


---

---

---

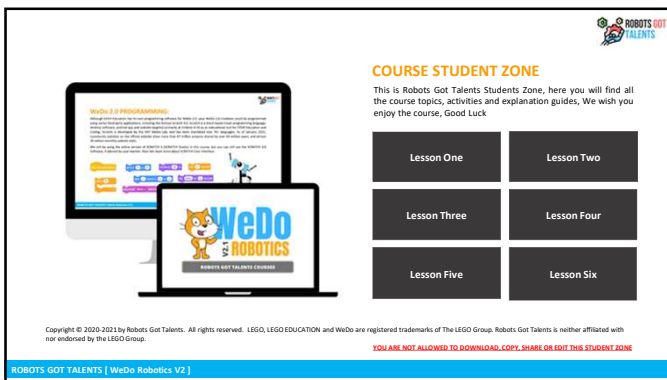
---

---

---

---

---




---

---

---

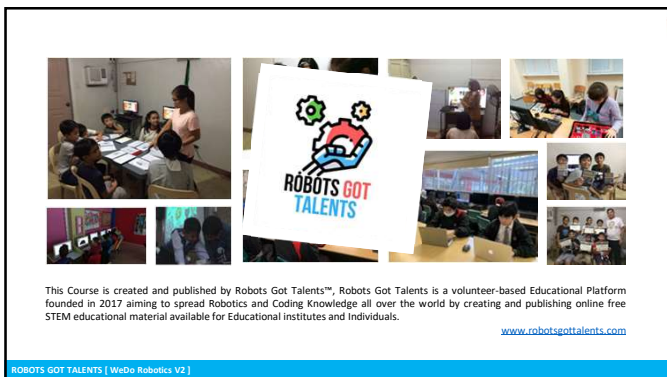
---

---

---

---

---




---

---

---

---


---


---


---

---

**THIS COURSE IS AVAILABLE ON:**







---

---

---


---

---

---

---


---



**COURSE INTRODUCTION:**

WeDo Robotics V2.1 is a Robots Got Talents™ classroom course for elementary school students. Throughout this course participants will learn the basics of robotics and coding as they design, build and program their own robots to finish different missions, tasks and exercises using LEGO Education WeDo 2.0 Educational Robotics Platform and SCRATCH 3.0.

WeDo Robotics Classroom Course V2.1 consists of 6 lessons that cover all the topics mentioned below in addition to 11 building and programming exercises:



• Robots Characteristics	• WeDo 2.0 Output Blocks
• Uses of Robots	• Flowchart
• WeDo 2.0 Introduction	• History of Robotics
• WeDo 2.0 Main Parts	• WeDo 2.0 Flow Blocks
• WeDo 2.0 Building Pieces	• Awesome Robots (ASIMO)
• Introduction to Algorithms	• Pseudocode
• Basics of Programming	• WeDo 2.0 Sensor Blocks
• WeDo 2.0 Programming	• Awesome Robots (Opportunity)
• WeDo 2.0 Software	• LEGO DIGITAL DESIGNER
• WeDo 2.0 Motor Blocks	• LDD Modes & Controls
• WeDo 2.0 Input Blocks	• LDD Exporting & Importing Models
• Types of Robots	• Awesome Robots (NAO)

ROBOTS GOT TALENTS | WeDo Robotics V2 |

---

---

---

---

---

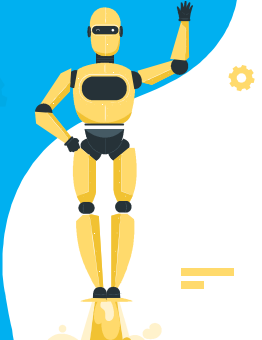
---

---

---

# LESSON ONE

- [Robots Characteristics](#) [ Book Page/s: 3-4 ]
- [Uses of Robots](#) [ Book Page/s: 4-5 ]
- [WeDo 2.0 Introduction](#) [ Book Page/s: 9 ]
- [WeDo 2.0 Main Parts](#) [ Book Page/s: 11 ]
- [Building Exercise One](#)
- [WeDo 2.0 Building Pieces](#) [ Book Page/s: 12-13 ]



---

---

---


---

---

---

---

---



**WHAT IS THE FIRST THING THAT COMES TO MIND WHEN YOU THINK OF A ROBOT?**

For many people it is a machine that imitates a human—like the androids in Star Wars, Terminator and Star Trek: The Next Generation. However much these robots capture our imagination, such robots still only inhabit Science Fiction. People still haven't been able to give a robot enough 'common sense' to reliably interact with a dynamic world. However, some people all over the world are working on creating such humanoid robots.

The type of robots that you will encounter most frequently are robots that do work that is too dangerous, boring, onerous, or repetitive. Most of the robots in the world are of this type. They can be found in auto, medical, manufacturing and space industries. In fact, there are over a million of these types of robots working for us today, but it is totally wrong to define Robots as machines that do our work or help us finish dangerous tasks, like many simple machines, could just do that for instance, Microwave heaters deals with harmful microwaves and they are not counted as Robots and the crane lifts heavy objects which a human could never deal with and they are defined as robots

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---

---

---

**A ROBOT HAS THESE ESSENTIAL CHARACTERISTICS:**

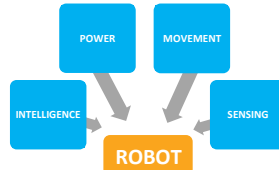
As strange as it might seem, there is no standard definition for a robot. However, there are some essential characteristics that a robot must have and this might help you to decide what is and what is not. It will also help you to decide what features you will need to build into a machine before it can count as a robot.

**SENSING:** Using the Sensors the robot should be able to sense its surroundings by one or more methods

**MOVEMENT:** Using Motors the robot should be able to move in its environment

**POWER:** Using the Power Source the robot should be able power itself

**INTELLIGENCE:** Using the Microcontroller (Robot's Brain) the robot should be able to take decisions according to its program



ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---

---

---


**ROBOTICS CHARACTERISTICS:**

**SENSING:** First of all a robot should be able to sense its surroundings It would do this in ways that are not similar to the way that we sense our surroundings, but robots need sensors to do that. Giving your robot sensors as light sensors (eyes), touch and pressure sensors (hands), chemical sensors (nose), hearing and sonar sensors (ears), and taste sensors (tongue) will give your robot awareness of its environment.

**MOVEMENT:** Moreover a robot needs to be able to move around its environment. Whether rolling on wheels, walking on legs or propelling by thrusters or even moving a claw. To count as a robot either the whole robot moves or just parts of the robot moves.

**POWER:** Also a robot needs to be able to power itself. It might be solar-powered, electrically-powered, or even battery-powered. The way your robot gets its energy will depend on what your robot needs to do.

**INTELLIGENCE:** Finally A robot needs some kind of Intelligence this is where programming enters the pictures, a programmer is a person who gives the robot its 'smarts.' The robot will have to have some way to receive the program so that it knows what it is to do.



ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---


---

---

---

---


---



**WeDo INTRODUCTION:**

The LEGO® EDUCATION WEDO robots may not be a type of robots that will go buy the breakfast every day or take care of your grandfather, but it could teach many things that will help you in your life from computational thinking to building LEGO® models. Since the creation of the MINDSTORMS Platform in 1998, LEGO Have tried to create an Educational Platform for Elementary School Students that would help them learn STEM in easy and fun ways.

The first Version of WEDO (9580) was released on 2009, it included two sensors, one motor and a USB Smart Hub. In 2016 WEDO 2.0 (45300) was released, with upgraded sensors, motors and Smart Hub.



ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---

---

---



ROBOTS GOT TALENTS

---

---

---

---

---

---

---

---

---

---



**LEGO EDUCATION WEDO 2.0**

The WEDO 2.0 Core Set includes 280 LEGO Pieces in addition to a two WEDO sensors (tilt sensor and motion sensor), one WEDO 2.0 Motor and a programmable Smart Hub that connects to a computer or tablet via Bluetooth. The Smart Hub is powered by either two AA batteries or a rechargeable battery pack. After recognizing the Set contents we can say that WEDO 2.0 is a robotic Set, as it includes all the 4 features of robots; Sensing, Movement, Intelligence and Energy.

			
INTELLIGENCE WEDO 2.0 Smart Hub	SENSING Motion Sensor	MOVEMENT WEDO 2.0 Motor	POWER WEDO 2.0 Battery

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

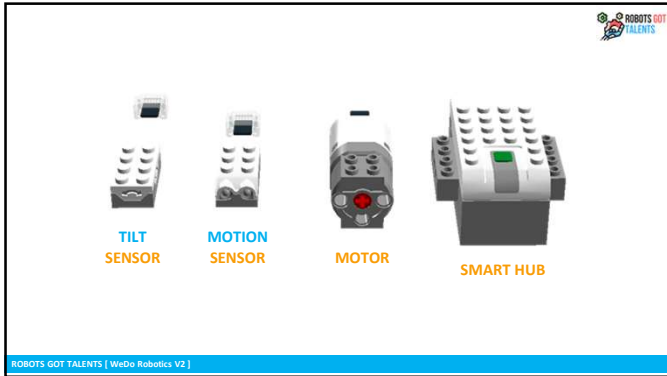
---

---

---

---

---




---

---

---

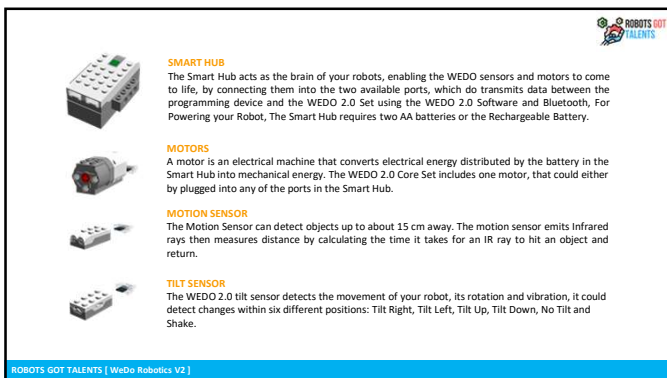
---

---

---

---

---




---

---

---


---

---

---

---

---





## BUILDING EXERCISE (1)

It is now time for your first building exercise, please press the button below to download the Building Instructions that you will use to build your first robot, for RoboApp users please long press the full screen button, to be able to access the downloadable files.

