

5. Enemies

Series

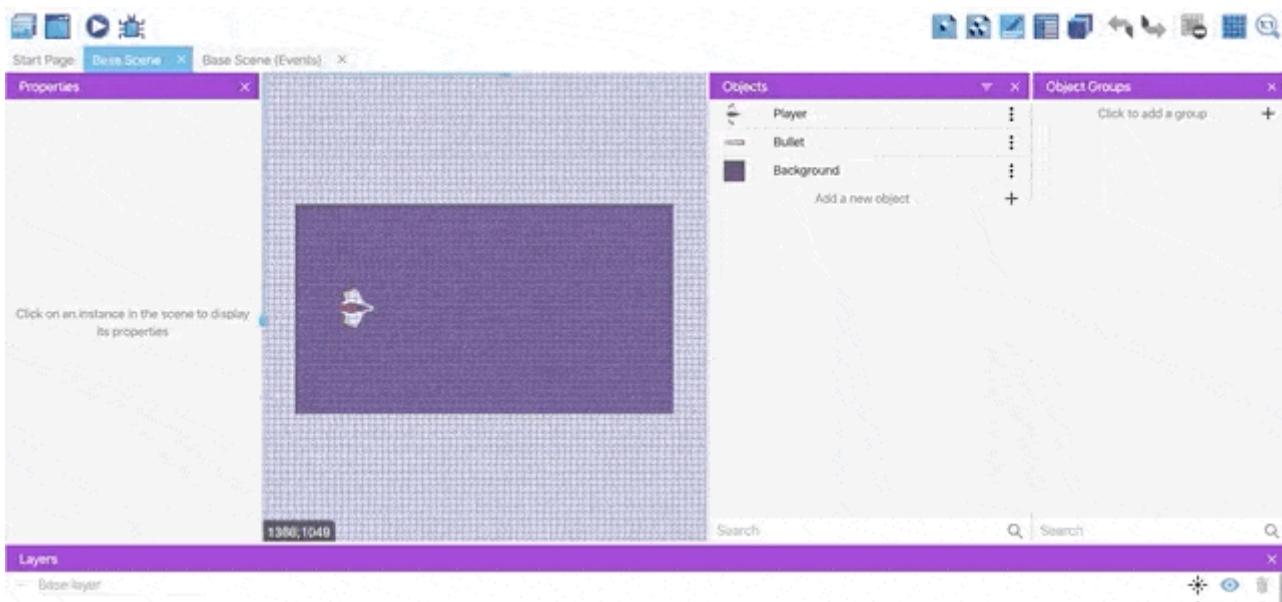
You are reading **Part 5** of the [Space Shooter Tutorial](#).

1. [Space Shooter, Part 1](#)
2. [Space Shooter, Part 2](#)
3. [Space Shooter, Part 3](#)
4. [Space Shooter, Part 4](#)
5. [Space Shooter, Part 5](#)
6. [Space Shooter, Part 6](#)
7. [Space Shooter, Part 7](#)
8. [Space Shooter, Part 8](#)
9. [Space Shooter, Part 9](#)
10. [Space Shooter, Part 10](#)
11. [Space Shooter, Part 11](#)
12. [Space Shooter, Part 12](#)
13. [Space Shooter, Part 13](#)

In this chapter, we are going to learn how to add enemies to our game.

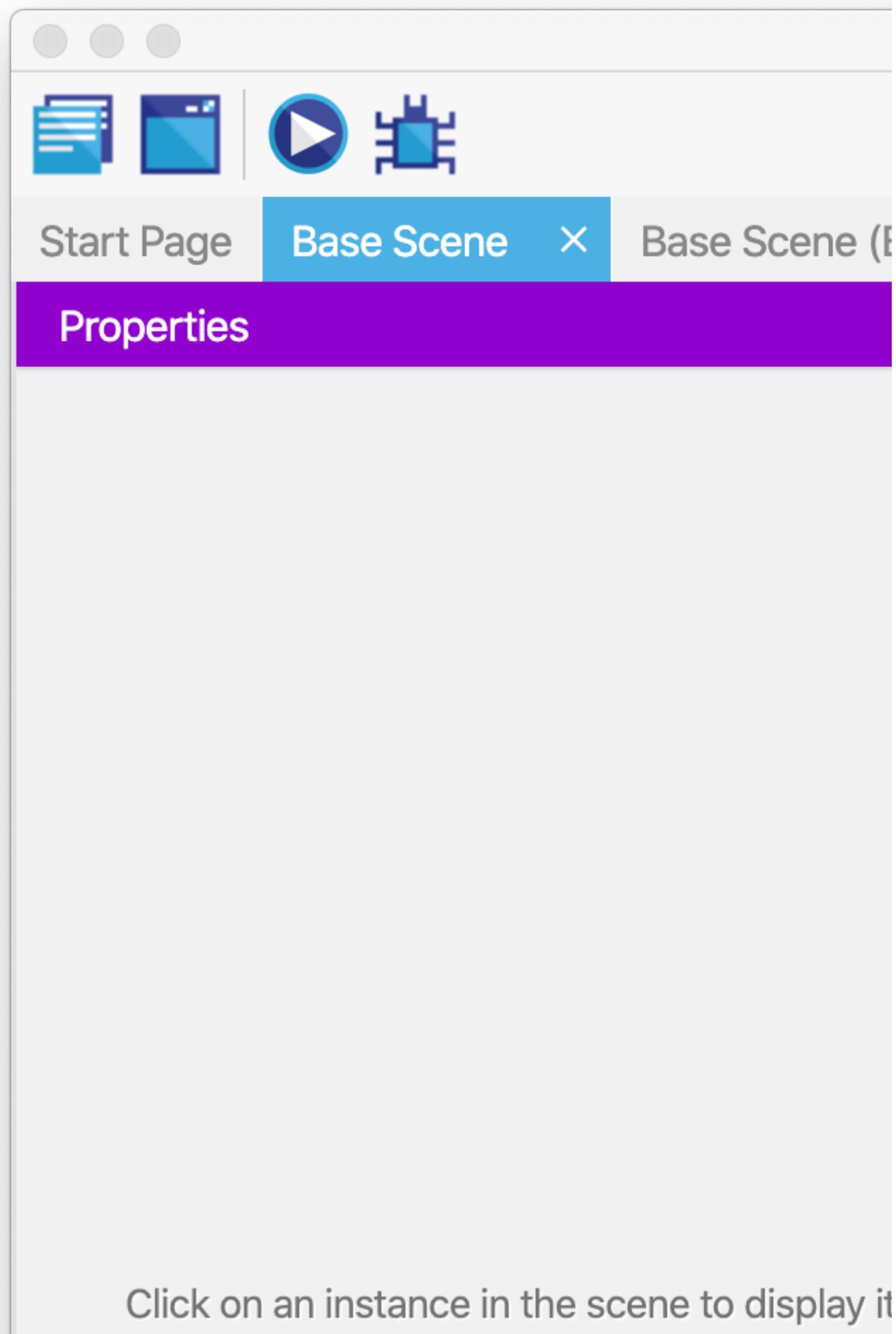
Creating enemies

We will add three different types of enemies. Hence, you should follow the same process to add enemies. Click on **Add a new object** and select **Sprite**. Then, type “**Enemy1**” in the name field and select “**enemy1.png**” from the asset folder as an image. Click apply.



Repeat this step to add all three enemies.

