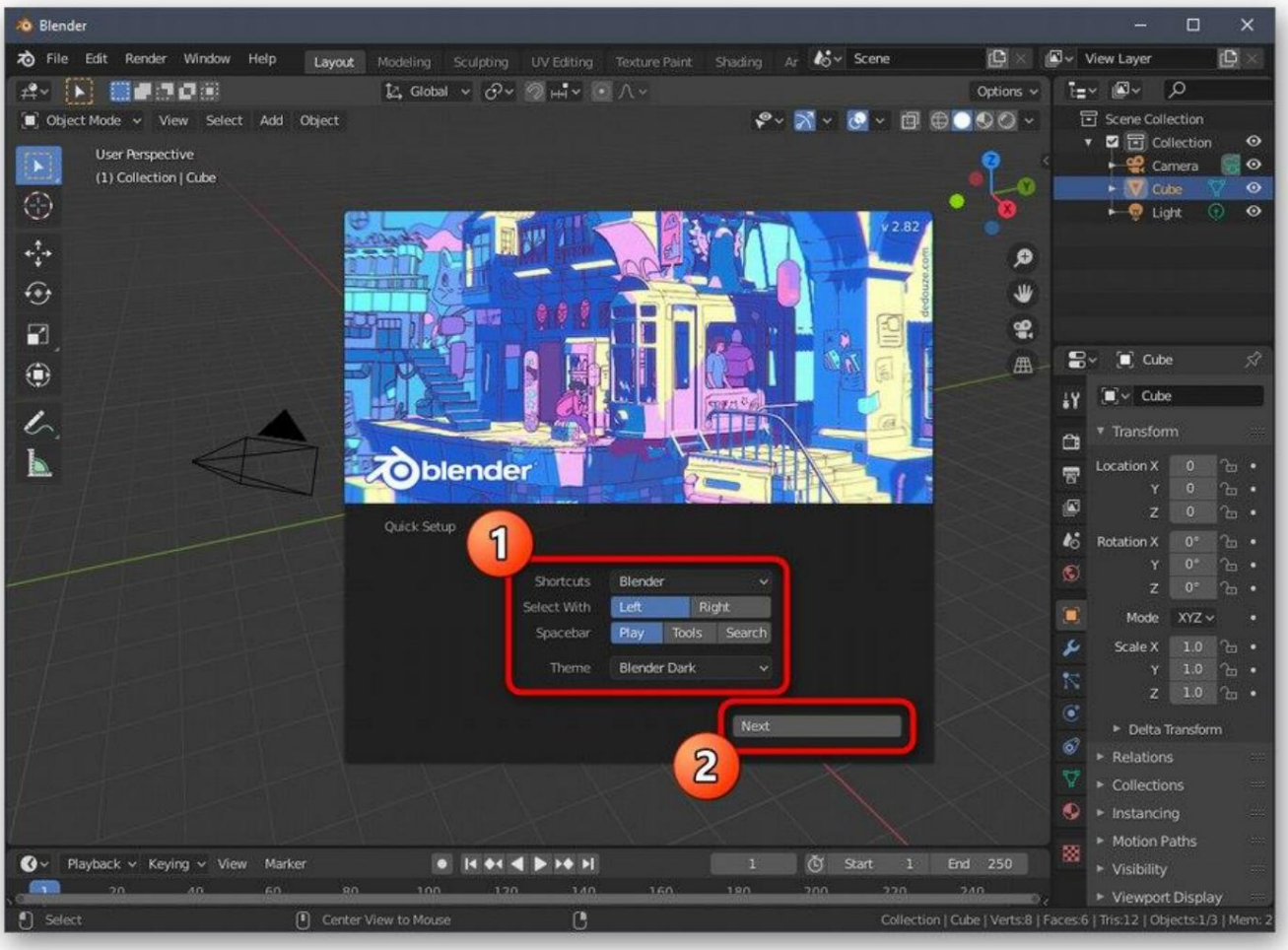
Blender is the first program whose main purpose is to create 3D models for their further animation or application in various fields of computer technology. It is distributed free of charge and is suitable for novice users who have encountered applications of this type for the first time, and therefore occupies this position. Let's take a quick look at the procedure for preparing a model for printing step by step, starting with the setup of the tool itself.