The following building instructions was created and provided by LEGO EDUCATION FOR the WEDO 2.0 Set:

Building Instructions Name: Fan  
Created by: LEGO EDUCATION  
Difficulty: Easy  
Number of Pieces: < 10

Download Instructions

---

---

---


---

---

---


---

---



### BUILDING PIECES

When you were Building your first robot (fan) you have noticed that the LEGO Building Pieces you used have different shapes and sizes. In addition to the Main Electronic Parts the WeDo 2.0 core set includes 280 System and Technic pieces, Which are classified to 6 groups, Bricks, Technic Bricks ,Beams, Connecting Pegs, Cross Axles and gears, each of these groups has its functionalities and shapes:



ROBOTS GOT TALENTS | WeDo Robotics V2 |

---

---

---


---


---

---


---

---






**BRICKS**  
Bricks are basic Building pieces of all LEGO Models. LEGO bricks are measured and identified by the number of the studs they have on top. The smaller number always comes first, so you say "a 2-by-4 brick," not "a 4-by-2 brick." The width of a 1x1 LEGO brick is the Fundamental LEGO Unit, or module (1 module or 1M is about 8 mm).



**TECHNIC BRICKS**  
Technic bricks are normal Lego Bricks with holes passing through them, in this holes you can insert pegs, cross axels to connect it with other Beams, Technic Bricks or other Building Pieces. Technic bricks are measured by the number of their studs. Except for the 1x1 Technic brick (A), there are no Technic bricks with an odd number of studs.



**BEAMS**  
Beams are basically used to hold cross-axes and other LEGO® components together, so they are the framework of most TECHNICAL and MINDSTORMS® models. There are 2 main types of beam: Straight and Angular. Straight beams are usually found only with circular holes in them while angular beams have a mixture of circular and cross-shaped holes, normally with the cross-shaped ones at either end. The straight beams come in sizes from 2 to 15M. These two groups can then be condensed down again into half-beams, beams and bricks.

ROBOTS GOT TALENTS | WeDo Robotics V2 |

---

---

---


---


---

---


---

---





**CONNECTOR PEGS**  
There are 3 types of connector pegs:  
Round Pegs, Cross-shaped Pegs, Half-half Pegs  
The round pegs can be used to connect beams together so that they can both swing freely. The second cross-shaped peg can be used to hold two beams together so that they cannot move, and the last peg can be used to connect a free-spinning beam to a fixed beam. There are also longer versions of the round pegs that can be used to connect multiple beams together. A longer version of the cross-shaped peg is an axle. Some pegs might also be different colours. The black and blue connectors are friction connector pegs, while the other pegs are smooth.



**CROSS AXLES**  
Cross-Axles are cross-shaped rods that can be used to hold wheels, gears beams etc. They are colour coded by their length or you can measure it using a counting the holes of a beam the same size, even-numbered lengths (2M,4M,6M etc) are black while odd-numbered lengths (3M,5M,7M etc) are grey. You can place a cross-axle through a circular hole so it can spin freely. This is a handy connection for building with gears and wheels.

ROBOTS GOT TALENTS | WeDo Robotics V2 |

---

---

---


---

---

---

---

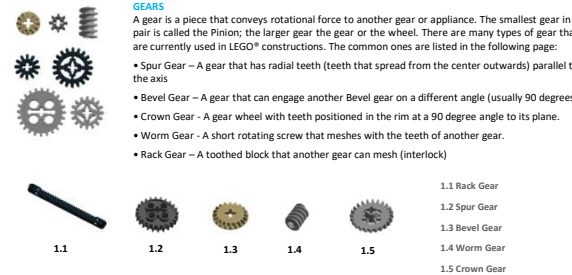
---



**GEARS**

A gear is a piece that conveys rotational force to another gear or appliance. The smallest gear in a pair is called the Pinion; the larger gear the gear or the wheel. There are many types of gear that are currently used in LEGO® constructions. The common ones are listed in the following page:

- Spur Gear – A gear that has radial teeth (teeth that spread from the center outwards) parallel to the axis
- Bevel Gear – A gear that can engage another Bevel gear on a different angle (usually 90 degrees)
- Crown Gear - A gear wheel with teeth positioned in the rim at a 90 degree angle to its plane.
- Worm Gear - A short rotating screw that meshes with the teeth of another gear.
- Rack Gear – A toothed block that another gear can mesh (interlock)



1.1 Rack Gear  
1.2 Spur Gear  
1.3 Bevel Gear  
1.4 Worm Gear  
1.5 Crown Gear

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

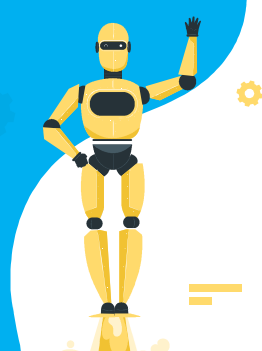
---

---

---

# LESSON TWO

- Introduction to Algorithms [ Book Page/s: 34 ]
- Programming [ Book Page/s: 19 ]
- WeDo 2.0 Programming [ Book Page/s: 31 ]
- SCRATCH Studio UI [ Book Page/s: 32-33 ]
- Programming Blocks [ Book Page/s: 34 ]
- Exercise One




---

---

---

---

---


---

---

---

---

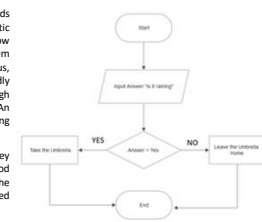
---



**ALGORITHMS:**

Of course, a robot won't be a robot without a program or in other words code and that's what would give your robot the forth robot characteristic intelligence, and that is what will be discussed now but we should know what is meant by an Algorithm. The word "algorithm" may not seem relevant to you, but the truth is that algorithms are all around us, governing everything from the technology they use to the worldly decisions they make every day. Algorithms are fascinating and, although some are quite complex, the concept itself is actually quite simple. An algorithm is a detailed step-by-step instruction set or formula for solving a problem or completing a task.

Algorithms are not just related to Programming or Computer Science they are everywhere. A recipe for making food is an algorithm, the method you use to solve addition or long division problems is an algorithm, the process of folding a shirt. Even your morning routine could be considered an algorithm.



ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

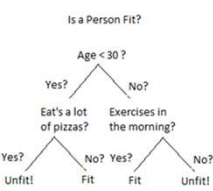
---

---

---

---

---




Is a Person Fit?

```

    graph TD
      A[Age < 30?] -- Yes? --> B[Eat's a lot of pizzas?]
      A -- No? --> C[Exercises in the morning?]
      B -- Yes? --> D[Unfit!]
      B -- No? --> E[Fit]
      C -- Yes? --> F[Fit]
      C -- No? --> G[Unfit!]
  
```

If a programmer wrote an algorithm for a robot to do a certain mission, then he gave it to another person to work on the code, if that algorithm isn't written in a way that anyone could understand, other programmers won't be able to work on the code, and That's why Global Algorithm Presenting Methods such as Pseudocode and FlowChart were made, these methods of writing algorithms follow a set of rules that all the programmers use and understand and that is what we will see now.



ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

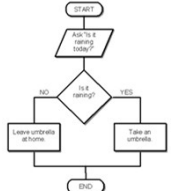
---

---

---

### PRESENTING ALGORITHMS:

**FLOWCHART**



**PSEUDOCODE**

```

1. PRINT "Please enter the length of the side in cm "
2. INPUT Side
3. Area = Side * Side
4. PRINT Area " cm2"
  
```

```

1. PRINT "Please enter the length of the side in cm "
2. INPUT Side
3. Area = Side * Side
4. PRINT Area " cm2"
  
```

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---


---

---

---

---

---



### PROGRAMMING:

Programming is a process that leads from an original formulation of a computing problem to executable computer programs. Programming involves activities such as analysis, developing understanding, generating algorithms, verification of requirements of algorithms including their correctness and resources consumption, and implementation (commonly referred to as coding) of algorithms in a target programming language. Source code is written in one or more programming languages. The purpose of programming is to find a sequence of instructions that will automate performing a specific task or solving a given problem.

**Programming Language:** In order for you to communicate with a computer (and to get it to execute your instructions) you must speak its language. In programming, a language is made up of a vocabulary and set of grammatical rules where it gets a little tricky is that each language is based on its own unique syntax (grammatical structure) and semantics (meaning).

