

SilverCoders

DIGITAL LITERACY IMPROVEMENT THROUGH EFFECTIVE
LEARNING EXPERIENCES FOR ADULTS



CHALLENGE 20 **GEOMETRY HUNTER**

CODING TRAINING PROGRAMME **FOR +55 ADULTS**



SILVER CODERS

ERASMUS+ No. 2020-1-SE01-KA227-ADU-092582



**Co-funded by
the European Union**

This document reflects only the author's view and the National Agency and the European Commission are not responsible for any use that may be made of the information it contains

STRUCTURE OF THE CHALLENGE

DESCRIPTION

You were provided with a simple setup that exemplifies the most important elements of the Gdevelop environment: a scene and a sheet of events. The scene has the Monster, your character or avatar (the object you control) and a set of geometric figures. The events available allow the user to move the monster and catch the geometric pieces. You will start by understanding the initial setup and then you will be asked to improve the game, making it more dynamic (pieces appear in different places) and to check when the player caught all the pieces.

GENERAL GOAL

In the Geometry Hunter game we play a monster that likes to eat geometrical pieces. Help the monster score as many points as possible.

LEARNING OBJECTIVES

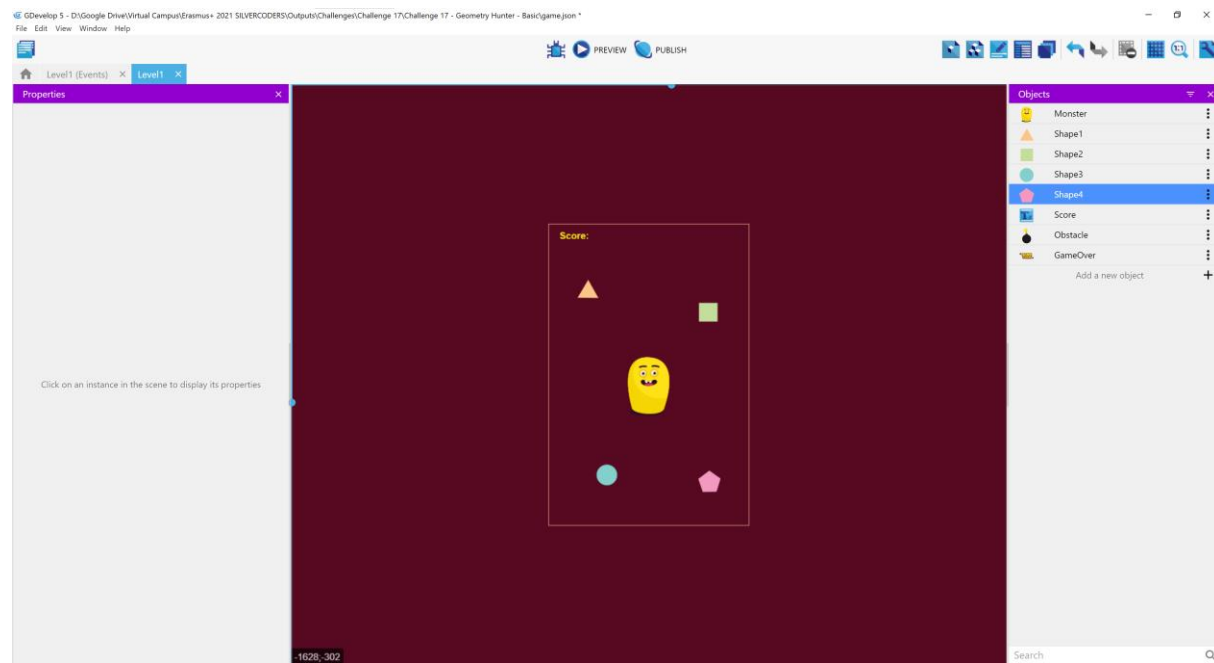
In the end of this challenge, you will be able ...:

- To understand how code is treated by a computer and what is the role of a compiler.
- To be familiar with the concept of low and high level languages and understand what their differences are and what is required to code in either of them.
- To have experience with a visual programming suite and be able to code standard small piece of software with it.
- Know what statements and command lines are and what they mean for a compiler.
- To be able to write instructions using correct syntax and with minimal errors.
- Know what operators are, what they do and which symbols stand for which operators.
- To be able to understand the assignment of values to variables and how to change them.
- To know all the basic arithmetic operations and how to use them.
- Recognize and know how to use all the data structures related to numbers.
- To know the structures linked to the use of text, such as strings and characters.
- To be able to use If statements correctly to execute code according to a certain defined fixed condition.
- To know how to use the Gdevelop editor
- To understand the concepts of scenes, events and objects

INSTRUCTIONS

Start by opening the Gdevelop editor and use the **File** menu to **open** the Challenge 20 – Initial

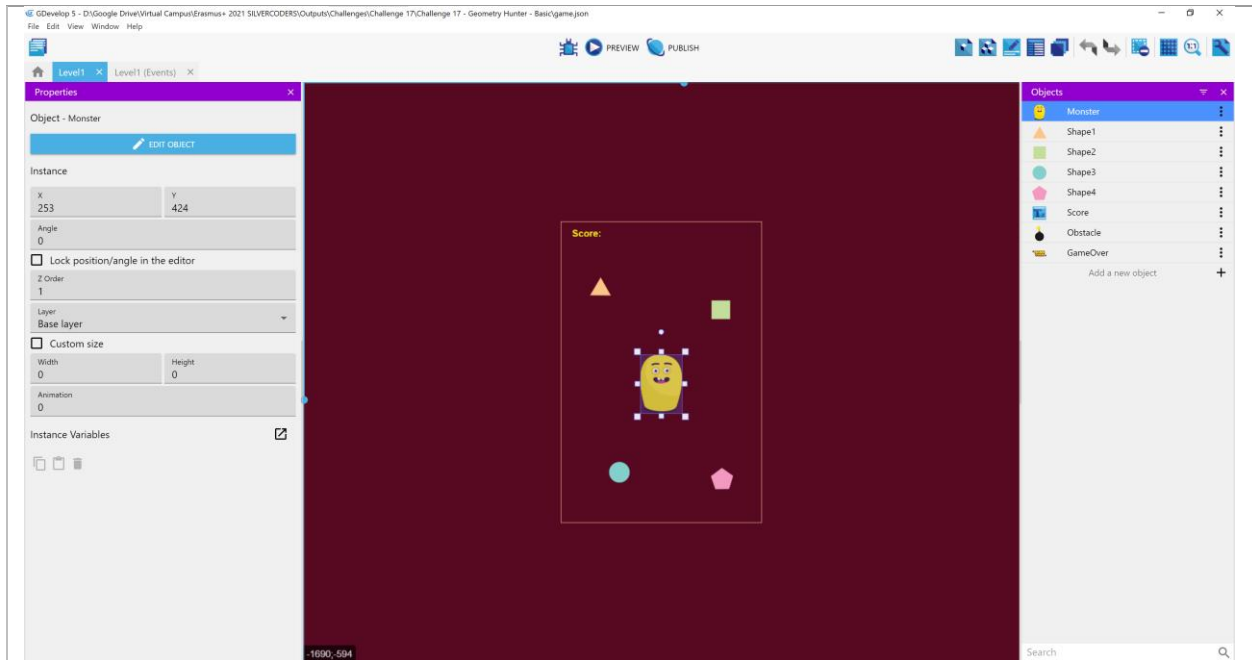
This should be what you get



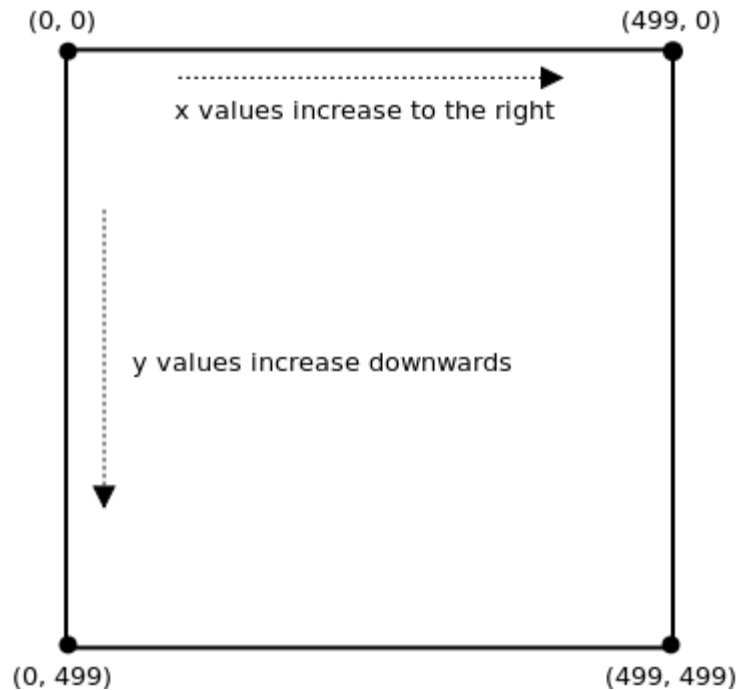
- Press the **Preview** button to play the game. You can move the monster with the arrow keys in your keyboard and you should catch the 4 geometrical shapes. For every piece you catch, one point is added to your score.
- Repeat the game as many times as you want. To repeat you have to close the game window and press the **Preview** button in the editor.

Now that you know what are the game mechanics (that is, what you can do in the game) let's recap some GDevelop concepts:

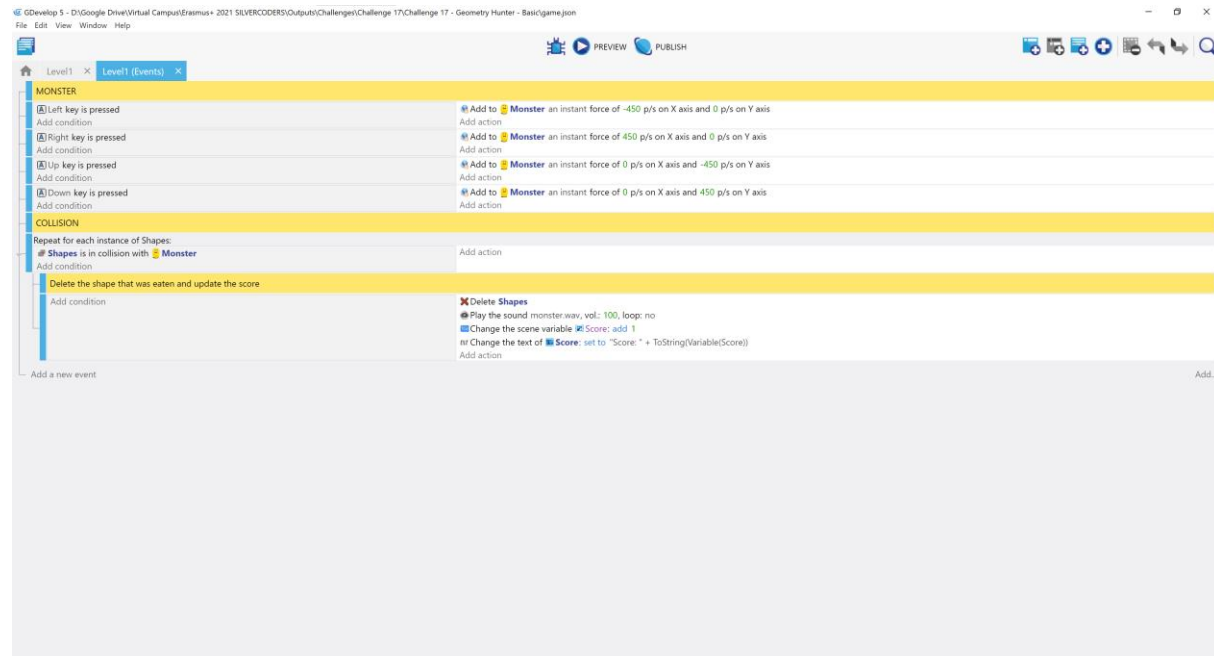
- In the middle of the screen you see the **scene**. That is the visual interface of the game. To the right you see the **Objects** of the game, the visual elements you play with. If you click on an **Object** you will see its properties on the left.
- In this game we have the object **Monster**, which is a **Sprite** (an animated image), four different **Shapes**, also **Sprites** and **ScoreTxt**, a text box.