Download Blender

Step 1: preparatory steps

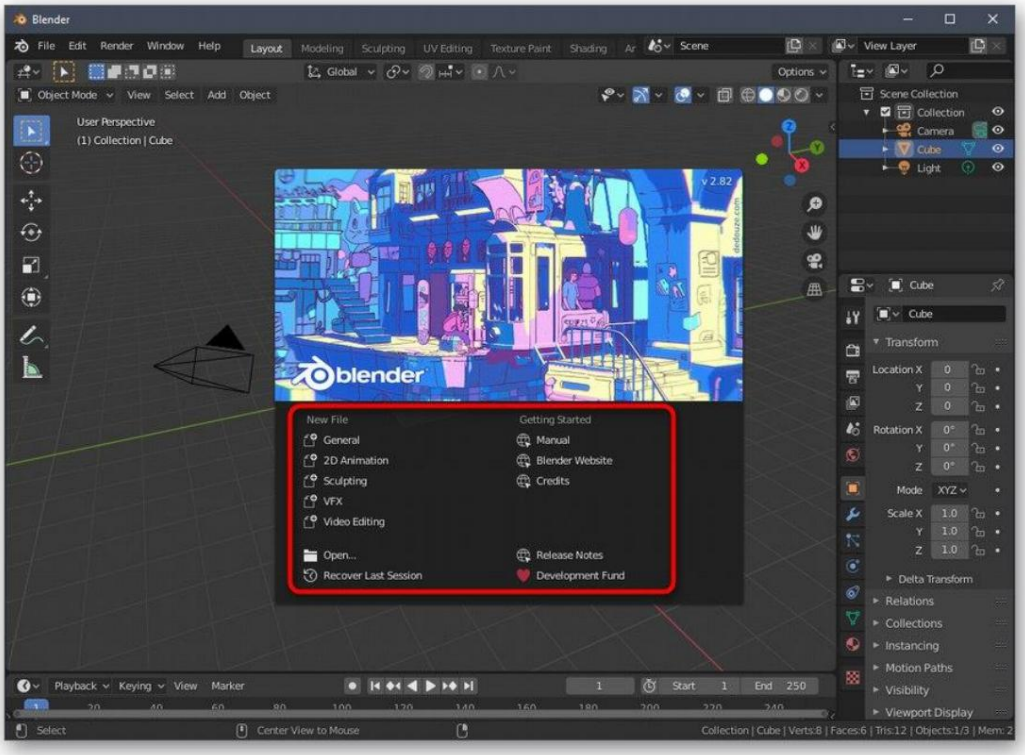
Of course, once you start Blender, you can immediately get acquainted with the interface and develop models, but first it is better to pay attention to the preparatory steps to set up the working environment for layouts for 3D printers. This operation does not take much time and requires only a few parameters to be activated.

1. As a start in the home window, select the parameters of the appearance and arrangement of the elements, starting with your personal needs.

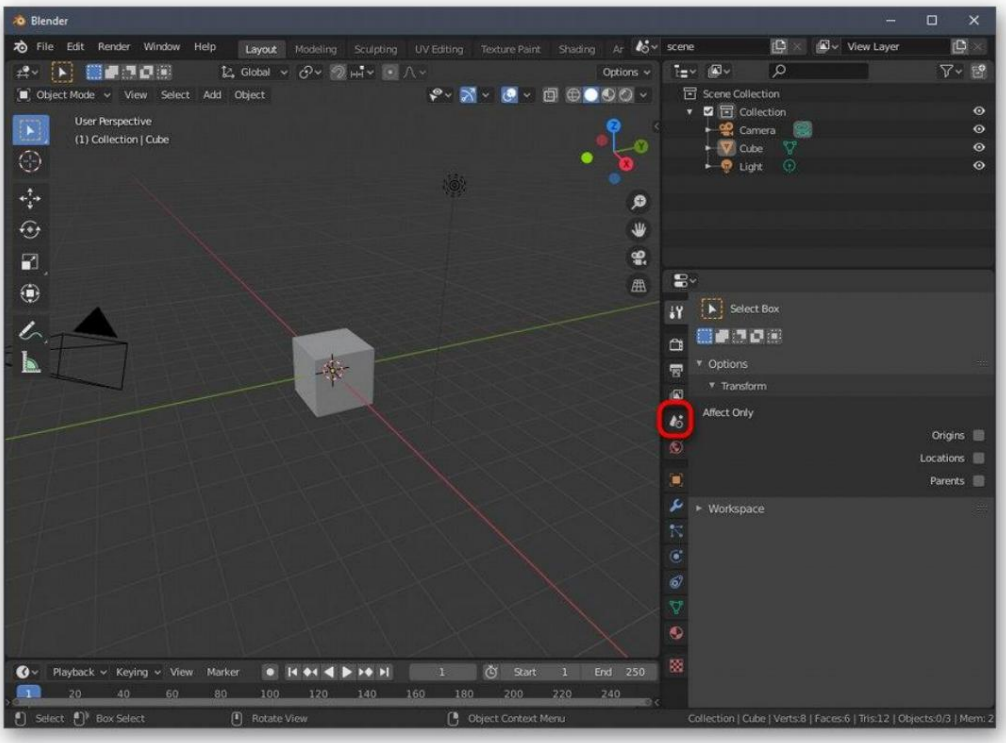


2. In the next section of the *Quick Setup* window you will see different templates to get started and links to resources with basic information that will be useful to you

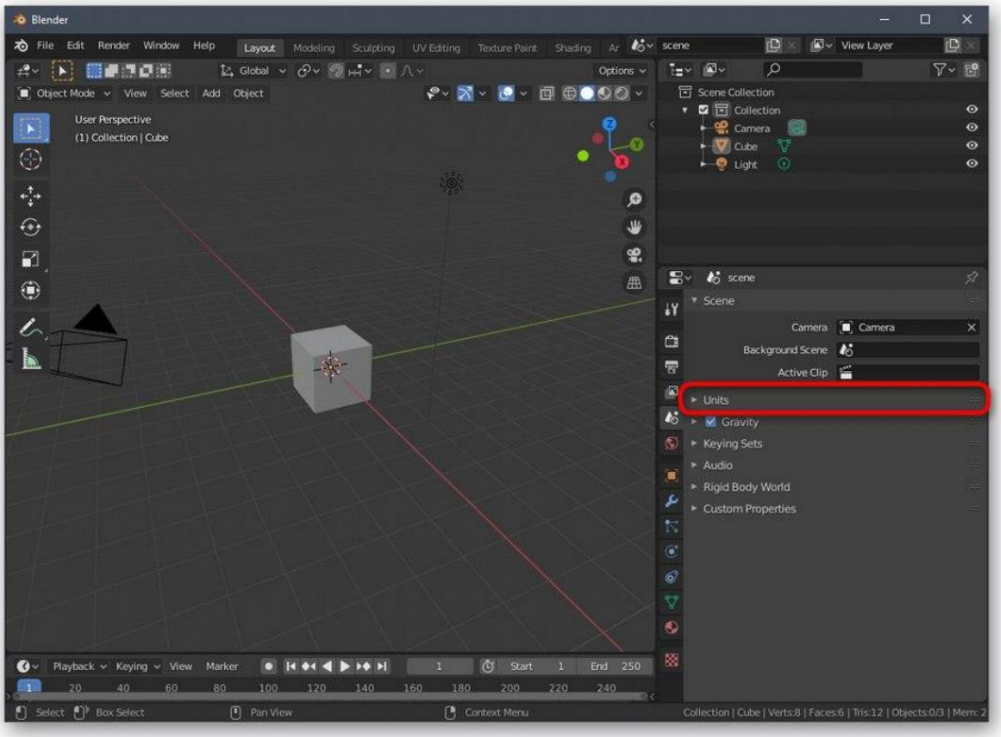
when mastering the software. Close this window to proceed to the next configuration step.