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

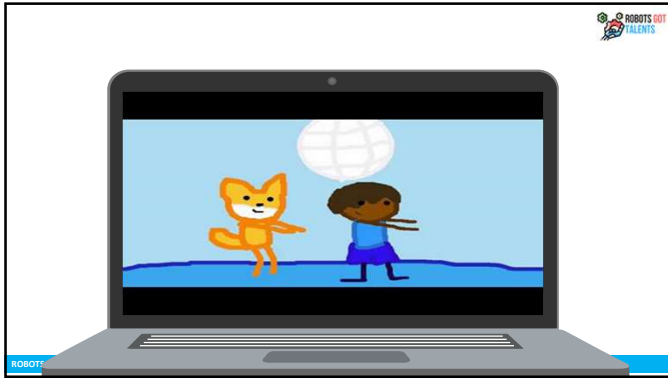
---

---

---

---






---

---

---

---

---

---


---

---

**WeDo 2.0 PROGRAMMING:**

Although LEGO Education has its own programming software for WeDo 2.0, your WeDo 2.0 creations could be programmed using various third-party applications, including the famous Scratch 3.0. Scratch is a block-based visual programming language, desktop software, android app and website targeted primarily at children 8-16 as an educational tool for STEM Education and Coding. Scratch is developed by the MIT Media Lab, and has been translated into 70+ languages. As of January 2021, community statistics on the official website show more than 67 million projects shared by over 64 million users, and almost 38 million monthly website visits.

We will be using the online version of SCRATCH 3 (SCRATCH Studio) in this course, but you can still use the SCRATCH 3.0 Software, if advised by your teacher. Now let's learn more about SCRATCH user interface:



ROBOTS GOT TALENTS | WeDo Robotics V2

---

---

---

---

---

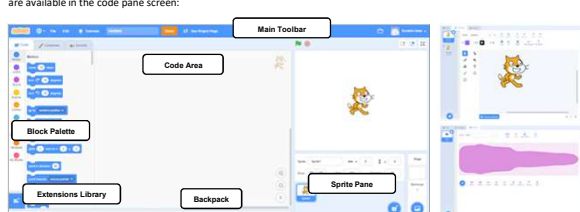
---

---

---

**SCRATCH STUDIO UI:**

Scratch 3.0 has a very simple user interface, which includes awesome features and capabilities. Scratch UI is made up of three main panes: the Code pane, the Sounds pane and the Costumes/Backdrop Pane, the following components covered below are available in the code pane screen:



ROBOTS GOT TALENTS | WeDo Robotics V2

---

---

---


---

---

---

---

---



### Scratch UI Components:

**Main Toolbar:** This is where you can find important project controls, like language, File, Edit, Tutorials, Project name in the image below, Share, Save, Revert, and New, User name, and open project folder are located.

**The Block Palette/Code Pane:** The area which includes all the Scratch Block, which are divided into 9 groups/ Categories; Motion, Looks, Sound, Events, Control, Sensing, Operators, Data, My Blocks in addition to the extensions. To use any drop just drag it from its group then drop it in the Code Area, You can either click press on any of the Category buttons or use the Scroll Bar, to find a block.

**Extensions Library:** To open the Extensions Library, press the blue button in the bottom of the Blocks Palette. There you can find 3<sup>rd</sup> party extensions, which add blocks to your blocks palette giving your projects extra features, the Extensions Library Includes Blocks for LEGO MINDSTORMS EV3, LEGO Education WeDo 2.0, LEGO Boost, Micro:bit and much more.

**The Code Area:** is the large empty area to the right of the Block Palette, where you drop the selected blocks from the Block Palette to form codes/scripts.

**Backpack:** This is an area where you save objects that you can use later in other projects. The objects can be costumes, sprites, backdrops, sounds, blocks, and codes. You can drag and drop these objects into the backpack and later drag and drop them from the backpack to reuse in other Scratch projects.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---


---

---

---

---

---





### Scratch UI Components:

**Sprite Pane:** This is the information panel located to the right of the Scripts Area and in the bottom of the stage there you will find a thumbnail for each sprite in your project. When selected, the thumbnail will appear highlighted and its details will show in the Sprite Header located above the list of sprites.

**The Sounds Pane:** The last tab in the top left area of the interface enables you to create, upload and manipulate sounds.

**The Costumes/Backdrop Pane:** The costume and backdrop panes can be accessed by clicking the middle tab in between the "code" tab and the "sounds" tab. This is where we can create and manipulate sprites and backdrops. To switch between costume and backdrop panes, select the desired thumbnail in the sprite info pane.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---


---

---

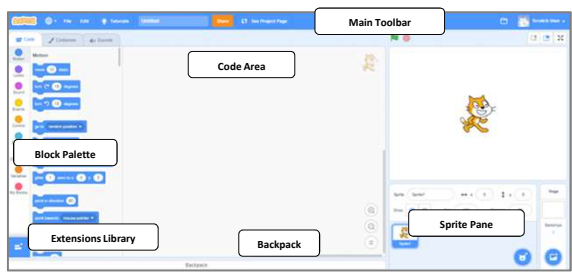
---

---

---



### SCRATCH STUDIO UI:



ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---


---

---

---

**PROGRAMMING BLOCKS:**

Blocks are puzzle-piece shapes that are used to create a code in Scratch. The blocks connected to each other vertically similar to a jigsaw puzzle, where each data type (hat, stack, reporter or Boolean) had its own shape and a specially shaped slot for it to be inserted into, which prevented syntax errors. Series of connected blocks were called scripts. There are 4 main types of blocks:



**Hat Blocks**      **Stack Blocks**      **Boolean Blocks**      **Reporter Blocks**

**types of blocks:**

**Hat Block:** A block that starts a code when a specific event occurs. All hat blocks are either Control blocks, Events blocks, or Blocks from the Extensions.

**Stack Block:** A block that is shaped to fit above and below other blocks. Stack blocks make up the majority of the blocks available in Scratch, being available in every group except Operators.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---


---

---

---

**PROGRAMMING BLOCKS:**

Blocks are puzzle-piece shapes that are used to create a code in Scratch. The blocks connected to each other vertically similar to a jigsaw puzzle, where each data type (hat, stack, reporter or Boolean) had its own shape and a specially shaped slot for it to be inserted into, which prevented syntax errors. Series of connected blocks were called scripts. There are 4 main types of blocks:



**Hat Blocks**      **Stack Blocks**      **Boolean Blocks**      **Reporter Blocks**

**types of blocks:**

**Reporter Block:** A block that reports a value to the device. Values could be anything from strings and numbers to sensors readings.

**Boolean Block:** A block that reports Boolean values. When the block is used, it acts as a reporter block, reporting "true" or "false" string values or the numbers "1" and "0".

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---


---

**EXERCISE ONE:**

It is now time for your first programming exercise, it won't be related to WeDo 2.0, but it is a really interesting activity with SCRATCH 3 which would help you understand the SCRATCH Blocks and user interface more.

Guided by an official SCRATCH tutorial you are going to build an basic jumping game.

Video Tutorial URL: <https://youtu.be/1j1tvXak1Iw>  
Channel Name: Scratch Team

 Now open SCRATCH Studio, start a new empty project and go to the next page to start the video tutorial

ROBOTS GOT TALENTS

---

---

---

---

---

---

---

---




---

---

---

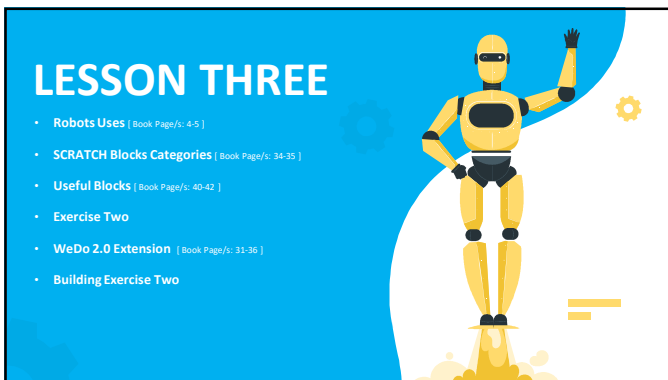
---

---

---

---

---




---

---

---

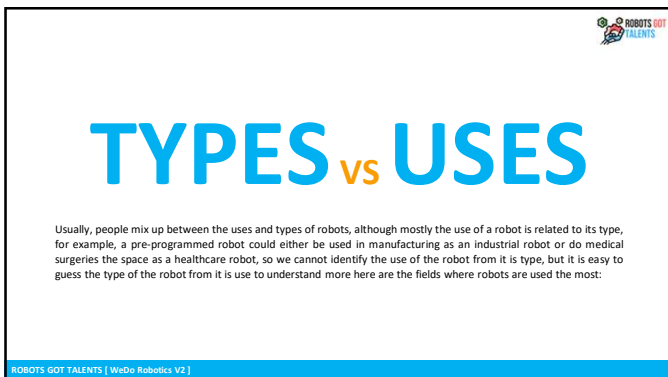
---

---

---

---

---




---

---

---

---


---

---


---

---

**USES OF ROBOTS**



**INDUSTRY:**  
Industrial robots are electronically controlled, both programmable and reprogrammable to carry out certain tasks with high precision and accuracy. Robots have been extensively used in highly advanced manufacturing facilities or high volume assembly lines for a long time. They are efficient and produce high yields or output.



**AEROSPACE:**  
Another application of robots is in aerospace for outer space exploration. Aerospace robots or unmanned robotic spacecraft play a key role in outer space probe. Scientists can explore in outer space without putting themselves in great danger considering the risks involved if they go to outer space themselves.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---


---

---


---

---

---



**HEALTHCARE:**  
A highly possible advancement in healthcare is using robots in robotic surgery. Due to technological advancement, this is possible even if the patient is located in remote areas. This possibility defies distance. With the proper tools and set-up in place, proper healthcare could be delivered to the patient even in remote areas without the corresponding risks involved.