In the end, you should see:



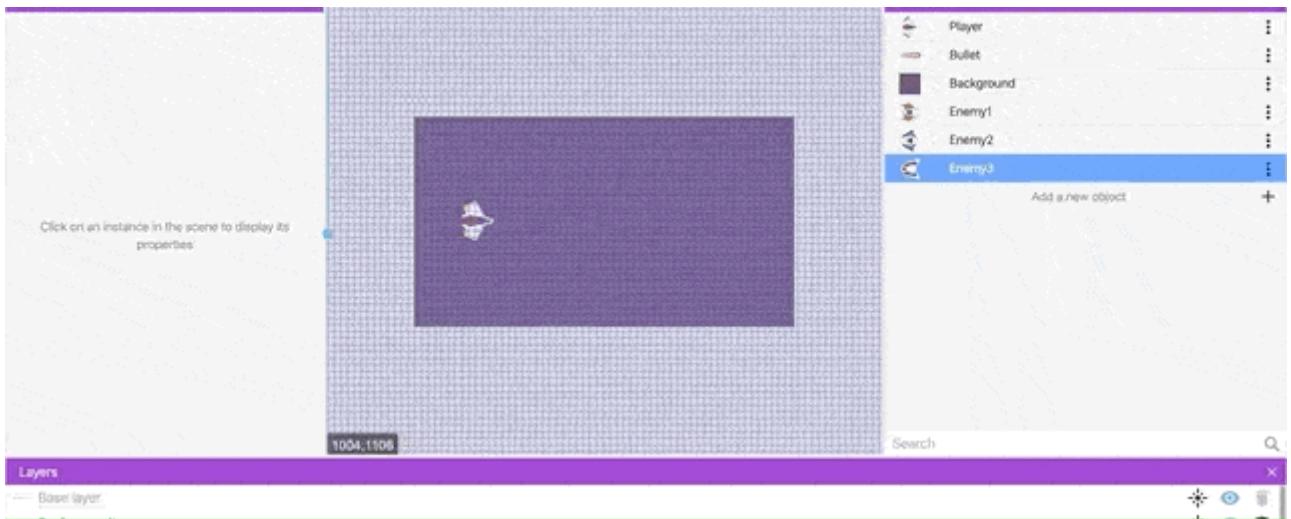
properties

We'll now create an object group called "**Enemies**". This will be useful in events, to refer to all the objects in the group without repeating events for each object.

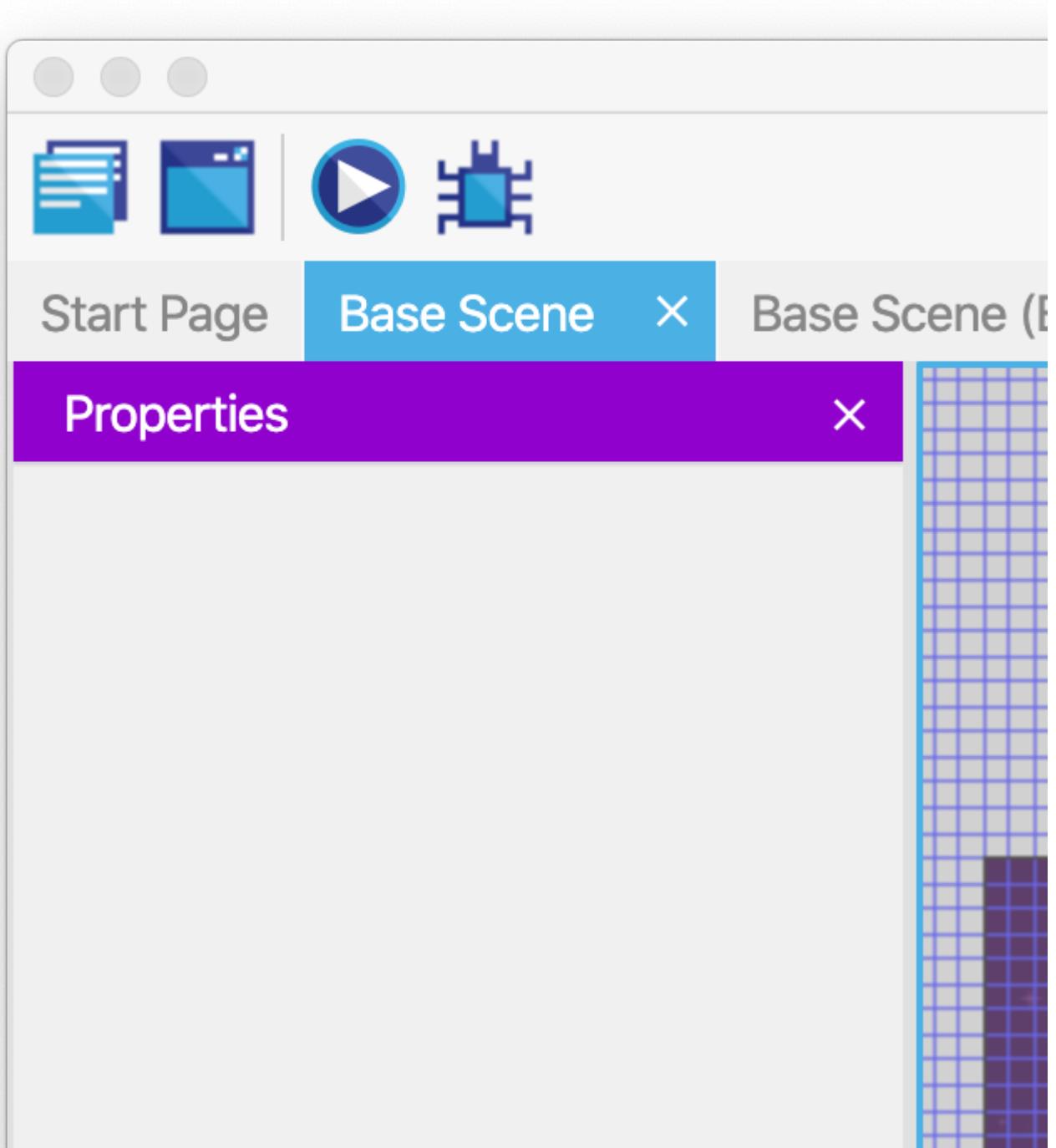
[Read this page](#) to learn more about object groups.

Click on the **object groups editor** button on the upper right. Then, click to **add a new group**, and add the enemy objects one by one.





Finally, rename the object group to “**Enemies**.”