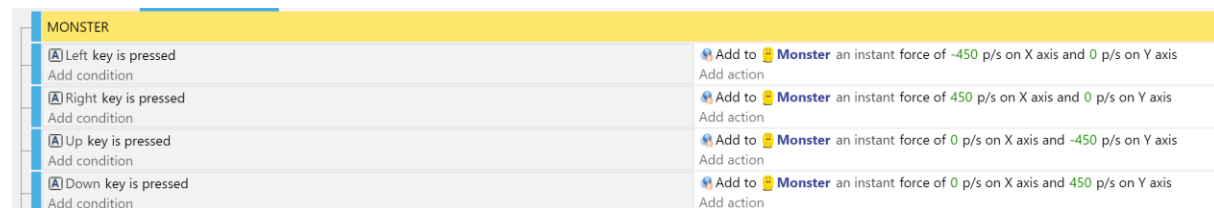
These are the properties for the monster object. For instance, you can see the X and Y values which is the object position (in pixels) in the scene. X and Y values start at 0 on the top-left corner of the screen and have their maximum value on the bottom-right corner of the scene. The maximum values can be changed.



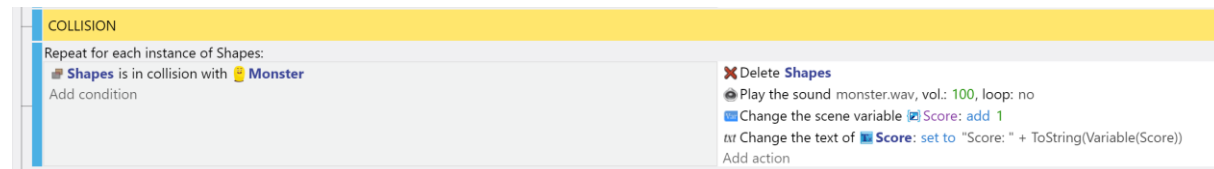
If you press the Level1 (Events) button, you can see the code that makes the game interactive. In Gdevelop the code is organized in **Events**.



Each event has two parts: to the left, there is one or more conditions; to the right, one or more actions that will take place when that condition happens.



These four events relate to the movement of the Monster. You can read that if the left arrow key is pressed a force is applied to the object Monster to make it move left (negative X). The same is done to the other three arrow keys.



The other event relates to when we move the Monster over a shape. This is called a **Collision**. When that happens several things happen:

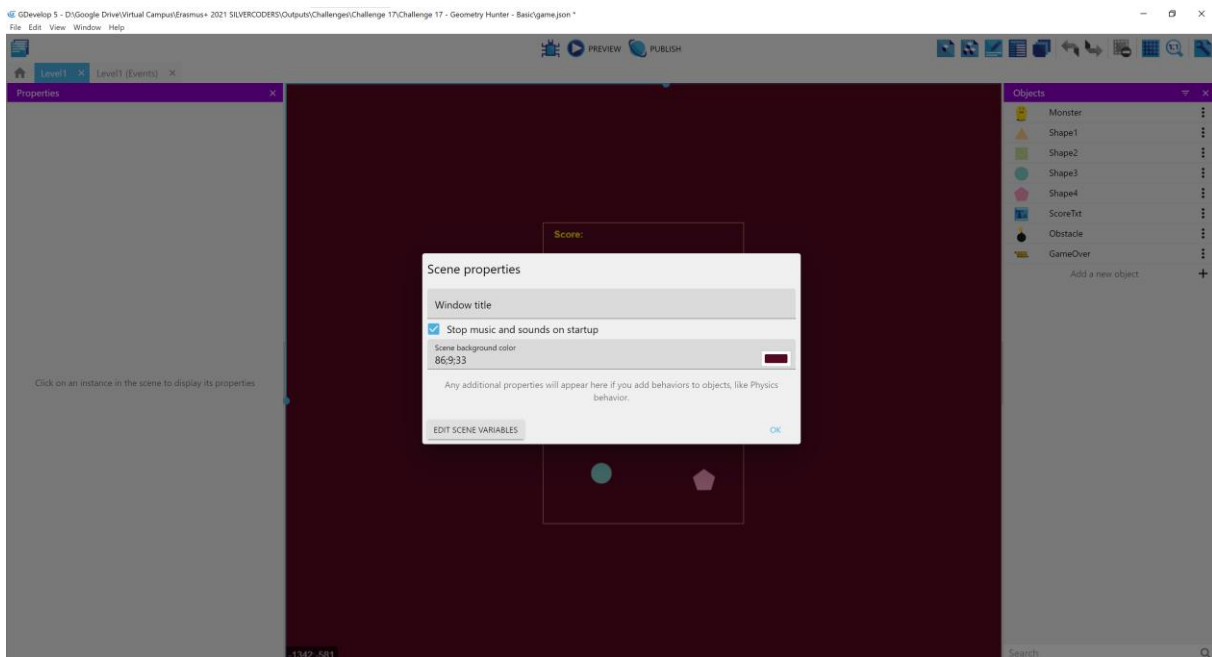
- The shape is deleted.
- A sound is played
- The value of the variable **Score** is increased by 1 point. A variable is a container for a value that can change while we play the game.
- The content of the **ScoreText** Text Box is updated to reflect the change in the variable **Score**. The content of a Text Box is a **String** that is a set of alphanumeric characters.

Note: Make sure that you understand all the concepts that we've present before moving on.

We are now going to make some changes in the game to make it more attractive. First, we'll have a random number of objects to catch to make the game different every time we play. We will start with the same 4 but when we catch one, another one will pop up until the maximum is reached.

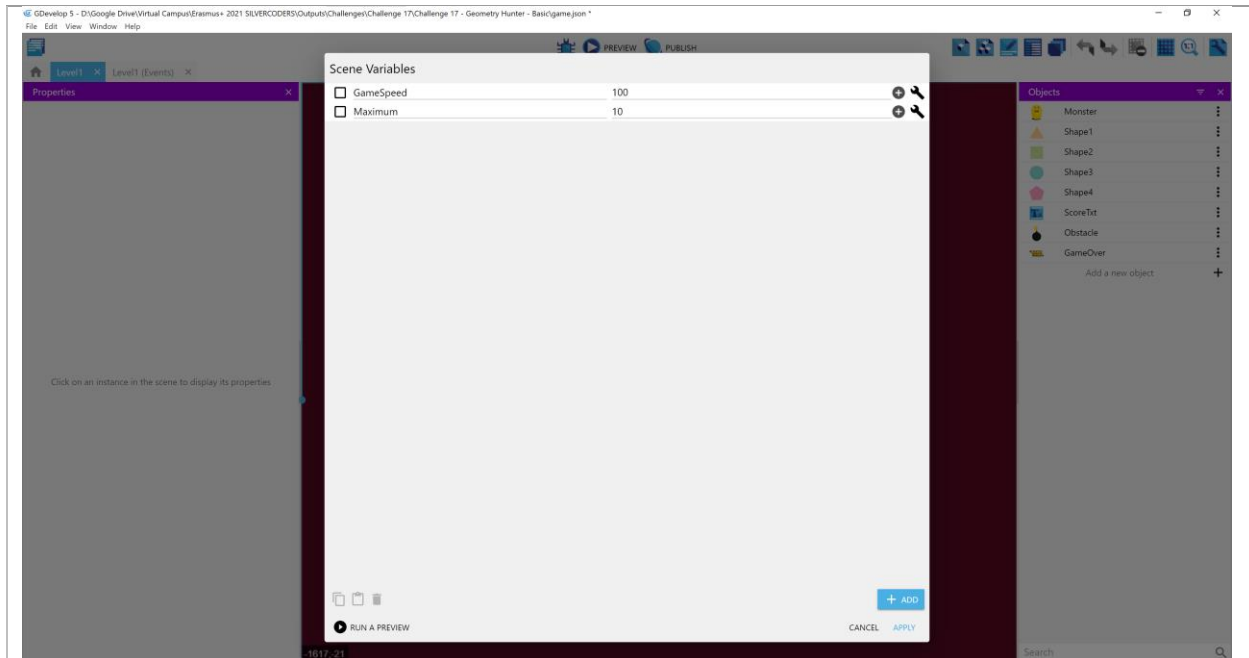
Let's start by creating a new variable Maximum that defines how many objects we will catch (we will set this amount to 10).