**MILITARY:**  
In the military and public safety sectors, robotic technology is being applied in many areas. These machines can be used for surveillance and support operations on the battlefield. Military drones flying over areas of war and conflict, in hostage situations, and for natural and manmade disasters are able to assess danger levels and provide soldiers and first responders with real-time information.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---

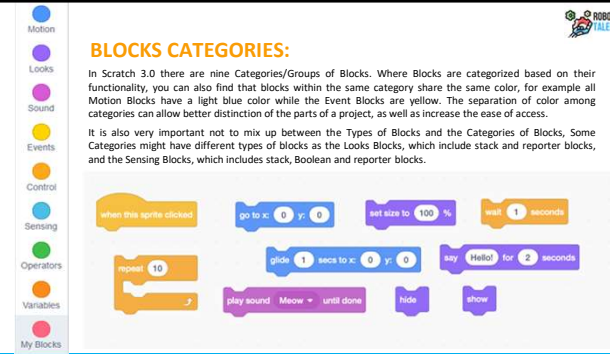
---

---

**BLOCKS CATEGORIES:**

In Scratch 3.0 there are nine Categories/Groups of Blocks. Where Blocks are categorized based on their functionality, you can also find that blocks within the same category share the same color, for example all Motion Blocks have a light blue color while the Event Blocks are yellow. The separation of color among categories can allow better distinction of the parts of a project, as well as increase the ease of access.

It is also very important not to mix up between the Types of Blocks and the Categories of Blocks, Some Categories might have different types of blocks as the Looks Blocks, which include stack and reporter blocks, and the Sensing Blocks, which includes stack, Boolean and reporter blocks.



ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---


---

---


---


---


---



**BLOCKS CATEGORIES:**

- 
**Motion**

Motion blocks deal with the movement of sprites. They relate mainly to the x and y position and direction of the sprite, as almost all the blocks correspond to them. The Stage does not contain any Motion blocks since it is a still object. There are currently 18 Motion blocks: 15 stack blocks and 3 reporter blocks.
- 
**Events**

Events blocks are related to various triggers in a project, or when one part signals another to run. The Events blocks used to be part of the Control category prior to Scratch 2.0. The "Events" category consists mostly of hat blocks, as there are currently 6 Hat blocks and just 2 Stack blocks.
- 
**Looks**

Looks blocks are related to the appearance of sprites and the stage. Some of the functionalities are changing costumes and applying graphic effects. There are currently 20 Looks Blocks: 17 Stack blocks and 3 Reporter blocks. 14 are for sprites, and 4 are for the Stage.

Check your book pages: 34-35

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---


---

---


---


---


---



**BLOCKS CATEGORIES:**

- 
**Sound**

Sound blocks are related to playing various sounds, whether they be MIDI notes or saved sounds. There are currently 9 Sound blocks: 8 Stack blocks and 1 Reporter block.
- 
**Control**

Control blocks run the basic flow of a project in the desired fashion, whether it be organized or unexpected. They provide functions for looping various blocks and scripts. They "control" the project and enhance its running. There are currently 11 Control blocks: 1 Hat block, 5 C blocks, 3 Stack blocks and 2 Cap blocks. Some types of blocks are only available in the control blocks as you read there are some types of Blocks are only available in the Control Blocks as the C Blocks and the Cap Blocks.
- 
**Sensing**

Sensing blocks associate with sprites and the stage detecting conditions. For example, sensing blocks can be used to detect when one sprite touches another. They consist of many Booleans and can work with Control blocks to stabilize a project' a flow. There are currently 18 Sensing blocks: 3 stack blocks, 5 Boolean blocks and 10 reporter blocks.

Check your book pages: 34-35

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---


---

---


---


---

---



**BLOCKS CATEGORIES:**

- 
**Operators**

Operator's blocks deal with many mathematical functions within a project and provide the capabilities of simple to complex mathematical operations. "Operators" also contains blocks for modifying strings and implementing them into various uses. There are some Boolean blocks, too, in which some are related to mathematical outputs, while others are used for adjoining other Booleans into one or a different output condition. There are currently 18 Operators blocks: 7 Boolean blocks and 11 Reporter blocks.
- 
**Variables**

Data blocks include two subcategories, Variables and Lists, but both are related to storing and accessing data. Prior to Scratch 2.0, this category was called "Variables". Data blocks are used for storing information, such as a score in a project, and using it in scripting and other beneficial purposes. There are currently 17 Variables blocks: 11 Stack blocks, 5 Reporter blocks, and 1 Boolean block. There are 5 variable blocks and 12 list blocks.

Check your book pages: 34-35

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---


---

---

---

---

---



### BLOCKS CATEGORIES:

**My Blocks**

My blocks are blocks that hold custom procedures for a selected sprite. The blocks are useful for running a script without screen refresh and organization of the scripts. Clicking Make a Block brings up a dialogue allowing the user to make a procedure. Once OK is pressed, the new block appears in the palette and an empty definition appears in the code area. When the procedure runs, Scratch will run the blocks below the corresponding Define block.

Check your book pages: 34-35

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---


---

---

---


---

---




### USEFUL BLOCKS [PART 1]

Below you can find the blocks which you might use regularly in your scripts, they are usually from the control, events, variables categories:



**Block Name:** Forever  
**Block Category:** Control  
**Block Type:** Cap & C



**Block Name:** Repeat ()  
**Block Category:** Control  
**Block Type:** C

The Blocks held inside the forever block will be in a loop which never ends unless the pause button is pressed, the Stop All block is activated, or the stop script block is activated within the loop.

The Blocks held inside the Repeat() block will repeat for a given number of times, before going to the next blocks. Note that the Repeat() Block only accepts positive integers.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---


---


---

---


---

---



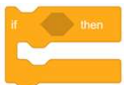


**Block Name:** Wait () Seconds  
**Block Category:** Control  
**Block Type:** Stack



**Block Name:** Wait Until ()  
**Block Category:** Control  
**Block Type:** Stack

The Wait () Seconds block pauses the script for the specified amount of seconds. The block pauses its script until the specified Boolean condition is true.



**Block Name:** If () Then  
**Block Category:** Control  
**Block Type:** Cap & C

The If () Then block checks its Boolean condition; if the condition inputted is true, the blocks held inside it will run, and then the script involved will continue. If the condition is false, the scripts inside the block will be ignored.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

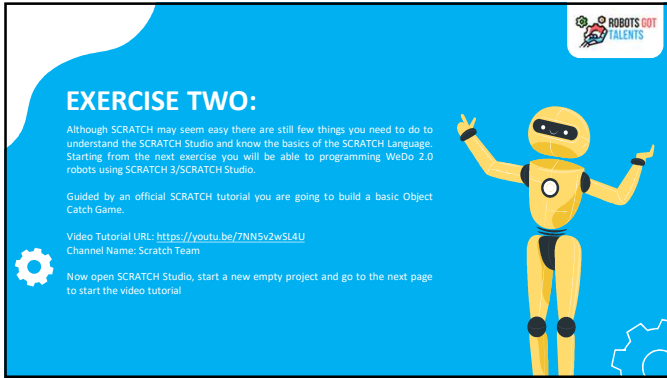
---

---

---

---

---



**EXERCISE TWO:**

Although SCRATCH may seem easy there are still few things you need to do to understand the SCRATCH Studio and know the basics of the SCRATCH Language. Starting from the next exercise you will be able to programming WeDo 2.0 robots using SCRATCH 3/SCRATCH Studio.

Guided by an official SCRATCH tutorial you are going to build a basic Object Catch Game.

Video Tutorial URL: <https://youtu.be/7NN5v2wSL4U>  
Channel Name: Scratch Team

Now open SCRATCH Studio, start a new empty project and go to the next page to start the video tutorial

---

---

---

---

---

---

---

---



How to make a  
**Catch Game**  
in **SCRATCH**

---

---

---

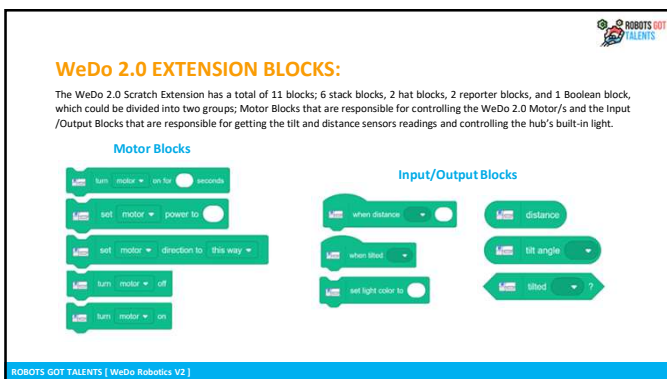
---

---

---

---

---



**WeDo 2.0 EXTENSION BLOCKS:**

The WeDo 2.0 Scratch Extension has a total of 11 blocks; 6 stack blocks, 2 hat blocks, 2 reporter blocks, and 1 Boolean block, which could be divided into two groups; Motor Blocks that are responsible for controlling the WeDo 2.0 Motor/s and the Input /Output Blocks that are responsible for getting the tilt and distance sensors readings and controlling the hub's built-in light.