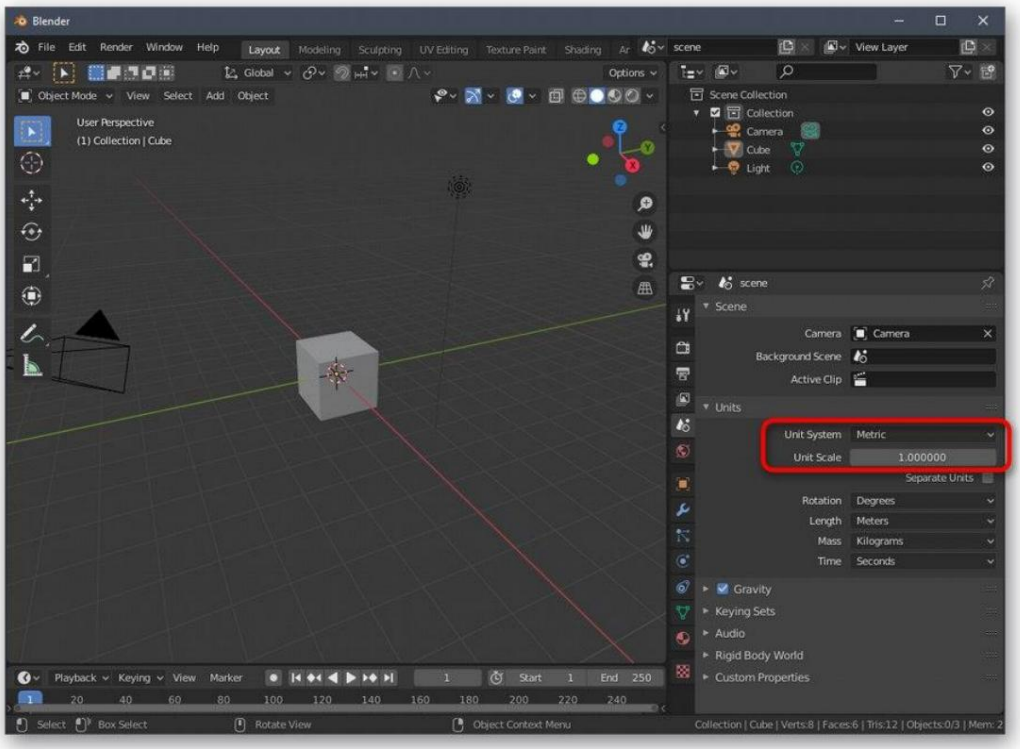
3. In the right pane, find *the Scene* icon and click on it. The name of the button appears a few seconds after holding the mouse cursor over it.