Click on an instance in the scene to
display its properties

389;6

Layers

☰ Base layer

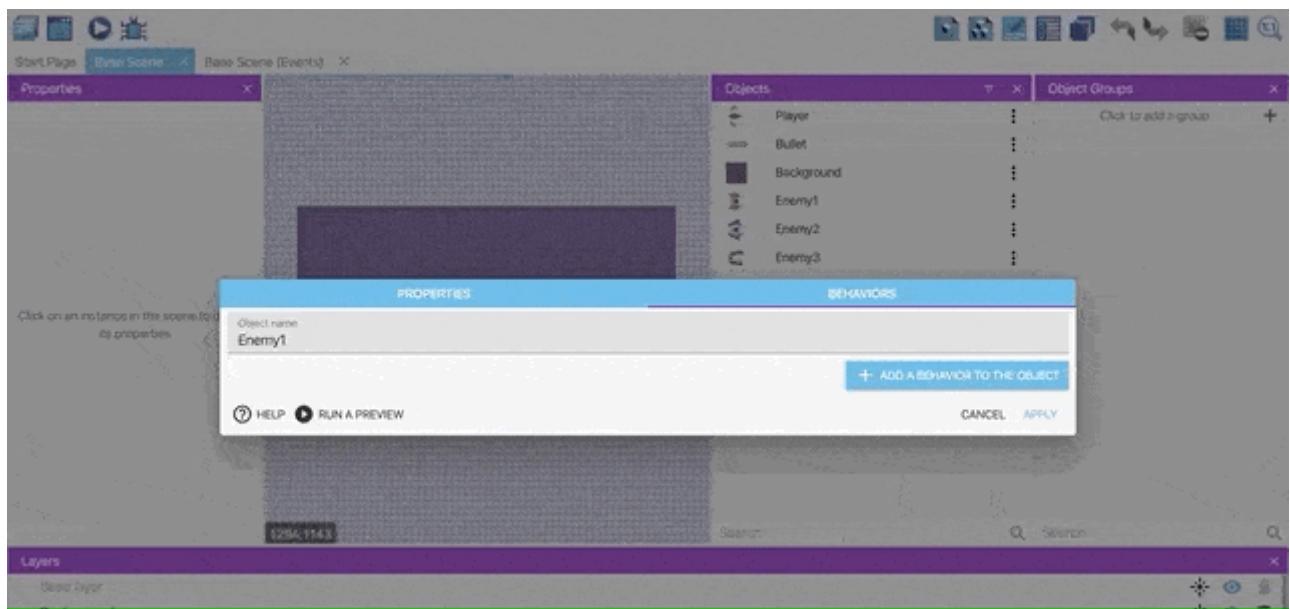
☰ **Background**

☰ Background color

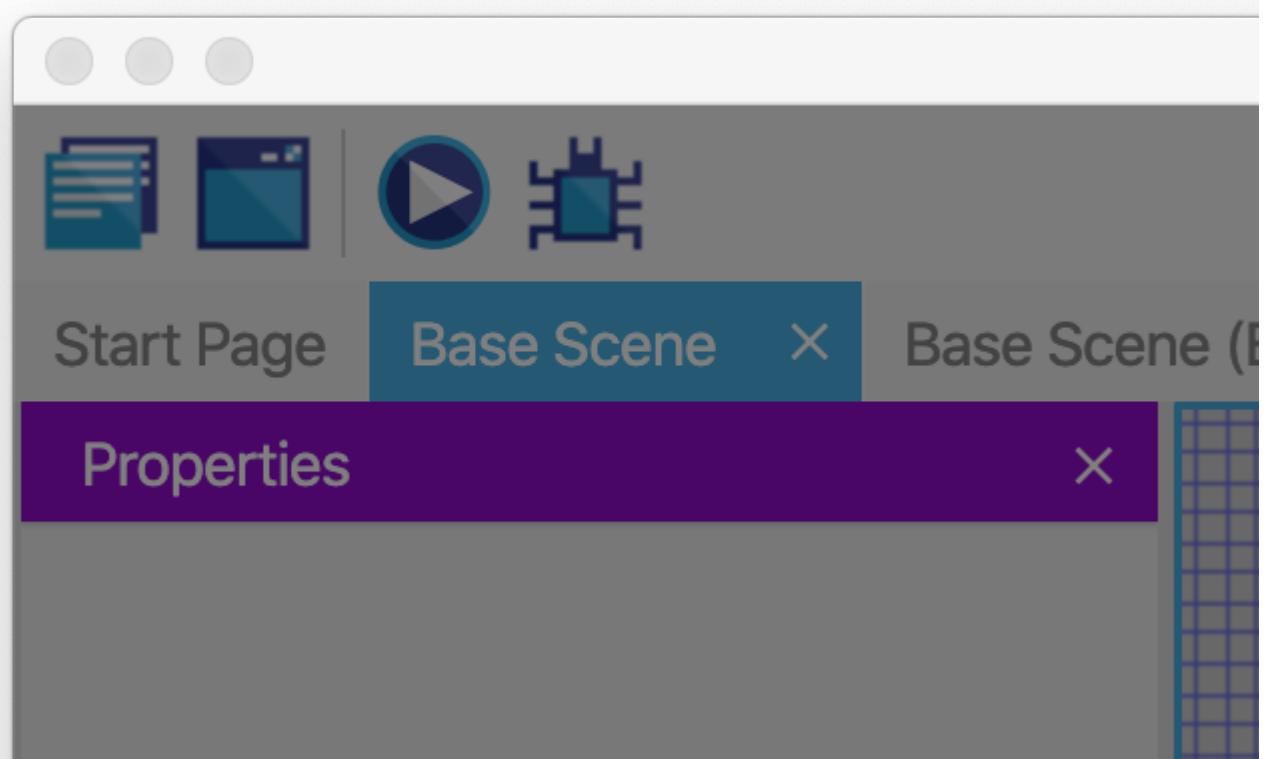
Giving health to enemies

We are using the same health behavior that we used for the player. Edit the **Enemy1** object and add the **Health** behavior.

Repeat this step for **every enemy object**.



Then, open the behavior page of **Enemy1** and leave it as is: **Health: 100**, **Maximum health: 100**



Object name

Enemy1

Behavior

Damage dealt

0

Health

100

Maximum health

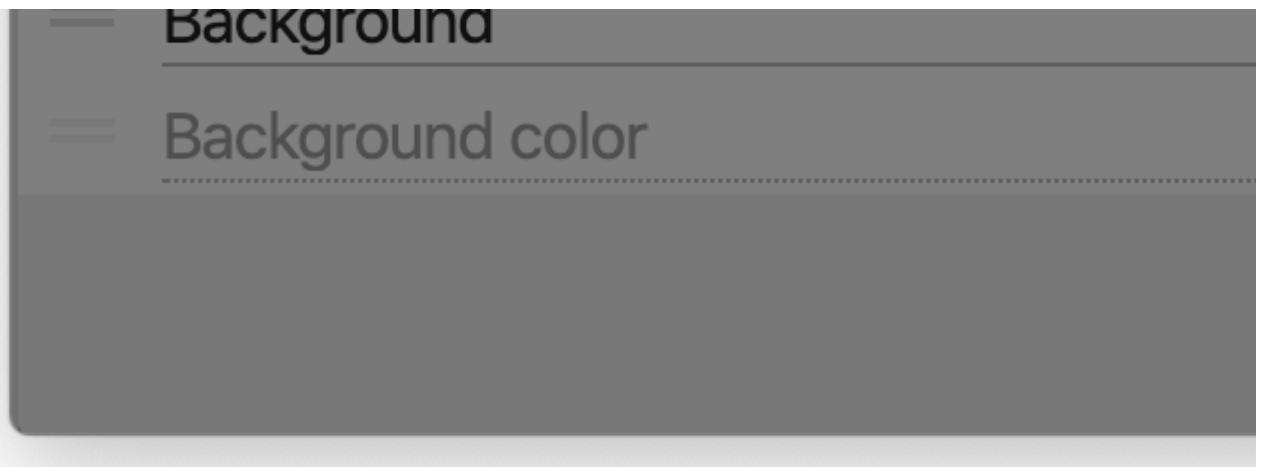
100

?