**Motor Blocks**

- turn motor on for seconds
- set motor power to
- set motor direction to this way
- turn motor off
- turn motor on

**Input/Output Blocks**

- when distance
- distance
- when tilted
- tilt angle
- set light color to
- tilted

---

---

---

---

---

---

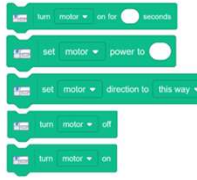
---

---



### SCRATCH WeDo 2.0 BLOCKS

**MOTOR BLOCKS**



4.10	4.10 Motor On for Seconds Block
4.11	4.11 Set Motor Power to Block
4.12	4.12 Set Motor Direction to Block
4.13	4.13 Turn Motor Off Block
4.14	4.14 Turn Motor On Block

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

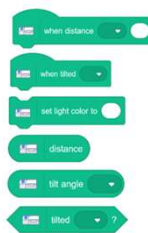
---

---

---

### SCRATCH WeDo 2.0 BLOCKS

**INPUT & OUTPUT BLOCKS**



4.15	4.15 When Distance Block
4.16	4.16 When Tilted Block
4.17	4.17 Set Light Colour to Block
4.18	4.18 Distance Block
4.19	4.19 Tilt Angle Block
4.20	4.20 Tilted Boolean Block

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---

---

---

### WeDo 2.0 SCRATCH SETTINGS:


**Adding the WeDo 2.0 Extension:**

- Open the Scratch extensions menu/library below the Blocks Pt
- Choose the LEGO Education WeDo 2.0 extension

**Connecting the WeDo 2.0 Hub:**

- Enable Bluetooth
- Download and Install the WeDo /Scratch link
- Choose your WeDo 2.0 hub then press connect

Download Scratch Link: <https://scratch.mit.edu/wedo>



ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---


---

---

---

---

---




## BUILDING EXERCISE (2)

In this exercise you will build a WeDo 2.0 helicopter, press on the button below to download the building instructions , for RoboApp users please long press the full screen button, to be able to access the downloadable files.

The following building Instructions was created and provided by LEGO EDUCATION FOR the WeDo 2.0 Set:

Building Instructions Name: Helicopter  
Created by: LEGO EDUCATION  
Difficulty: Medium  
Number of Pieces: > 40



Download Instructions

---

---

---

---

---

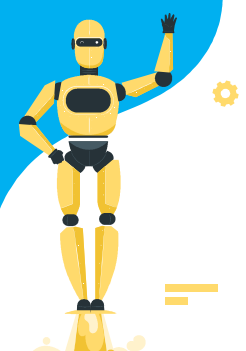
---

---

---

## LESSON FOUR

- **Flowchart** [ Book Page/s: 17-18 ]
- **Motor Blocks** [ Book Page/s: 37 ]
- **Types of Robots** [ Book Page/s: 5-7 ]
- **Programming Exercise Three**
- **History of Robotics** [ Book Page/s: 8 ]
- **Useful Blocks** [ Book Page/s: 40-42 ]



---

---

---


---


---

---

---

---






### Flowchart

Flowcharts represent algorithms in a graphical manner based on shapes as ovals, parallelogram, square and diamond. There is a standard set of rules we must follow when we draw flowcharts. Here are some simple examples of algorithms written by Flowchart:

Example One



**EXAMPLE ONE**

Here is a non-computer related example, where it's asked whether it's raining today or not if the answer is YES, an umbrella will be taken, and if the answer is NO, the umbrella will be left home.

As you can recognise the flowchart starts with an oval shape labelled "START" and ends with the same shape labelled "END" which identifies the end of the flow chart. The question asked is an Input value, so it's written inside a parallelogram with an additional output text "is it raining". The Condition in this flowchart, has 2 paths if the replay is YES the person will take an umbrella, while if the answer is NO the person won't take the umbrella. The action done or the Process, whether to take an umbrella or not is written in a square.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

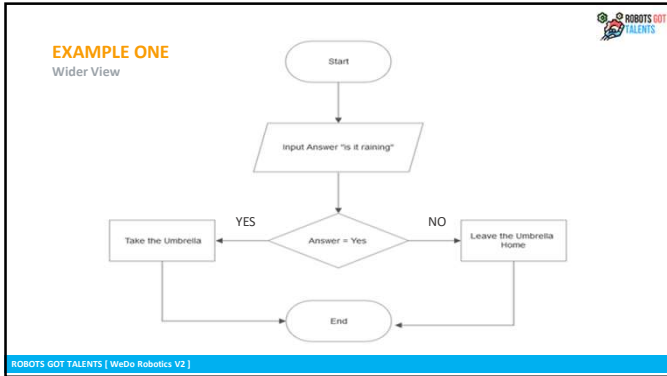
---

---

---

---

---




---

---

---

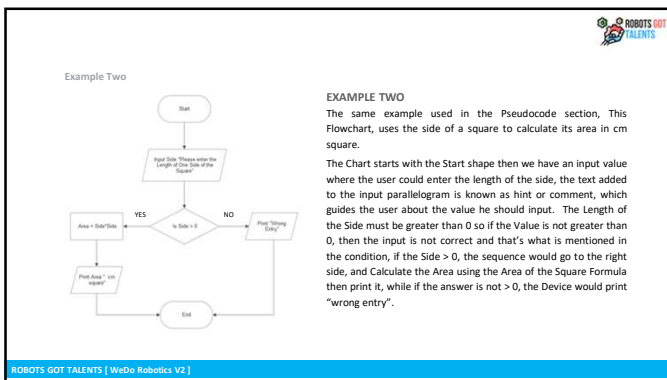
---

---

---

---

---




---

---

---

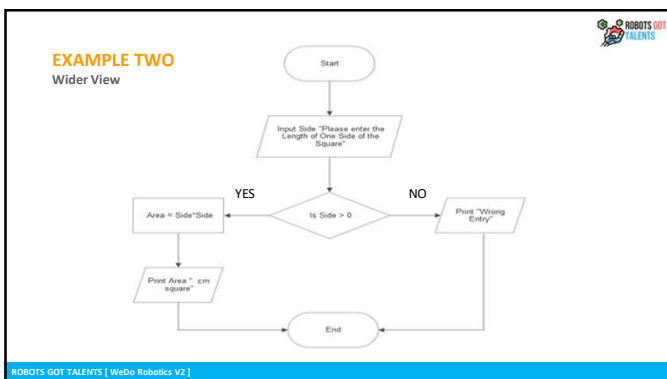
---

---

---

---

---




---

---

---





---

---

---

---

---

Symbol	Name	Description
	Beginning of the algorithm	It denotes the start of the algorithm. Everything in the algorithm comes ultimately from this symbol.
	Input statement/ Output statement	Input statements either read from another file or receive input from the user. Output statements output information onto the screen.
	Assignment/ Process statement	Assignment statements assign a value to a variable.
	Procedure call	This symbol 'calls' a pre-defined procedure or function.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

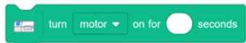
---

---

---

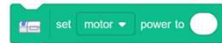
### MOTOR BLOCKS:

The Motor Blocks are the commands responsible for controlling the WeDo 2.0 Motor/s. There are currently 5 stack motor blocks. Scratch WeDo Motor Blocks manages the power, direction, and the duration of working:



Block Name: Motor On for Seconds  
Block Type: Stack

**The Turn Motor On for ( ) Seconds Block:** Starts one or two WeDo 2.0 motors for a chosen amount of time specified in seconds. The amount of time can be set with a numeric input, using whole or decimal numbers.



Block Name: Set Motor Power to  
Block Type: Stack

**Set Motor Power to ( ):** Sets the WeDo 2.0 motor/s power to the specified level which can be set with a numeric input from 0 to 100 Only whole numbers accepted.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

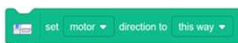
---

---

---

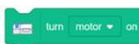
---

---



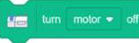
Block Name: Set Motor Direction to  
Block Type: Stack

**Set Motor Direction to ( ):** Sets the WeDo 2.0 motor/s to turn the axle in the direction chosen from the drop down menu, it could be any of the following: "this way", "that way", or "reverse"



Block Name: Turn Motor On  
Block Type: Stack

**Turn Motor On Block:** Activates WeDo 2.0 motor/s, until it is manually turned off with the turn motor off block or until a timer is added with the turn motor on for ( ) seconds block.



Block Name: Turn Motor Off  
Block Type: Stack

**Turn Motor Off:** Deactivates WeDo 2.0 motor/s

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---

**TYPES OF ROBOTS:**

**Pre-programmed Robots**  
Pre-programmed robots operate in a controlled environment where they do simple, monotonous tasks. An example of a pre-programmed robot would be a mechanical arm on an automotive assembly line. The arm serves one function — to weld a door on, to insert a certain part into the engine, etc.

**Humanoid Robots**  
Humanoid robots are robots that look like and/or mimic human behavior. These robots usually perform human-like activities (like running, jumping and carrying objects), and are sometimes designed to look like us, even having human faces and expressions.

**Autonomous Robots**  
Autonomous robots operate independently of human operators. These robots are usually designed to carry out tasks in open environments that do not require human supervision.

**Human Controlled Robots**  
Human Controlled Robots are usually directly controlled by humans via a method of wired/wireless connection. These robots usually work in extreme geographical conditions, weather, and circumstances.