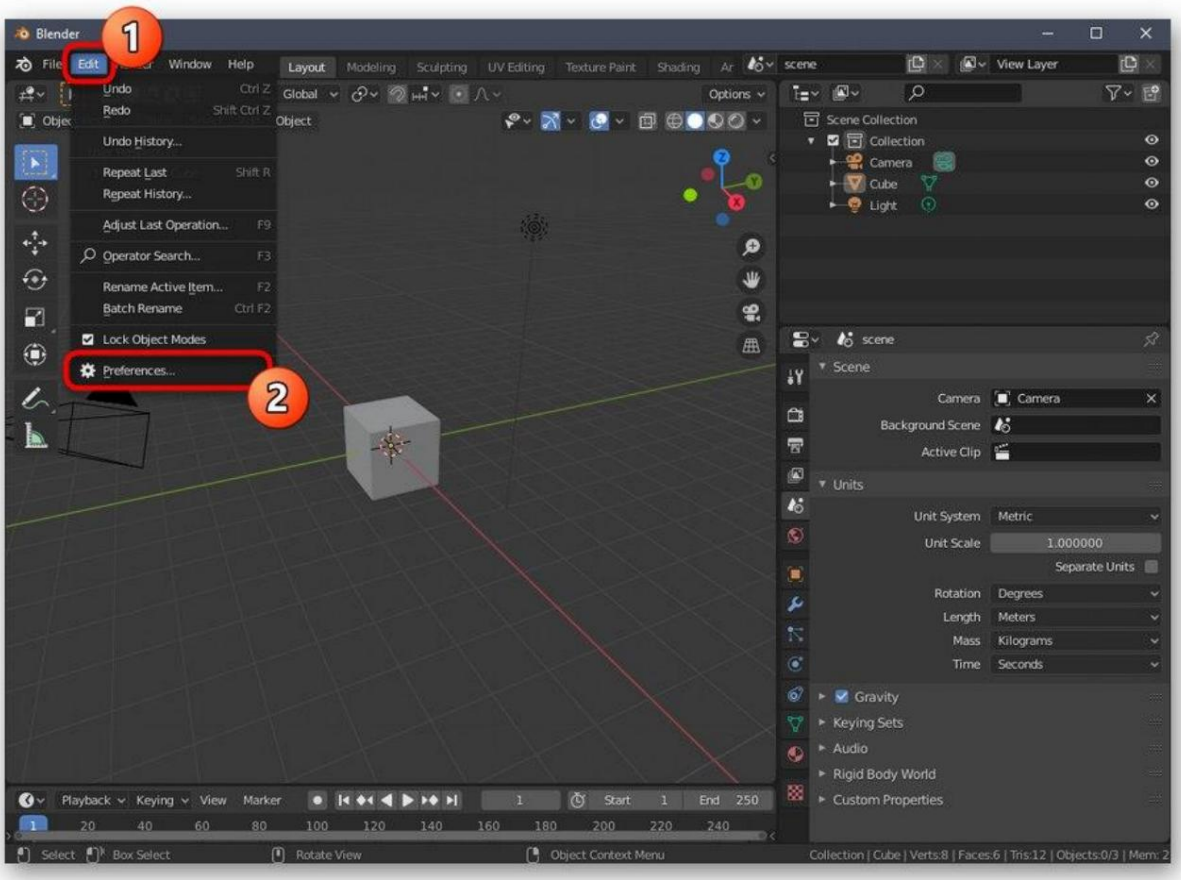
4. In the category that appears, expand the *Units block.*



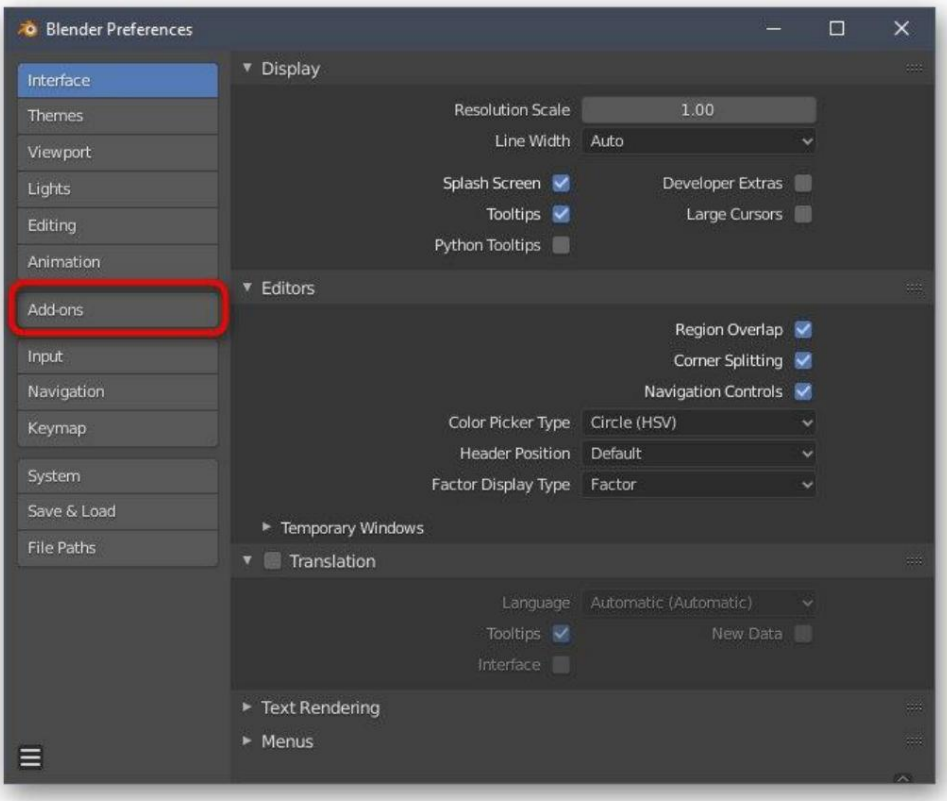
5. Set the metric system of measurements and set the scale "1" ... This is necessary so that the scene parameters can be transferred to the space of the 3D printer in the correct form.