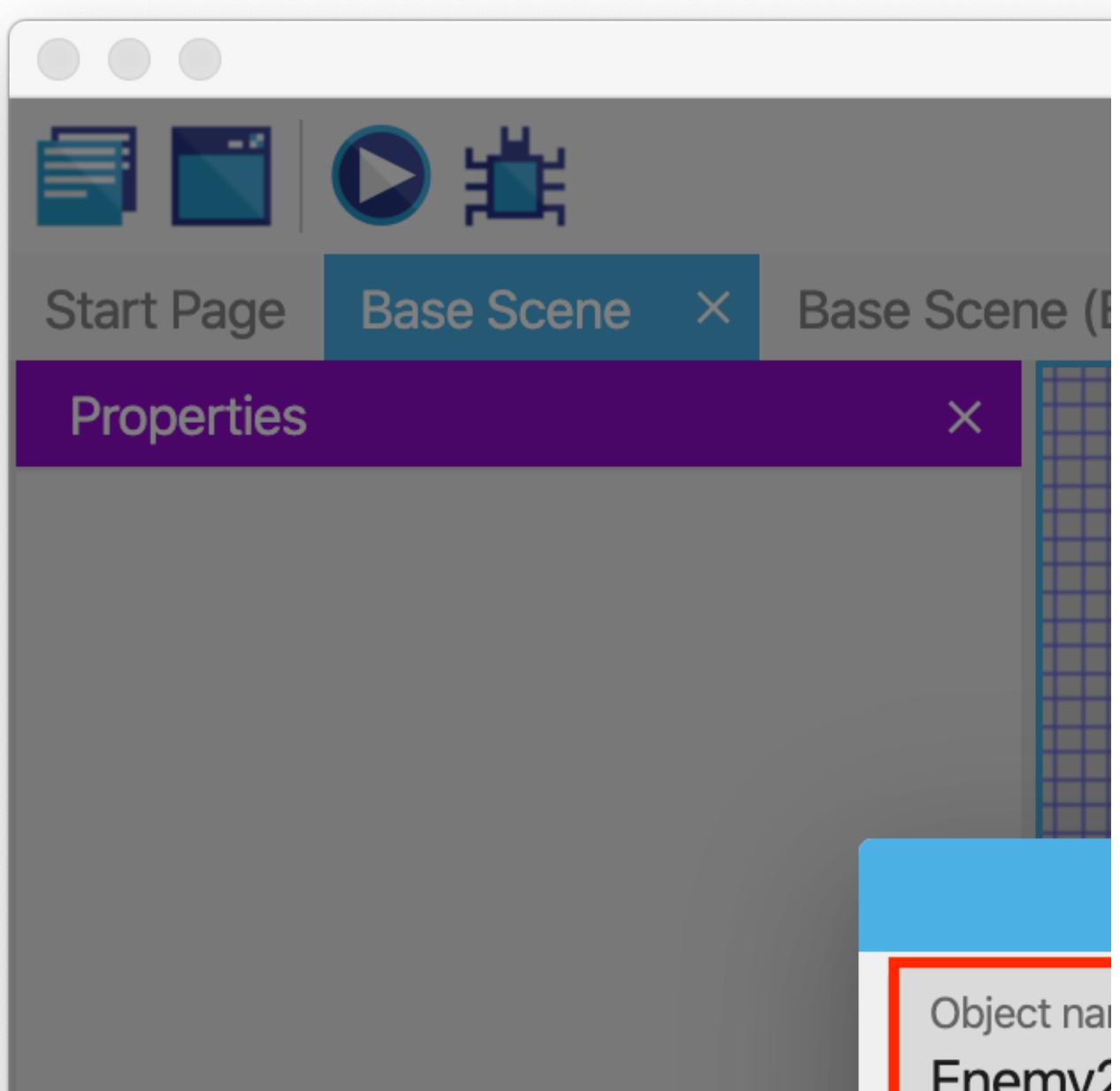
-17

Layers

Base layer



Open the behavior page of **Enemy2** and change **Health & Maximum Health** to **50**.



Behavior

Click on an instance in the scene to edit its properties

Damage capacity

0

Health

50

Maximum health

50

?