**Augmenting Robots**  
Augmenting robots either enhance current human capabilities as the exoskeletons built by Hyundai to carry heavy objects or robots that replace the capabilities a human may have lost as Robotic arms and legs. Some examples of augmenting robots are robotic prosthetic limbs or exoskeletons used to lift hefty weights.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---

---

---

**EXERCISE THREE:**

Ready for your first WeDo 2.0 Scratch Exercise Add the WeDo 2.0 Extension and make sure the WeDo 2.0 hub is connected to your device. If you are having any problems please [press here](#)

**TASK (1) :** Program your WeDo 2.0 Helicopter to move its motor for 5 seconds with a power of 50 to the right then wait 3 seconds and rotate for 4 seconds in the opposite direction.

**TASK (2) :** Do Task (1) without using Motor On for Seconds Blocks



ROBOTS GOT TALENTS

---

---

---

---

---

---

---

---

---

---

**EXERCISE THREE SOLUTIONS:**

**TASK (1)**

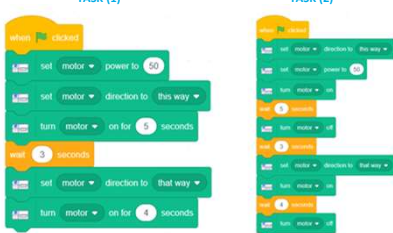

```

when clicked
  set motor to power to 50
  set motor to direction to this way
  turn motor on for 5 seconds
  wait 3 seconds
  set motor to direction to that way
  turn motor on for 4 seconds
  
```

**TASK (2)**

```

when clicked
  set motor to direction to this way
  set motor to power to 50
  turn motor on
  wait 3 seconds
  set motor to left
  turn motor on
  set motor to direction to that way
  turn motor on
  wait 3 seconds
  set motor to left
  turn motor on
  
```

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---


---

---

---

---

---



### HISTORY OF ROBOTICS:

The first known robot was created around 400-350 BC by the mathematician Archytas and was an artificial bird. Archytas, who is known as the "father of mechanical engineering," constructed his bird out of wood and used steam to power the movements of the robot. This bird was then suspended from a pivot bar. In its best-recorded run, it "flew" about 200 meters before running out of steam. This is not only the first known robot but was also one of the first recorded instances of a scientist researching how birds fly.

The modern concept began to be developed with the onset of the Industrial Revolution, which allowed the use of complex mechanics, and the subsequent introduction of electricity. This made it possible to power machines with small compact motors. In the early 20th century, the notion of a humanoid machine was developed. Today, one can envisage human-sized robots with the capacity for near-human thoughts and movement.

The first uses of modern robots were in factories as industrial robots – simple fixed machines capable of manufacturing tasks which allowed production with less need for human assistance. Digitally controlled industrial robots and robots using artificial intelligence have been built since the 2000s.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---


---

---

---


### USEFUL BLOCKS [PART 2]

Below you can find the blocks which you might use regularly in your scripts, they are usually from the control, events, variables categories:



**Block Name:** Repeat Until ()  
**Block Category:** Control  
**Block Type:** C

Blocks held inside the Repeat Until () block will loop until the specified Boolean statement is true, or a certain event occur.



**Block Name:** Set () to ()  
**Block Category:** Variables  
**Block Type:** Stack

The Set () variable to () variable block will set the specified variable to the given value (string or number)

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---


---

---


---

---

---



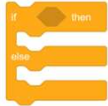
**Block Name:** Change () by ()  
**Block Category:** Variables  
**Block Type:** Stack



**Block Name:** Variable ()  
**Block Category:** Variables  
**Block Type:** Reporter

The block will change the specified variable by a given amount. If the variable is a string and not a number, it is set to the quantity the variable was to be changed by (casting the string to 0).

Whenever a variable is created, a version of the block appears with the variable's given name on it – this results in a version of this block for every variable. Each version of the block holds its 'assigned' variable.



**Block Name:** If () then, else  
**Block Category:** Control  
**Block Type:** Cap & C

The If () then, else block checks its Boolean condition; if the condition inputted is true, the code held inside the first C (space) will activate; if the condition is false, the code inside the second C will activate.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

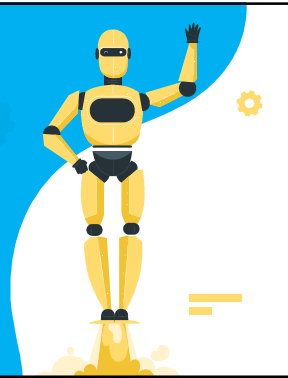
---

---

---

# LESSON FIVE

- Useful Blocks | Book Page/s: 40-42 |
- Programming Exercise Four
- Building Exercise Four
- Programming Exercise Five
- Input/Output Blocks | Book Page/s: 38 |
- Exercise Six




---

---

---

---

---


---

---

---


## USEFUL BLOCKS [PART 3]

Below you can find the blocks which you might use regularly in your scripts, they are usually from the control, events, variables categories:



Block Name: When Green Flag Clicked  
Block Category: Events  
Block Type: Hat

Scripts placed below the When Green Flag Clicked, also known as the Start Block will activate when the Green Flag (Play Button) is pressed.



Block Name: When () Key Pressed  
Block Category: Events  
Block Type: Hat

Scripts placed below the When () Key Pressed will activate when a specified pre-chosen key is pressed. The keys that can be sensed with this block include all the English alphabet (a, b, c ...), the number keys (0, 1, 2 ...), the arrow keys (← → ↵ ↓), and the space key. The block also include the "Any" option which activates the code when any key is pressed

ROBOTS GOT TALENTS | WeDo Robotics V2 |

---

---

---

---

---

---

---

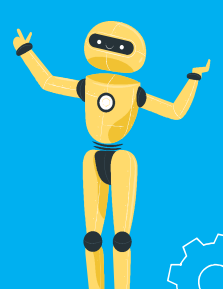
---

## EXERCISE FOUR:

Do not forget to add the WeDo 2.0 Extension and make sure the WeDo 2.0 hub is connected to your device. If you are having any problems please [press here](#)

**TASK (1)** : Program your WeDo 2.0 Helicopter to move its motor with a power of 100 when the green flag is pressed and to stop the motor when the space key is pressed.

**TASK (2)** : When the green flag is pressed program your helicopter's motor to rotate to the left with a power 50 for 5 seconds then waits for 2 seconds and do that for two more times. You must not use more than 6 blocks.




---

---

---

---

---


---

---

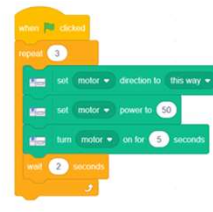
---


**EXERCISE FOUR SOLUTIONS:**

**TASK (1)**



**TASK (2)**





ROBOTS GOT TALENTS | WeDo Robotics V2

---

---

---

---

---

---

---

---

**BUILDING EXERCISE (4)**

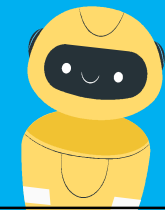
In this exercise you will build WeDo 2.0 robot MILO. The Building Instructions for MILO are in two files, first download the "Base Instructions" then the "Arm Instructions" press the buttons below to download the building instructions , for RoboApp users please long press the full screen button, to be able to access the downloadable files.

The following building Instructions was created and provided by LEGO EDUCATION for the WeDo 2.0 Set:

Building Instructions Name: MILO  
 Created by: LEGO EDUCATION  
 Difficulty: Medium  
 Number of Pieces: > 40

Download Base Instructions

Download Arm Instructions




---

---

---

---

---

---

---


---

**EXERCISE FIVE:**

Do not forget to add the WeDo 2.0 Extension and make sure the WeDo 2.0 hub is connected to your device. If you are having any problems please [press here](#)

**TASK (1) :** Program MILO to do the following:

- When the up arrow is pressed: Move forward with half of its power
- When the down arrow is pressed: Move backwards with half of its power
- When the space bar is pressed: stop moving
- When the letter (a) is pressed: Move forward with half of its power for 2 seconds
- When the letter (b) is pressed: Move backwards with half of its power for 2 seconds




---

---

---

---

---


---

---

---



### EXERCISE FIVE SOLUTION:



**TASK [1]**

ROBOTS GOT TALENTS | WeDo Robotics V2

---

---

---

---

---

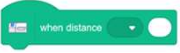
---

---

---


### INPUT/OUTPUT BLOCKS:

The Input and Output Blocks are the commands responsible for detecting the WeDo 2.0 Sensor Inputs and managing the WeDo 2.0 Outputs. There are currently 6 blocks: 2 hat blocks, 2 reporter blocks, 1 stack block and 1 Boolean block.



**Block Name:** When Distance  
**Block Type:** Hat

**The When Distance () Block:** Activates the code added below when the distance is less than or greater than the specified value, depending on the argument selected from the drop-down menu. options for the when distance block are "> greater than" "< smaller than"



**Block Name:** When Tilted  
**Block Type:** Hat

**The When Tilted () Block:** Activates the code added below when the tilt sensor reads a change in a direction according to the option selected from the drop-down menu, options for the when tilted block are "up", "down", "left", "right", and "any"

ROBOTS GOT TALENTS | WeDo Robotics V2

---

---

---

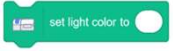
---

---

---

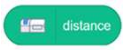
---

---



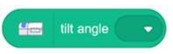
**Block Name:** Set Light Color to  
**Block Type:** Stack

**Set Light Color to ():** Sets the WeDo 2.0 hub's light to a specified value (representing colors)



**Block Name:** Distance  
**Block Type:** Reporter

**Distance Block:** The block reports the WeDo 2.0 distance sensor value



**Block Name:** Tilt Angle  
**Block Type:** Reporter

**Tilt Angle () Block:** Returns the angle detected by the WeDo 2.0 tilt sensor. Its options are "up", "down", "left", and "right".