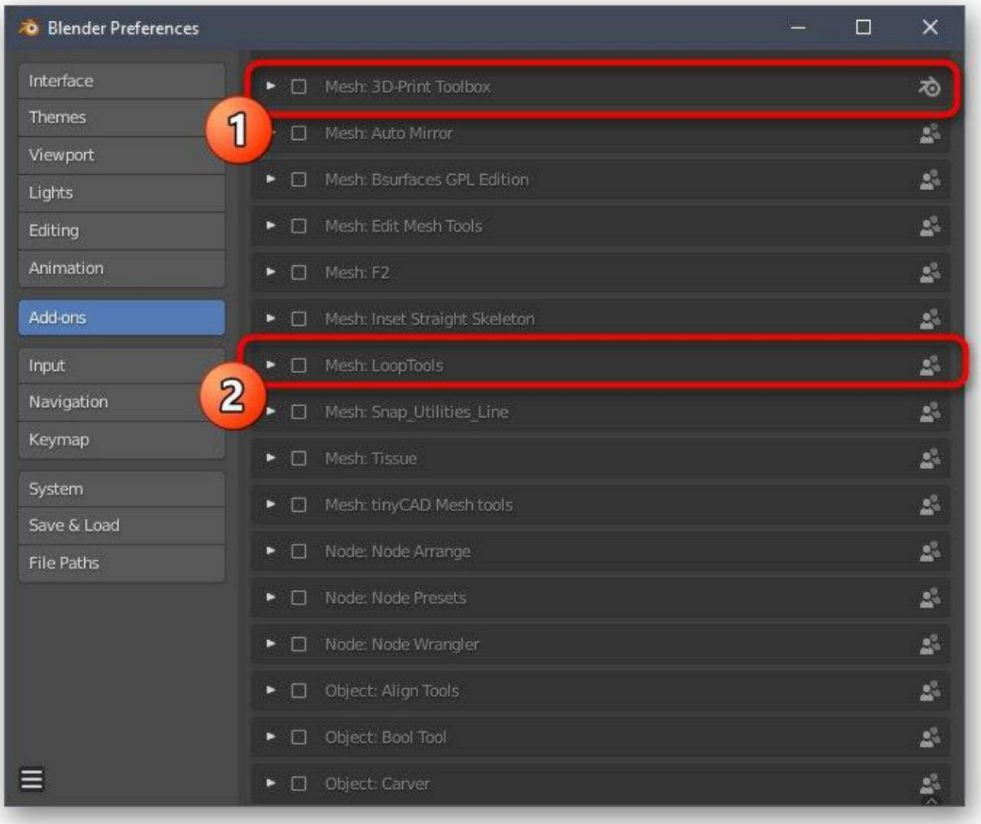
6. Now pay attention to the top bar of the program. Hold the cursor there *the "Edit"* mouse and select *"Preferences"* from the pop-up menu that appears .



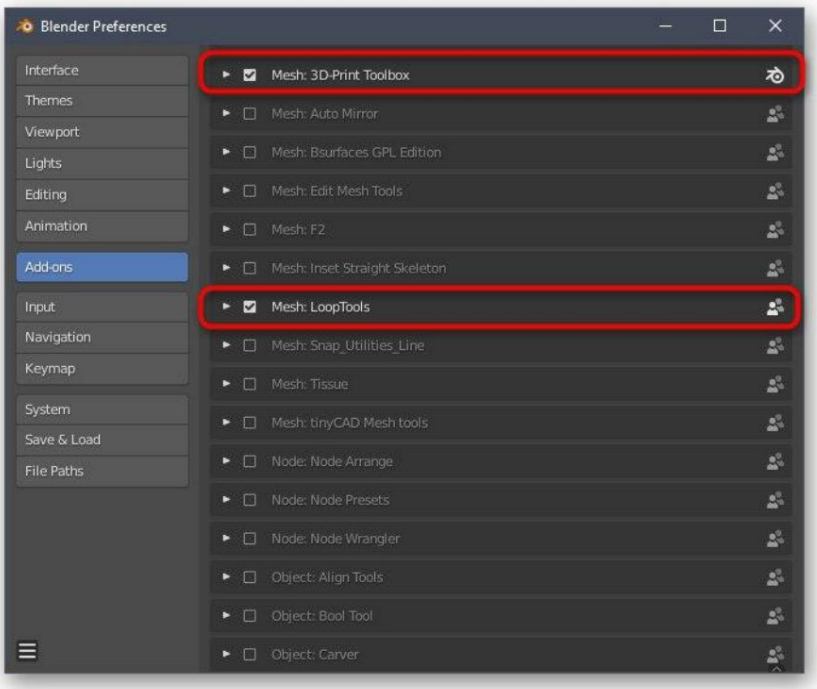
7. In the settings window, open *Add-ons*.



8. Find and activate two items under the names *"Mesh: 3D-Print Toolbox"* and *Network: LoopTools*.



9. Make sure the boxes are checked and then leave this one window.



We also recommend that you pay attention to other elements of the configuration. Here you can customize the appearance of the program, change the order of the interface elements, transform them or disable them altogether. After completing all these steps, proceed to the next step.

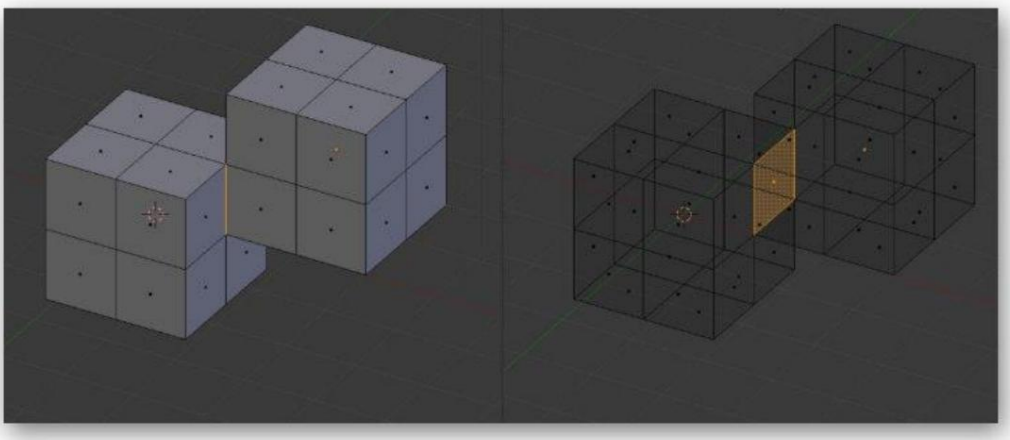
Step 2: Create a 3D object

Modeling isthe main process of creating a project for further printing onthe relevant equipment. This topicwillhave to beunderstood byevery userwho wants to work independently on different shapes and objects. To dothis, however, youwillneed to study quite a large layer of information as well
Blender's functionality isso vast that youcanintuitively understand only the basics. Unfortunately, the format of today's article will not allow you to fit even a small part of all the information and instructions, so we advise you to refer to the official documentation, where all the information is divided into categories and described in detail. To do this, just click on the following link.