-1

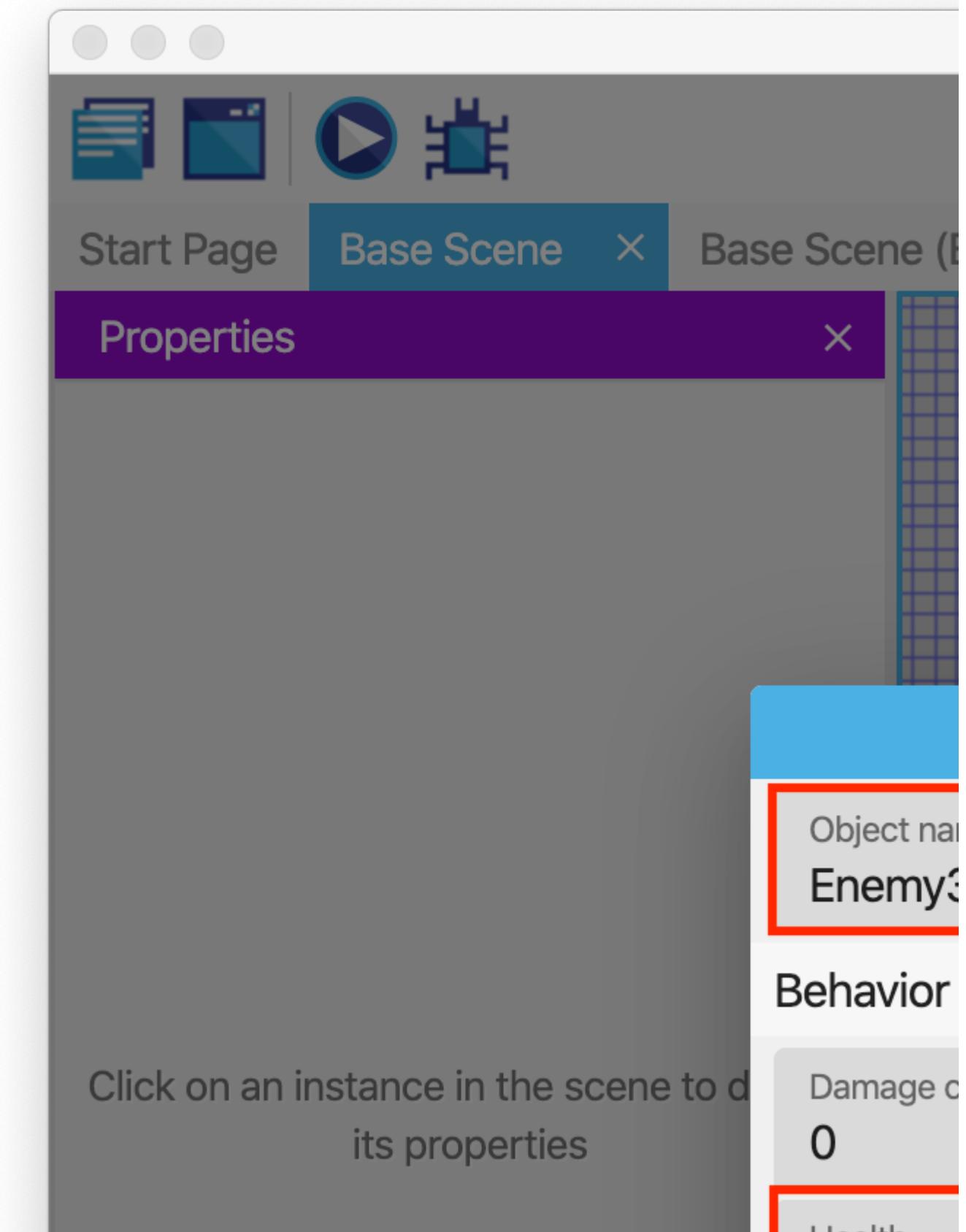
Layers

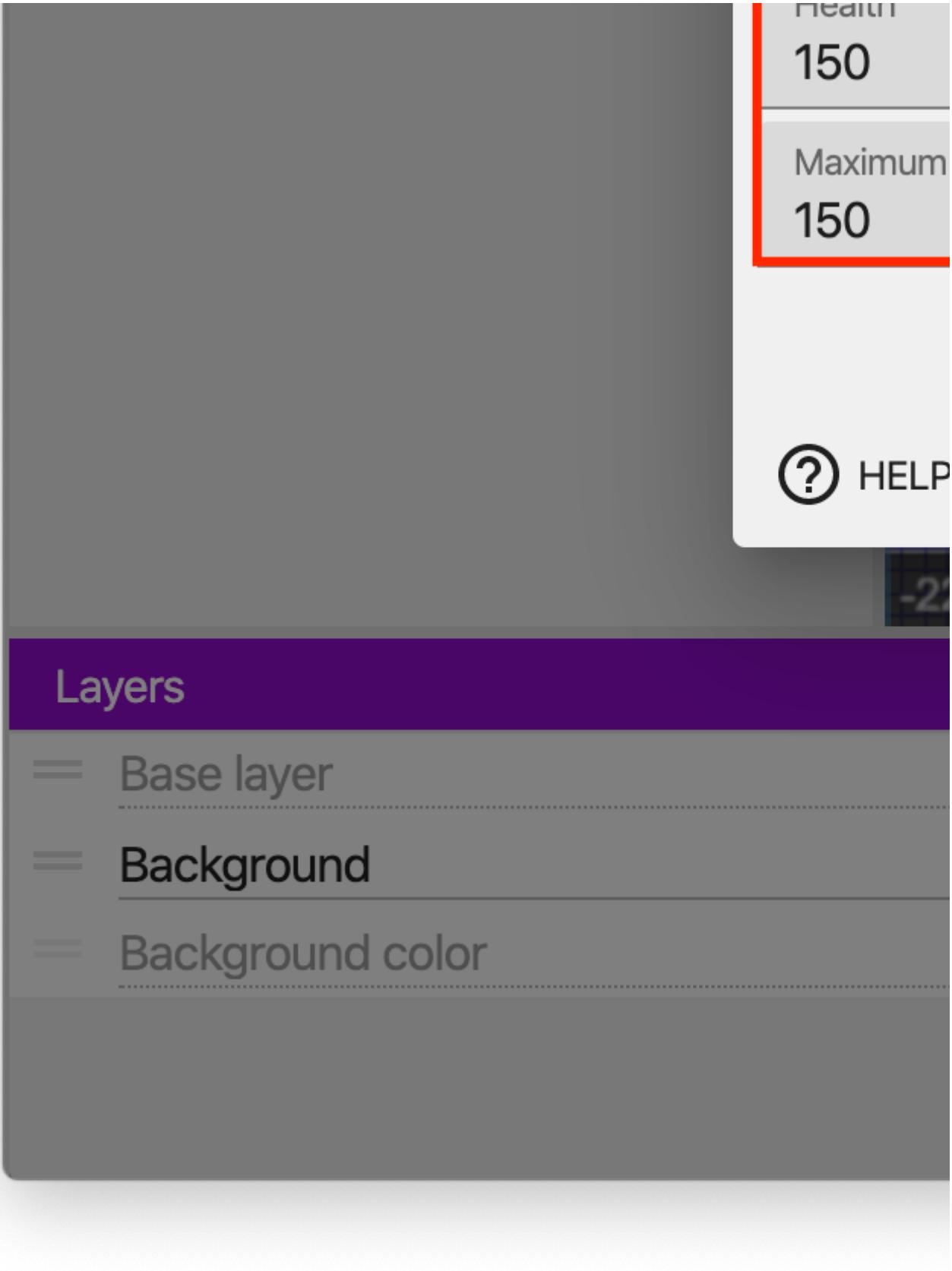
Base layer

Background

Background color

Open the behavior page of **Enemy3** and change **Health & Maximum Health** to **150**.





Shooting enemies

We will add a few actions to shoot the enemies. Firstly, open the **Events** tab. Then, add a new condition, choose the **Enemies** object group that we created before. Select action as **Collision** and the object as **Bullet**; by doing that, we can track the collision between our bullet objects and the enemies.



Start Page

Base Scene

Base Scene (B)

Controls

ene



Enemy1



Enemy2



Enemy3

OBJECT GROUPS



Enemies

NON-OBJECTS AND OTHER CONDITIONS



Value of a scene variable

Variables



Text of a scene variable

Variables



At the beginning of the scene

Scene



Scene just resumed

Scene



Value of a scene timer

Timers and time



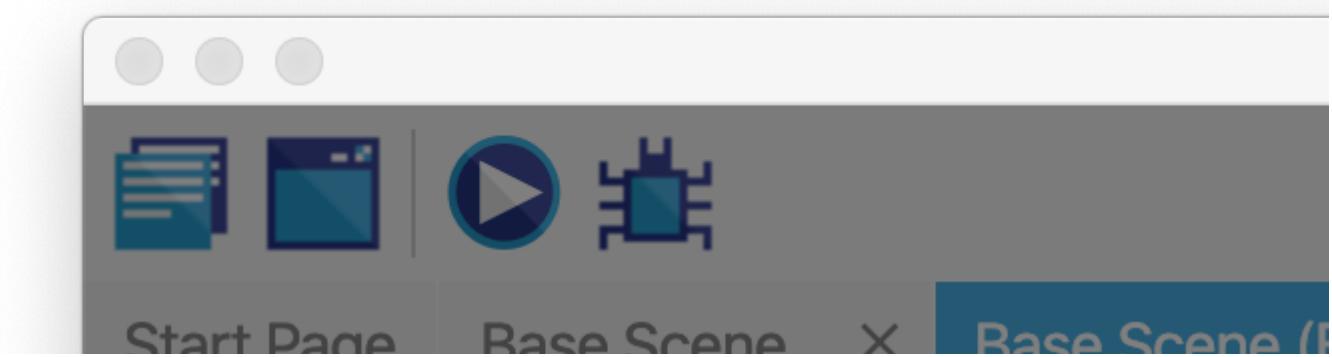
Scene timer paused

Timers and time



HELP FOR THIS CONDITION

If the collision happens between bullets and enemies, we need to remove the **Bullet** objects. Hence, we will add a new action next to the previous condition. Pick **Bullet** object and select **Delete an object**.