On the Scene screen, click with the right mouse button over the scene and select **Open Scene Properties**



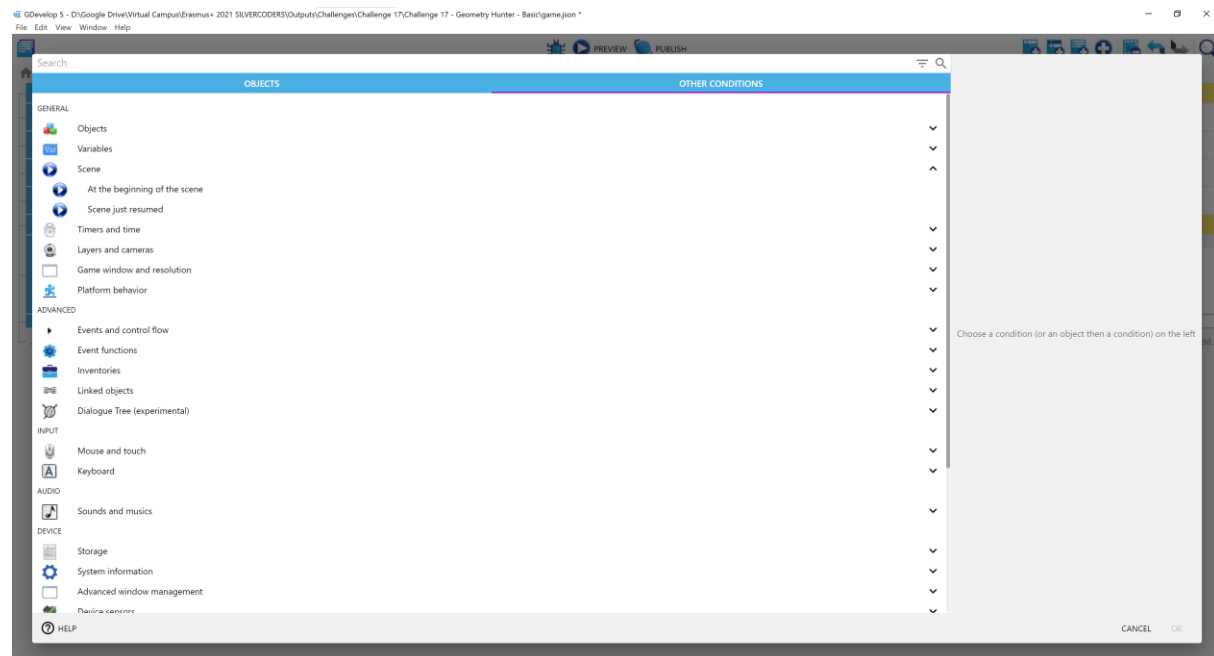
Then press **Edit Scene Variables** and then press **Add**