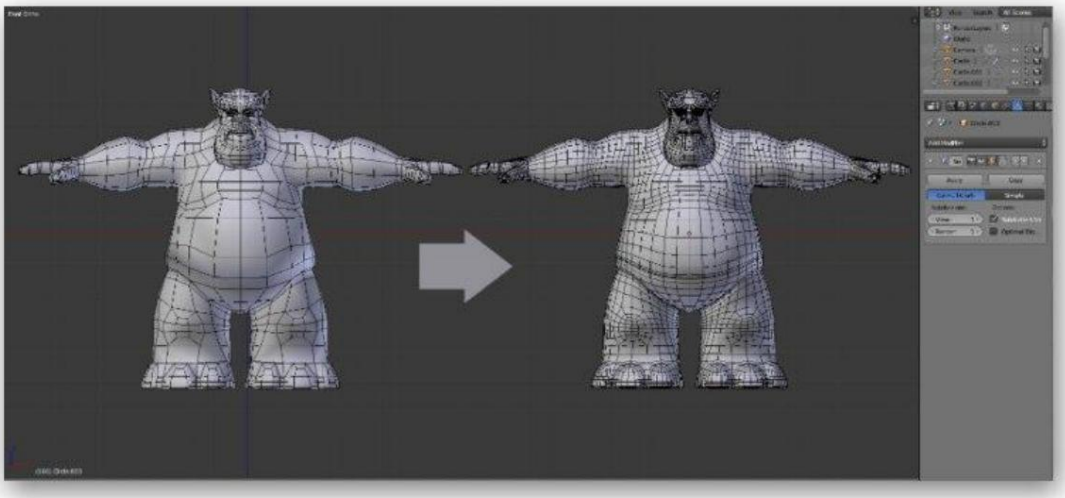
Go to the official Blender documentation

Step 3: Verify the project for compliance with the general guidelines

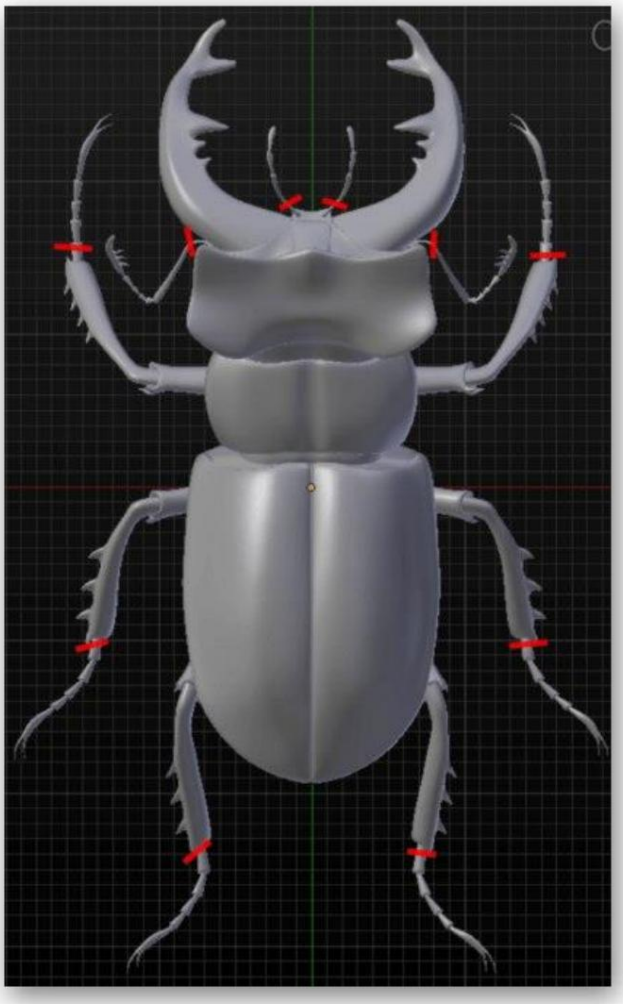
Before completing the model, we advise you not to neglect the most important aspects that need to be implemented in order to optimize the project and make sure that it is printed correctly on the printer. First, make sure that none of the surfaces overlap. They only need to touch, forming an object. If something exceeds the frame, there will probably be problems with the quality of the form itself, as a small printing error will occur in the wrong place. For convenience, you can always turn on the transparent grid display to check each row and field.



Then start reducing the number of polygons, as a large number of these elements only artificially complicate the shape itself and hinder optimization. Of course, it is advisable to avoid unnecessary polygons even when creating the site itself, but it is not always possible to do this at the current stage. All ways of this optimization are available to you, which is also recorded in the documentation and described in training materials by independent users.



Now we want to mark thin lines or some transitions. As you know, the nozzle itself has a certain size, which also depends on the model of the printer, and plastic is not the most reliable material. Therefore, it is better to avoid the presence of very thin elements, which in theory may not work on the print at all or will be extremely fragile. If such moments are present in the project, slightly increase them, add support or, if possible, get rid of them.