Start Stage

Base Scene

Base Scene 1

Controls

[A] Right key is pressed

Add condition

[A] Left key is pressed

Search

OBJECTS



Player



Bullet



Background



Enemy1



Enemy2



Enemy3

OBJECT GROUPS

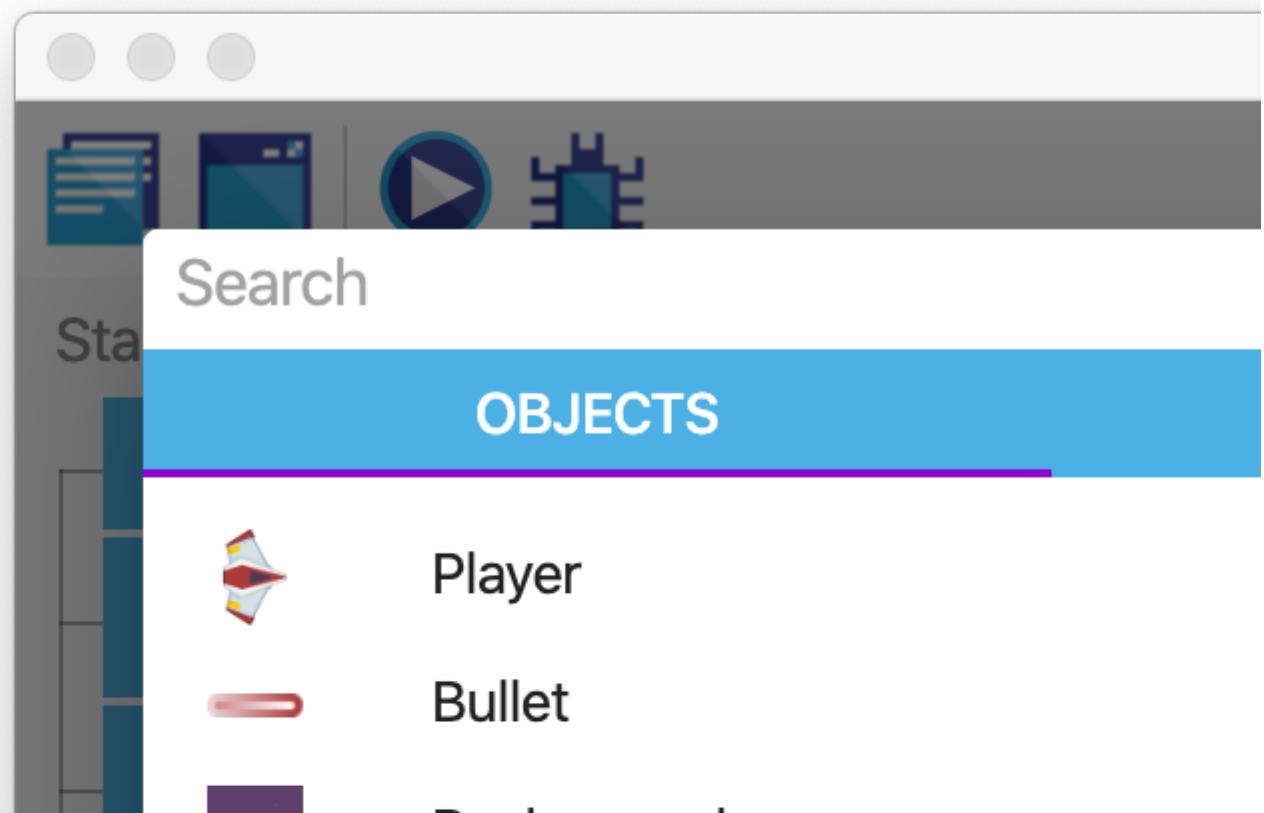


Enemies



HELP FOR THIS ACTION

Now, we need to give damage to the **Enemies**. Choose **Enemies** and select **Damage the object**. Then, enter **50** for the value of the damage.





Background



Enemy1



Enemy2



Enemy3

OBJECT GROUPS



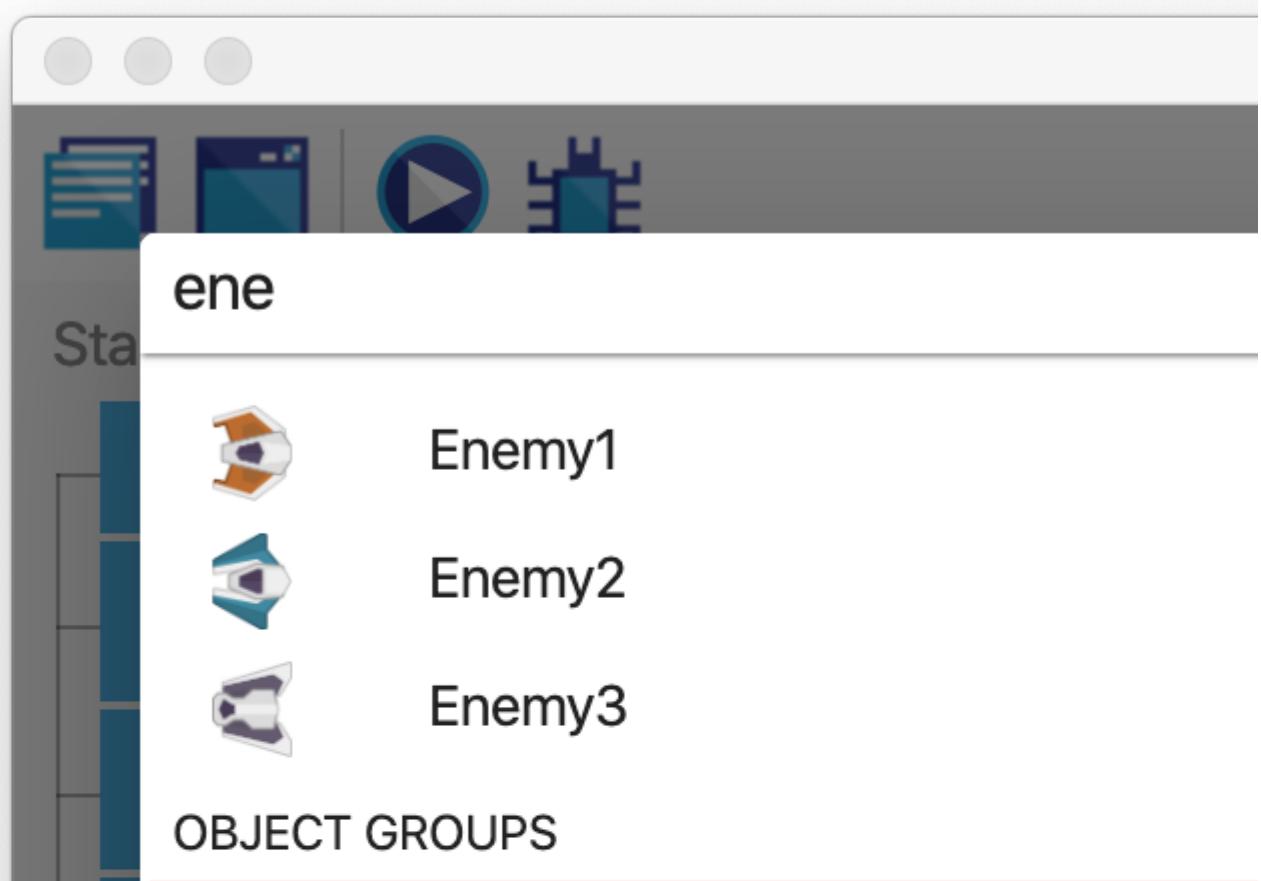
Enemies



You should see these actions:

A screenshot of the Scratch script editor. A blue vertical bar on the left indicates the script's position. The main area shows a blue event icon with a brown square and a grey rectangle, followed by the text "Enemies is in collision with Bullet". Below this, there is a grey "Add condition" button.

Then, we need to add an extra condition to determine whether the enemy is dead or not. We will add a new event under the last one. Select **Enemies** as objects and choose **Is dead**. Now we can track if the health of the enemy is dead or not.