ROBOTS GOT TALENTS | WeDo Robotics V2

---

---

---

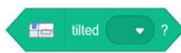
---

---

---


---

---



**Block Name:** Tilted Block  
**Block Type:** Boolean

**Tilted ( )?** Block: Returns whether or not the WeDo 2.0 tilt is being tilted in the specified direction. Its options are "up", "down", "left", "right", and "any".



ROBOTS GOT TALENTS | WeDo Robotics V2

---

---

---

---

---

---

---

---

---

---


## EXERCISE SIX:

Do not forget to add the WeDo 2.0 Extension and make sure the WeDo 2.0 hub is connected to your device. If you are having any problems please [press here](#).

**TASK (1)** : When the distance between MILO and the nearest object is less than 50 program the sprite to say "Hello" for 2 seconds and MILO to move forward with a power of 100 for 5 seconds.

**TASK (2)** : When the green flag is pressed MILO Should check using the appropriate sensor if there is an object near it. if yes MILO should stop and if no MILO should move forward with a power of 50 , then check that again.

There are different algorithms for creating Task two, so it is not a must to have the same code as the one in the Exercise Solution, if the robot is operating the task correctly then your code is correct !



ROBOTS GOT TALENTS

---

---

---

---

---

---

---

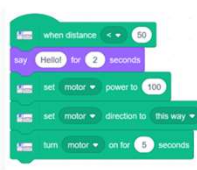
---

---


---


## EXERCISE SIX SOLUTIONS:

**TASK (1)**



**TASK (2)**





ROBOTS GOT TALENTS | WeDo Robotics V2

---

---

---

---

---

---

---

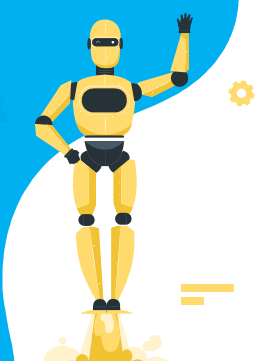
---

---

---

# LESSON SIX

- Pseudocode [Book Page/s: 40-42]
- Building Exercise Five
- Programming Exercise Seven
- LEGO Digital Designer
- LDD Modes & Controls
- LDD Basics




---

---

---

---

---

---

---

---

---

---

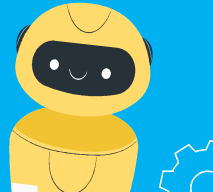
## BUILDING EXERCISE (5)

In this exercise you will just need to change the sensor in MILO, we do also recommend re-building MILO, as that would improve your building and designing skills, press the buttons below to download the building instructions , for RoboApp users please long press the full screen button, to be able to access the downloadable files.

The following building instructions was created and provided by LEGO EDUCATION for the WeDo 2.0 Set:

Building Instructions Name: MILO  
 Created by: LEGO EDUCATION  
 Difficulty: Medium  
 Number of Pieces: < 30

[Download Base Instructions](#)   [Download Arm Instructions](#)




---

---

---

---

---

---

---

---

---

---

### Pseudocode

Pseudocode is an informal way to represent an algorithm in a narrative manner. With established conventions we must follow, this includes keywords, which are usually capitalized as (INPUT, OUTPUT, PRINT, IF, ELSEIF, REPEAT), and formatting features like indentation. Here are some simple examples of algorithms written in pseudocode:

**Example One**

```

1. PRINT "Please enter one of the Robotics Characteristics "
2. INPUT Answer
3. IF Answer == "Movement" OR Answer == "Sensing" OR Answer == "Energy" OR Answer == "Intelligence"
4.   THEN PRINT Answer " is one of the Robotics Characteristics"
5.   ELSE PRINT "Wrong Answer"
6.   ENDIF
    
```

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---


---

---

---

---

---



**EXAMPLE ONE:**

This Pseudocode asks the user to input any of the Robotics Characteristics, and if the Answer is correct it would print that the answer is correct. The sequence that the code runs with is line by line, so let's explain the algorithm in that way:

Line1: A message saying "please enter one of the Robotics Characteristics" will be printed on the screen

Line2: A Cursor will appear on the screen for the user to type the answer

Line3: Using the Logical condition of IF, the computer checks whether the inputted answer is "Movement", "Sensing", "Energy" or "Intelligence". [LOGICAL EXPRESSIONS will be explained later in this book]

Line4: If the answer is [correct] any of the mentioned words, the Computer will print the answer then " is one of the Robotics Characteristics", there is space before the printed statement, so that the answer could be easily seen.

Line5: If the answer is [not correct] any of the mentioned words, the computer will print "Wrong Answer"

Line6: The logical statement is closed, and the code ends.

ROBOTS GOT TALENTS | WeDo Robotics V2 |

---

---

---

---

---


---

---

---

---

---



**Example Two**

```

1. PRINT "Please enter the length of the side in cm "
2. INPUT Side
3. Area = Side * Side
4. PRINT Area " cm2"
    
```

This algorithm is much easier than the one before, it is for a program that is made to calculate the area of a 2D square using the Formula [AREA = SIDE x SIDE], the Computer asks the user to input the length of any of the sides then it uses the formula of the area to calculate the Area of the square, after that it prints the total in cm.

Line1: A message saying "Please enter the length of the side in cm" will be printed on the screen

Line2: A Cursor will appear on the screen for the user to type the length of the side

Line3: Using the formula Area = Side \* Side, the computer will calculate the area

Line4: the Computer will print the Area then " cm", there is space before the unit of the area, so that the value could be easily identified.

ROBOTS GOT TALENTS | WeDo Robotics V2 |

---

---

---

---

---


---

---

---

---

---



**PSEUDOCODE FORMATTING & KEYWORDS**

<b>INPUT</b>	Used to input/add a value or a statement to the code.
INPUT "Please Enter your name" Name	
<b>PRINT, OUTPUT</b>	Used to Print/output a value or a statement.
PRINT "Your name is " Name	
<b>IF.....ELSE.....ENDIF</b>	A type of logical expressions that asks for a certain condition to complete the code.
IF Password = "12345" THEN PRINT "Correct Password" ELSE PRINT "Wrong Password" ENDIF	

ROBOTS GOT TALENTS | WeDo Robotics V2 |

---

---

---

---

---

---

---

---

---

---

### PSEUDOCODE FORMATTING & KEYWORDS

<b>REPEAT.....UNTIL</b>	A type of Loops (Repeat functions) that repeats a part of the code until a certain condition is found.
<pre> REPEAT   INPUT Password UNTIL Password = "12345"           </pre>	
<b>FOR.....TO.....NEXT</b>	A type of Loops (Repeat functions) that repeats a part of a known number of times before going to the next part of the code
<pre> FOR Count 1 TO 5   INPUT Password NEXT Count           </pre>	

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---

---

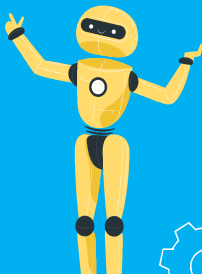
---

### EXERCISE SEVEN:

Do not forget to add the WeDo 2.0 Extension and make sure the WeDo 2.0 hub is connected to your device. If you are having any problems please [press here](#).

**TASK (1)** : Program MiLO to move forward with a power of 50 for 5 seconds if the tilt sensor is tilted down and move backward with a power of 50 for 5 seconds if the tilt sensor was tilted up.

**TASK (2)** : Program your sprite to move upwards when the tilt sensor is tilted up downwards when the sensor is tilted down, right when the sensor is tilted right and left when the sensor is tilted left. You might need to use the change coordinates by () Blocks.



ROBOTS GOT TALENTS

---

---

---

---

---

---

---


---

---

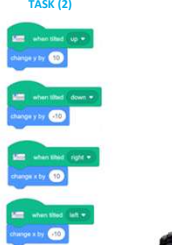
---


### EXERCISE SEVEN SOLUTIONS:

**TASK (1)**



**TASK (2)**





ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---

---

---

### LEGO DIGITAL DESIGNER

LEGO Digital Designer (LDD) is a freeware computer program produced by the Lego Group as a part of LEGO Design byME. It is available for macOS and Windows. The program allows users to build models using virtual Lego bricks, in a computer-aided design like manner. Until January 16, 2012, these could be uploaded, along with instructions and a box design, to the Lego Design byME website, from where the models could be ordered for delivery as a real, packaged set. Users can also take screenshots of their models and store the models on their computer in an .LXF file.



ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---


---

---

### WELCOME SCREEN:

Each time you open LEGO Digital Designer or start a new model, you'll see the pop up Welcome Screen, that includes the following modes:

**1.1**



**LEGO DIGITAL DESIGNER (1..1)**  
Used to build Static LEGO Models using the official LEGO pieces form most basic LEGO Sets.

**LEGO MINDSTORMS DESIGNER (1..1)**  
Used to build LEGO MINDSTORMS Models using pieces from Mindstorms Ev3, Mindstorms NXT, WEDO 2.0, WEDO and Power Functions

**LEGO DIGITAL DESIGNER EXTENDED (1..1)**  
Used to build up LEGO Models using Customizable LEGO Pieces from all the LEGO Sets

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

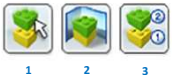
---

---

### LEGO Digital Designer has three operation modes:

1. Build mode
2. View mode
3. Building guide mode

You can switch between modes by clicking one of the three mode icons located in the bar at the top of the application.



ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

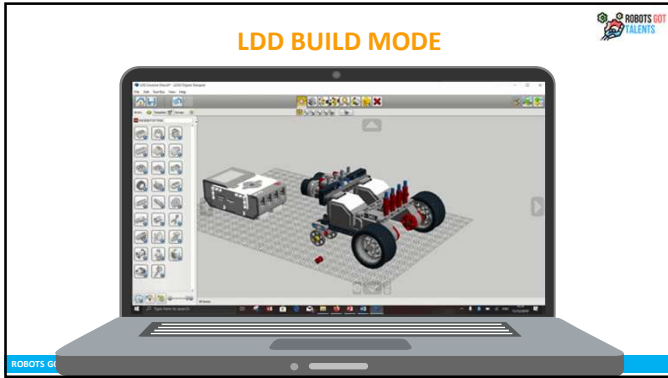
---

---

---

---

---



---

---

---

---

---

---

---

---



---

---

---

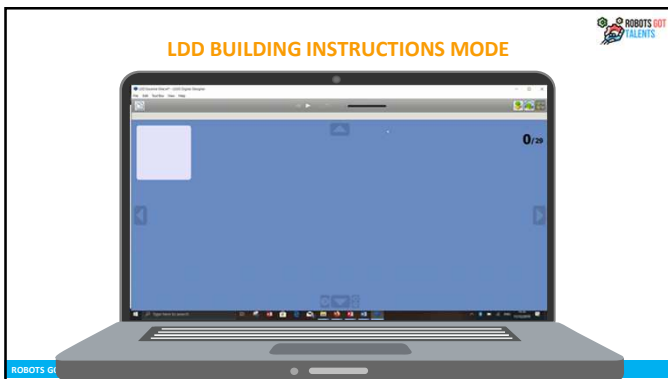
---

---

---

---

---



---

---

---

---

---


---

---

---

**LDD BUILD MODE:**

In this session we will learn about the Lego Digital Designer Building Mode, The Building Mode is the main mode you need to learn for using the LDD, here is the user interface for the Building Mode.



- 2.1 Brick palette: The Brick palette includes all bricks and pieces from different sets.
- 2.2 Model Building Area: It's the area, where you put bricks together to build up your model.
- 2.3 Building Tools: The Building Tools are tools that will help you build and modify your model.
- 2.4 Mode Selector: Buttons you will use to select the mode your going to work with
- 2.5 Brick Counter: The Brick Counter Shows how many pieces are included in the model.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---


---

---

---

---

---



ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---

---

---

**BRICK PALETTE**

The Brick palette includes all bricks and pieces from different sets. Choose and click on the brick you want to use, then drag it your model.

- Show group/Hide group: Shows or hides bricks in the group within the palette. Each group includes different types and shapes of bricks.
- Filter Bricks by LEGO® sets: Click this icon to choose a LEGO® set. Your brick list will display only the bricks contained in the selected set. It will count down to zero as you use the bricks. If the number of bricks used exceeded the bricks in the set chosen the number would turn to red with a negative value.
- Scale Divider: Resizes the dividers and their contents to better fit your screen resolution.
- Find bricks by colour: This will filter the brick list so it only shows bricks of the color or material you have selected.

Search field: Searches for the brick you need by Typing Keywords.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---









---

---



### BUILDING TOOLS:

The Building Tools are Tools included in the LEGO DIGITAL DESIGNER Build Mode, to help you building, customizing your model. Here are all the available Building Tools, if you do not know the shape of any tool, you can highlight it to see its name.

 <b>SELECTION TOOL</b> Used to select brick/s in your model	 <b>FLEX TOOL</b> Used to bend and twist flexible elements
 <b>CLONE TOOL</b> Used to duplicate brick/s from the model	 <b>PAINT TOOL</b> Used to change the color or material of bricks in the model
 <b>HINGE TOOL</b> Used to rotate bricks that are connected with a hinge or a single stud connection.	 <b>HIDE TOOL</b> Used to hide brick/s in your model
 <b>HINGE ALIGN TOOL</b> Used to automatically connect two separate connection points.	 <b>DELETE TOOL</b> Used to delete brick/s from the model

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---

### LDD CONTROLS:

**Mouse Controls**  
**Left mouse:** selects a brick or move it around on the scene if clicked & dragged.  
**Right mouse:** rotates the scene's camera view.  
**Mouse wheel scroll:** Used to zoom in and out in the scene.

**Camera Controls**  
**Rotate view:** (Num Lock, numbers 8, 2, 4, 6). You can rotate the camera view using either the buttons on the Camera control or your keyboard  
**Zoom view:** (Num Lock, + and - keys). Select the + and - keys on your keyboard or the + and - buttons on the Camera control to zoom in or out.  
**Reset view:** (Num Lock, number 5). Click Reset to return the view point to its standard position, with all bricks and models visible and centered on the scene.  
**Pan view:** (shortcut only) (Shift+right click). Hold down the Shift key and click the right mouse button to pan your model from the left, right, top or bottom.

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---

### HELPFUL KEYBOARD SHORTCUTS

TASK	WINDOWS OS	MAC OS
Rotate view left	4 Key (NumLock: ON)	4 Key (NumLock: ON)
Rotate view right	6 Key (NumLock: ON)	6 Key (NumLock: ON)
Rotate view up	8 Key (NumLock: ON)	8 Key (NumLock: ON)
Rotate view down	2 Key (NumLock: ON)	2 Key (NumLock: ON)
Reset view	5 Key (NumLock: ON)	5 Key (NumLock: ON)
Hinge tool	HKey	HKey
Hinge Align tool	Shift+H	Shift+H
Clone tool	CKey	CKey
Paint tool	BKey	BKey
Hide tool	LKey	LKey
Delete tool	DKey	DKey

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---

### HELPFUL KEYBOARD SHORTCUTS

TASK	WINDOWS OS	MAC OS
Group	Ctrl+G	Cmd+G
Create a template	Ctrl+ALT+G	Cmd+ALT+G
Open	Ctrl+O	Cmd+O
Save	Ctrl+S	Cmd+S
Print	Ctrl+P	Cmd+P
Undo	Ctrl+Z	Cmd+Z
Redo	Shift+Ctrl+Z	Shift+Cmd+Z
Upload to Gallery	Shift+Ctrl+B	Shift+Cmd+B

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---


---

---

---

### LDD VIEW MODE:

The LDD View Mode, is the mode where you can see your model and save its screenshots. This Mode is much easier than the Build mode as it do only include few features.



**3.1 View Mode Tools:**  
It includes all the tools available for the view mode

**3.2 Model View Area:**  
It's the area, where you can see your model, rotate it, zoom in or out

**3.3 Mode Selector:**  
Buttons you will use to select the mode your going to work with

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---




---

---

---

### VIEW MODE TOOLS:

As the Build Mode, the View Mode also includes some tools, that allows you to do few things with your robot while viewing it, In addition to the Mode Selector there are 3 tools in the View Mode toolbar.

- 
**Screenshot:**  
 This Button is used to take a screenshot of your model and save in your computer, but screenshots in the View Mode are different from screenshots taken using the shortcut Ctrl+K in the Build Mode, as those screenshots include the background and not transparent as screenshots taken from the Build Mode.
- 
**Explode Model:**  
 This Button is used to Explode your model, so you can see all the pieces you have used to build your model, But It's really sad seeing all your hard work explode :).
- 
**Change Background:**  
 This Button is used to change the background for your model, there are 4 backgrounds available, which represents 4 different environments Sand, Forest, Sea and Space

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---


---

---

---

### BUILDING GUIDE MODE:

The LDD Building Guide Mode, Shows you all the Steps you have done building your model as 3 to 4 bricks per page, also you can save your Building Guide as an HTML and PDF Building Instructions.



**4.1 Building Guide Mode Tools:**  
It includes all the tools available for the Building Guide Mode.

**4.2 Building Guide:**  
It's the area, where you can view all the steps you have done to build your model.

**4.3 Mode Selector:**  
Buttons you will use to select the mode your going to work with

ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

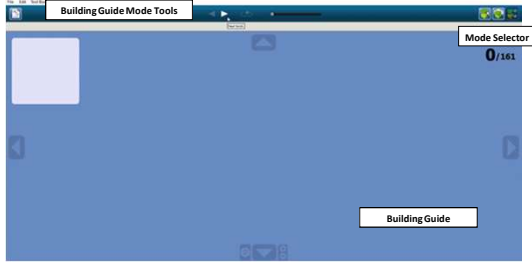
---

---

---

---

---



ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---

---

---

### EXPORTING/IMPORTING MODELS:

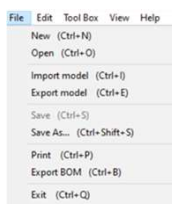
The LEGO DIGITAL DESIGNER Allows users to import and export models, that means that you can add a LEGO Model to your Software, or output your Model out of the LDD in the [.lxf] format, this format could be read by all LEGO DIGITAL DESIGNER Software on any computer.

**To Import a model you should follow the following Steps:**

1. Press File, then choose Import Model or press Ctrl + L
2. Choose the folder containing the file
3. Type in the file Name or Select it
4. Press Open

**To Export a model you should follow the following Steps:**

1. Press File, then choose Export Model or press Ctrl + E
2. Choose the folder you want to export your project in
3. Type in the file Name
4. Press Save



ROBOTS GOT TALENTS [ WeDo Robotics V2 ]

---

---

---

---

---

---

---

---

---

---



---

---

---

---

---

---

---

---