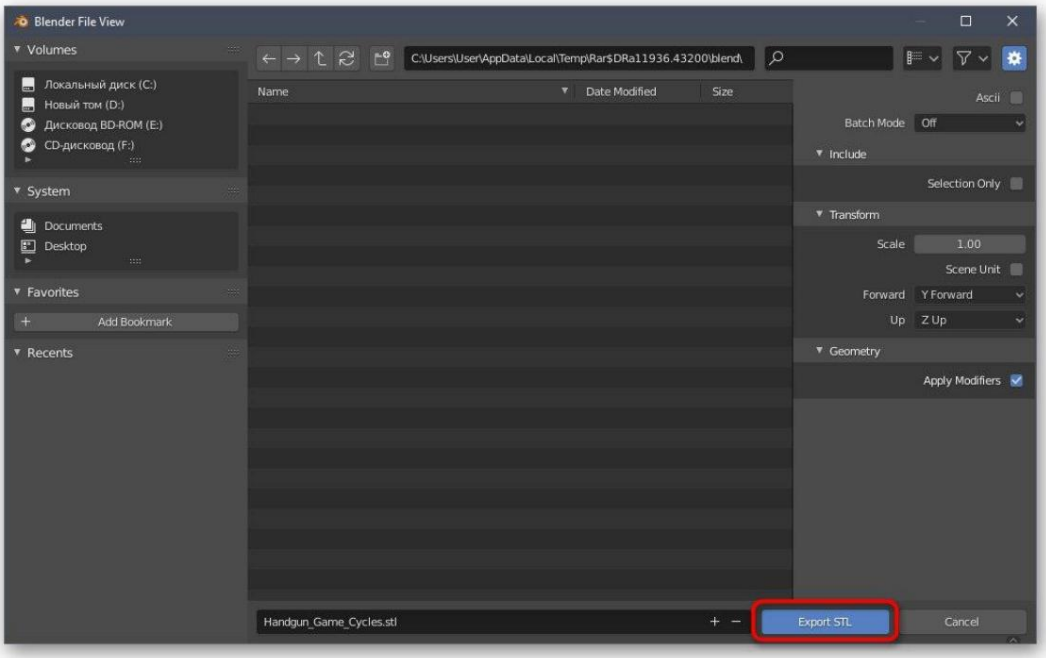
Step 4: Export the project

The last stage of preparation of the model for printing is its export in a suitable STL format. This type of data is supported by 3D printers and will be recognized correctly. It is not necessary to do any rendering or additional processing if colors or some simple textures are already set in the project.

1. Open *the File* menu and hold down the *Export* mouse cursor .

2. In the pop-up list that appears, select *"Stl (.stl)".*

3. Specify the location of removable or local media, specify a model name, and click *Export STL.*



The project will be immediately saved and made available for other actions. You can now insert a USB stick into the printer or connect it to your computer to begin the existing task. We will not give advice on setting it up, as they are purely individual for each device model and are clearly stated in the instructions and various documentation.

Chapter 4. INTEGRATION OF ROBOTICS, 3D PRINTING AND AR USING MOBILE DEVICES

FabLab SchoolNet Contest. Game Rules

Treasure Hunt & Charade

1. THEME

The theme will aim a treasure hunt + Charade in the representative touristic ancient cities of the participating countries. The starting point should be Italy (Palermo), Grecia (Trikala), Bulgaria (Varna), Romania (Galati), and Lithuania (Siauliai). The objects to be found will be a book/papirus that explain the design of greek/roman theatres. (to be discussed).

2. TEAMS AND TEAMS' MISSION

Each team is in charge of one robot to follow the robot mission (see below). Each team is composed of two sub-teams, sub-team "A" and sub-team "B". The "A" and "B" sub-teams can communicate with each other just using AR contents and markers.
From the beginning of the game each "A" team is informed about the starting point of each relative "B" team and about each path to be followed and which object on the board the "B" team has to move.
Each "A" team must prepare the AR contents about all this information, linking the content to the AR markers that the "B" teams have to read to play the game.

3. ROBOT MISSION

Robot mission is to move around the play-board taking into account the standard sensors on EV3 and/or mBot robots.

a. Robot construction

The robot could include the following line sensor, obstacle detection sensor (ultrasonic), lateral deflection sensor (Giro sensor), optical or display signaling, acoustic alarms, or vocal commands using / sound sensor / microphone. In addition, a color sensor could be added for different other indications.

b. Robot Programming

The robot programming should be made using the standard platforms of the robots used.

c. AR development (programming and markers recognition)

The AR programming should be made using the application developed by CNR (Here, we can include information about how to download it and install.)

4. BOARD

The board will be constructed like a map. The instructions for the teams should be in the form of markers, placed on / under objects on the board.

5. OBJECTS ON THE BOARD

The objects on board should be 3D modelled and printed. The objects will represent a special heritage monument that is representative for each city.

6. SCORING

No.	Activities	Scoring
1.	Creating the 3D model of the monument	- 50 points
2.	Printing the 3D model of the monument	- 50 points
3.	Construction of the robot	- 50 points
4.	Reading/Associating the AR marker	- 50 points
5.	Placing the object on the right target box (T)	- 50 points
6.	Recognising the monument	- 20 points
7.	Robot completely stops within Start/Finish Area (S)	- 10 points

Activity points can be multiplied each time when an activity is repeated.