Enemies

NON-OBJECTS AND OTHER CONDITIONS



Value of a scene variable
Variables



Text of a scene variable
Variables



At the beginning of the scene
Scene



Scene just resumed
Scene



Value of a scene timer
Timers and time

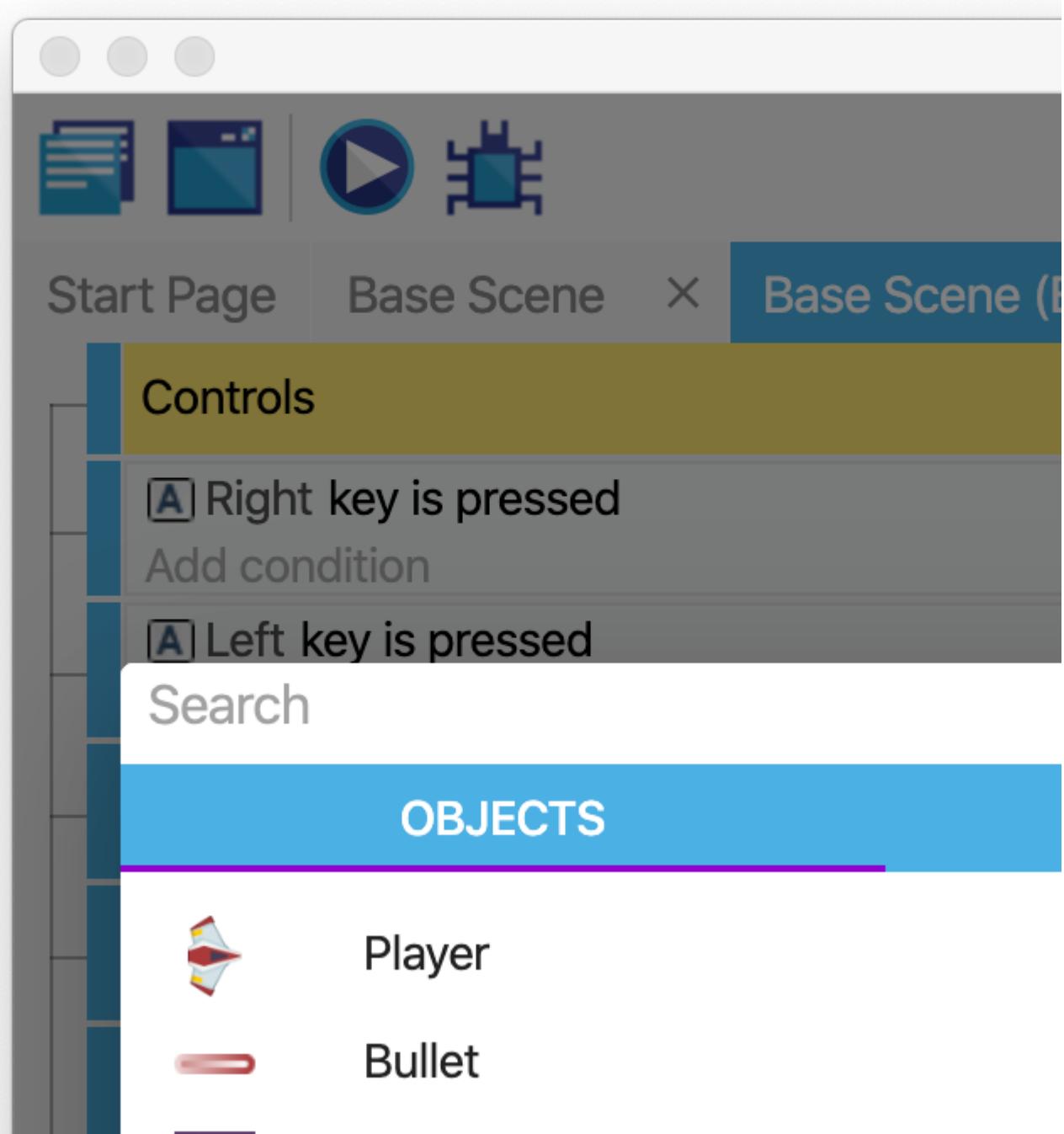


Scene timer paused
Timers and time



HELP

If the **enemy** is **dead**, we should **delete** the scene's object. We will add a new action to delete the object, just like in the previous steps.





Background



Enemy1



Enemy2



Enemy3

OBJECT GROUPS



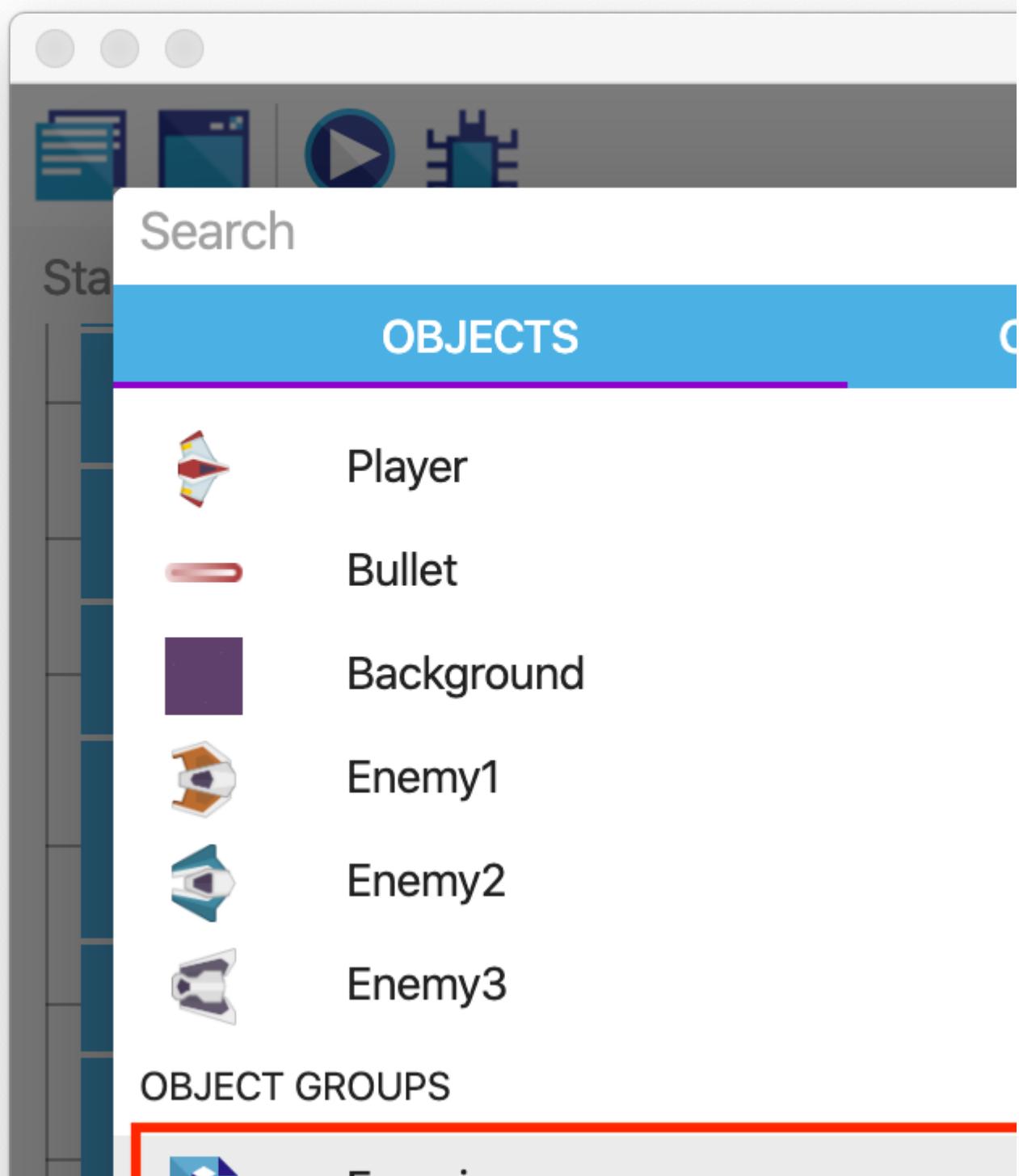
Enemies



HELP FOR THIS ACTION

Finally, we need to give **a force to move enemies**. However, we should provide a force when they are on the camera. Hence, we will add a new condition to perform our idea. In this condition, we are looking at the **Enemies' X position** when it is less than the **center Camera X position + 450**. Make sure that **less or equal to** is selected in the sign of the test.

You can ask why we are using **450** because it is **half of the screen width**. Remember, our **width** is set to **900**. If you use any other resolution, you need to change it.