Change the variable name to **Maximum** and give it the initial value of **10**



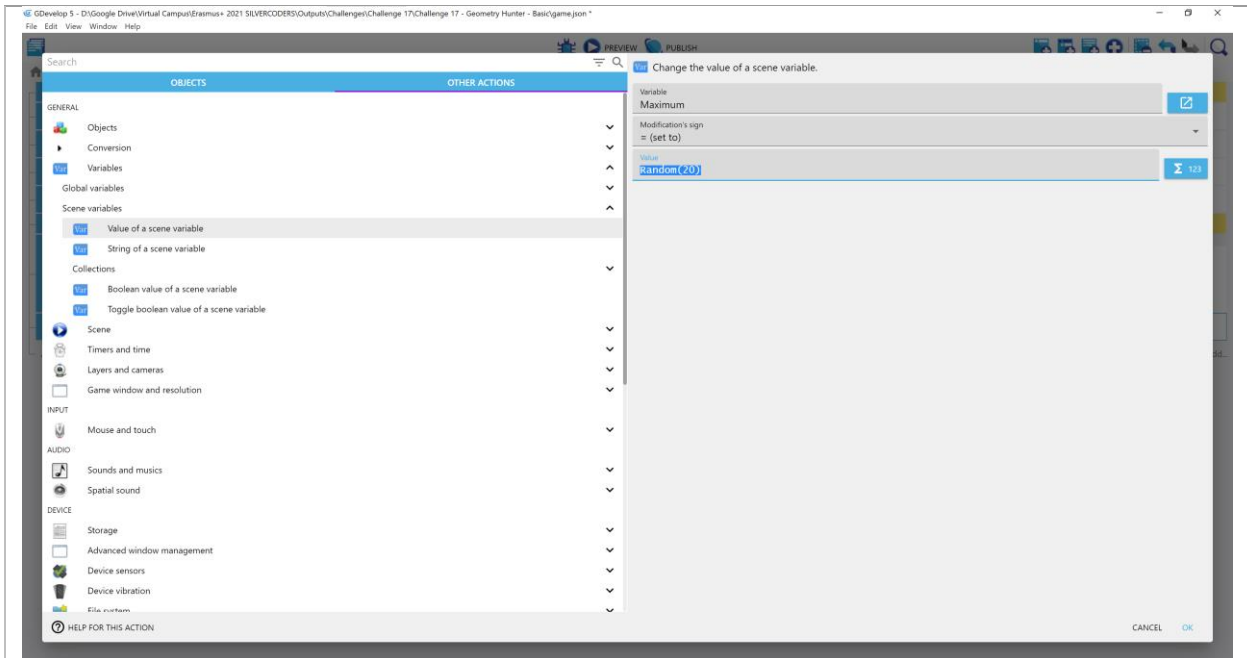
Then do **Apply**

On the events window do **Add a new event** and then press **Add condition**



Select **Other Conditions**, **Scene**, **At the beginning of a scene**, **Ok**

Select **Add Action**



Select **Other Actions**, **Variables**, **Scene Variables**, **Value of Scene Variable**

Variable = Maximum, **Modifications's sign = (add)**, **value = random (20)**

This means that randomly our game may have between 10 to 30 objects. Why?

We now have to change some of the events to reflect what we want. That means that every time we catch a shape we have to create another until we reach the maximum.

On the events sheet on the collision event add the following actions

Create object **Shapes** at position Random(600); Random(900) (layer:)

Change the scene variable **Maximum** : **subtract 1**

Add a new event

Add the condition **The scene variable Maximum = 0**

Add the action **Create object GameOver** at position 100;100 (layer:)

Add the action **Quit the game**

RESOURCES

Challenge 20 - Initial