Penalty points

No.	Activities	Scoring
1.	Re-programming sessions	- 5 points
2.	Collisions on forbidden areas	- 5 points

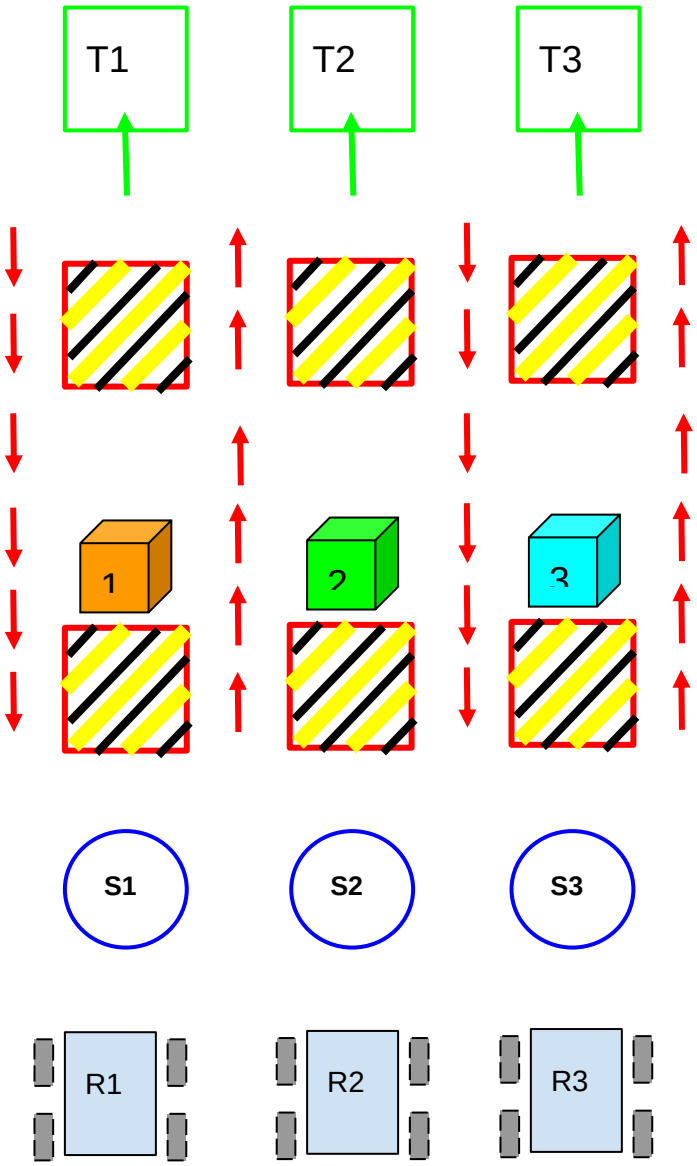
The penalty points can be multiplied each time when an activity is repeated.

The winner will be the team that reaches the highest score in the shortest time.

(In the case of the same score, the second winning criteria is the time**).**

7. SAMPLE PRACTICE GAME

Sample Board:



8. GAME DESCRIPTION

The playing field is like the one shown in the figure above: there are 3 identical Robots available (R1, R2, R3), three starting points (S1, S2, S3), the forbidden parts which the robots must avoid (the black and yellow areas), the preferential directions in the routes (one way, red arrows), 3 objects (1, 2, 3) that will be moved and brought to the destinations areas (T1, T2, T3).

As well as the playing field, the objects to be brought to the destination (3D printed stylized representations of local monuments) and many markers, which will be used to communicate the instructions, must be prepared in advance.

At the beginning of the game, each team is arbitrarily assigned a robot (initially disassembled) but nothing is known about the "starting points", nor about the objects etc.
Each team is divided into sub-team A and sub-team B.
All sub-teams can openly collaborate only when assembling robots. After completing the assembly, sub-teams A and B will only be able to communicate via augmented reality content.

The judges (teachers or else) will deliver closed envelopes to sub-teams A. Inside the envelopes there are markers and a description of the information to be linked to each marker. All markers are different and have been printed in duplicate.

Each sub-team A places a copy of each marker at the points on the playing field where they will be "found" by sub-team B to proceed with the treasure hunt.

The first marker will be connected to the instruction relating to the starting box of each robot (e.g. Robot 2 must start from box S3). The sub-team A2, therefore, will have to create an AR content (textual or visual), to communicate to the sub-team B2 to place the robot R2 in the starting box S3.

The B2 sub-team must display the AR information through the appropriate application to understand in which starting box to place the R3 robot.
In the starting box S3, the sub-team B2 will find the copy of the second marker. Sub-team A2 must create AR 3D content to show sub-team B2 which object they will have to place in their arrival box, for example object 2.

Sub-team B2 will have to read the AR 3D content which will be made up of the 3D file corresponding to object 4, then it will have to program the robot R3 to make it autonomously touch object 2, respecting the "one-way" and forbidden areas (the use or not of the sensors depends on the level of difficulty you want to attribute to the game. The basic version can be without sensors.)

In correspondence with object 2, sub-team B2 will find a copy of the third marker. Sub-team A2 must create AR content (textual or visual) to describe to sub-team B2 what is the next point to reach and how to reach it (with or without sensors). In the simplest version, proceed directly to the arrival box; we assume that for team 2 it is box T1.

(the number of intermediate points and the use of sensors will make the game more or less long and complicated).

The treasure is won if the object is placed in the arrival box with the robot.

For example, team 2 will win the treasure if it manages to bring object 2, through Robot R2, to the arrival box T1, beginning from the starting box S3.

The score of each team is calculated based on the parameters provided in the “Scoring” section.