Enemies

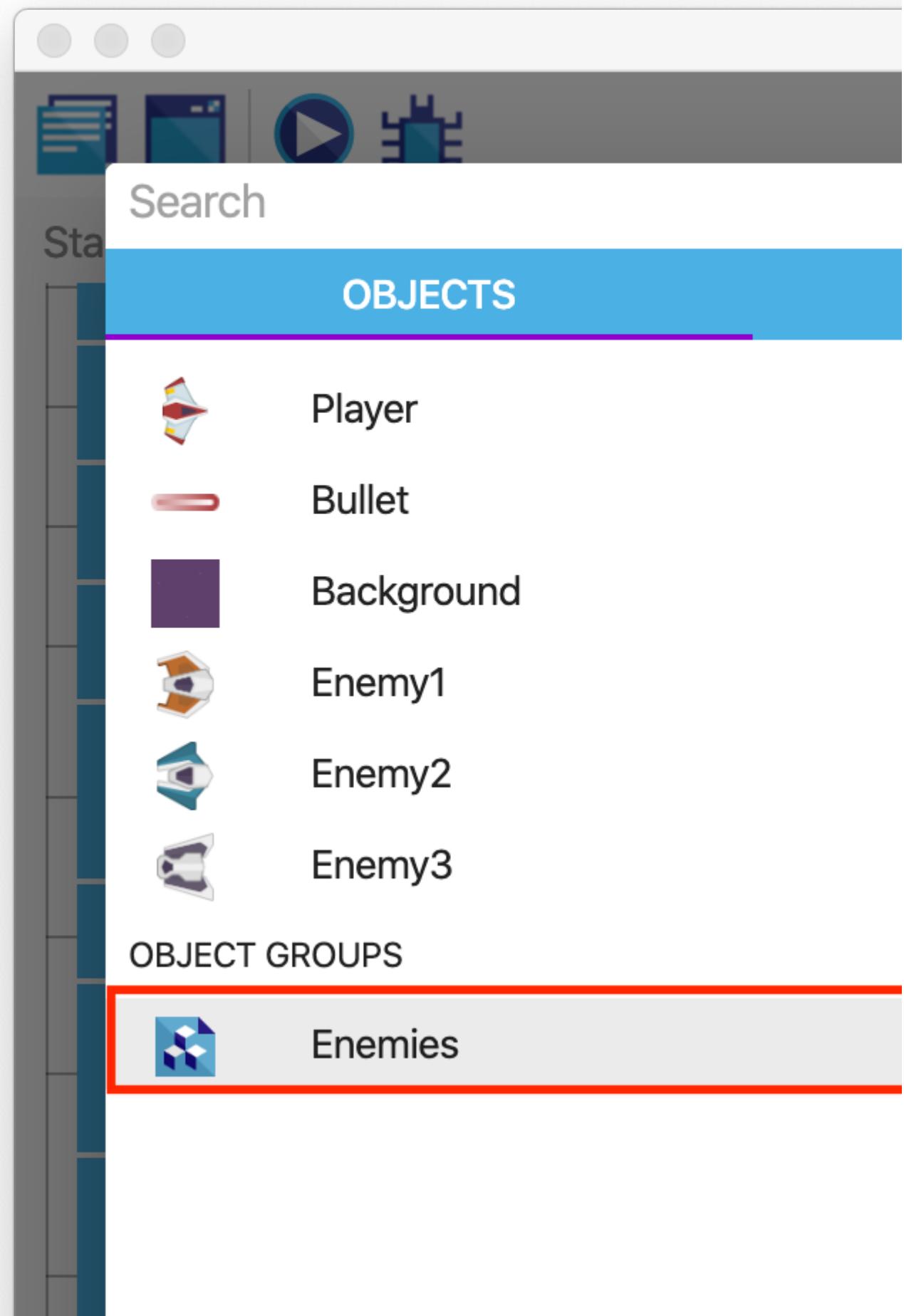


HELP FOR THIS CONDITION

Add a new event

In the event's action, we will give **a force to the Enemies** as we did in the previous

steps. Ensure that **the angle is 180**, **the speed is 200** pixels per second, and the **Instant** option is checked.





HELP FOR THIS ACTION

Add a new event

Now all the events are ready for shooting **Enemies**. You can add a comment on the top to classify these actions.

Enemies

Enemies is in collision with Bullet

Add condition

 **Enemies** is dead

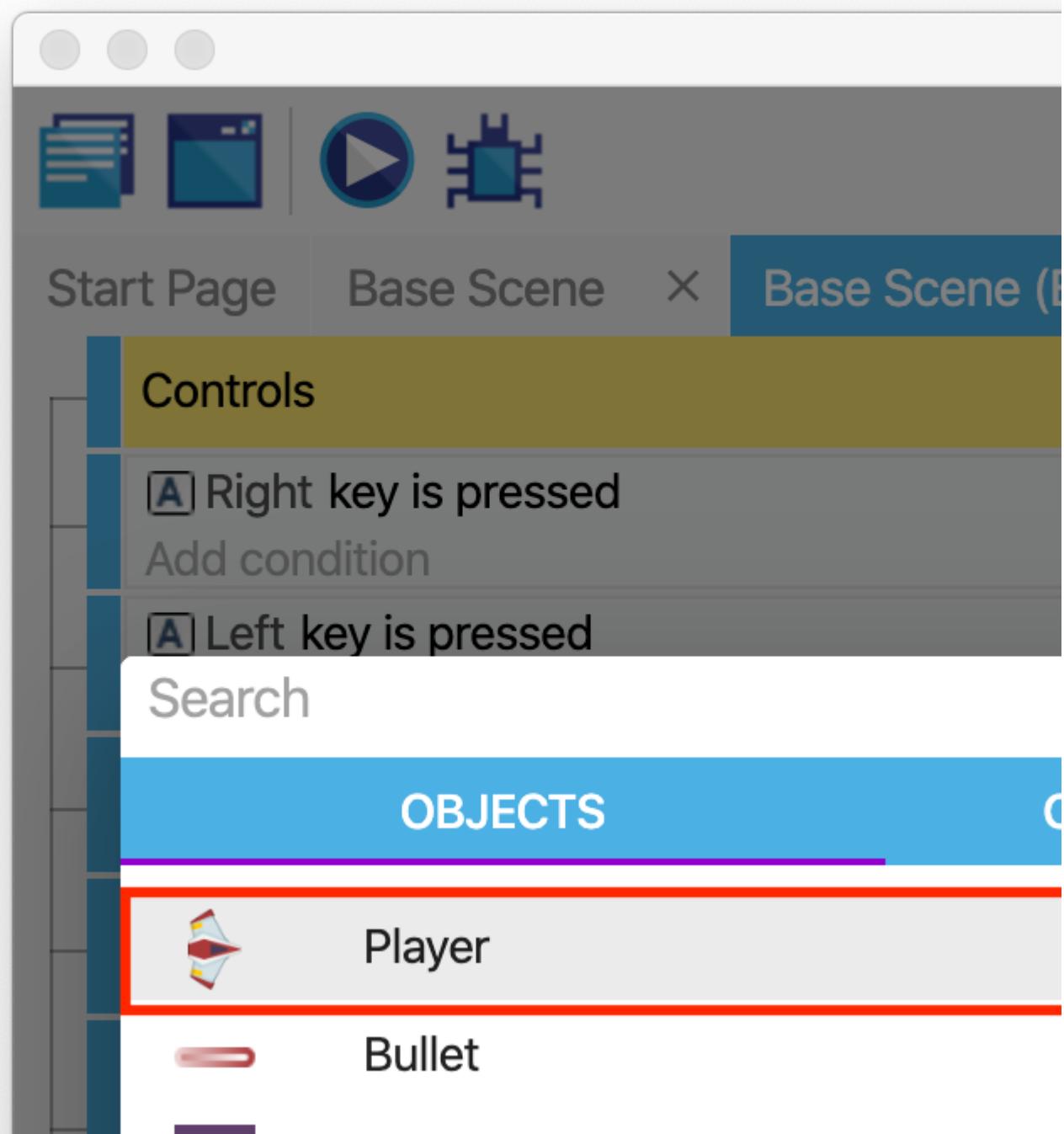
Add condition

The X position of **Enemies** \leq CameraX() + 450

Add condition

Damaging the player

We will start adding a new condition to determine the collision between the **player** and the **enemies**. Follow the same procedure in the above.





Background



Enemy1



Enemy2



Enemy3

OBJECT GROUPS



Enemies



HELP FOR THIS CONDITION



Enemies is dead

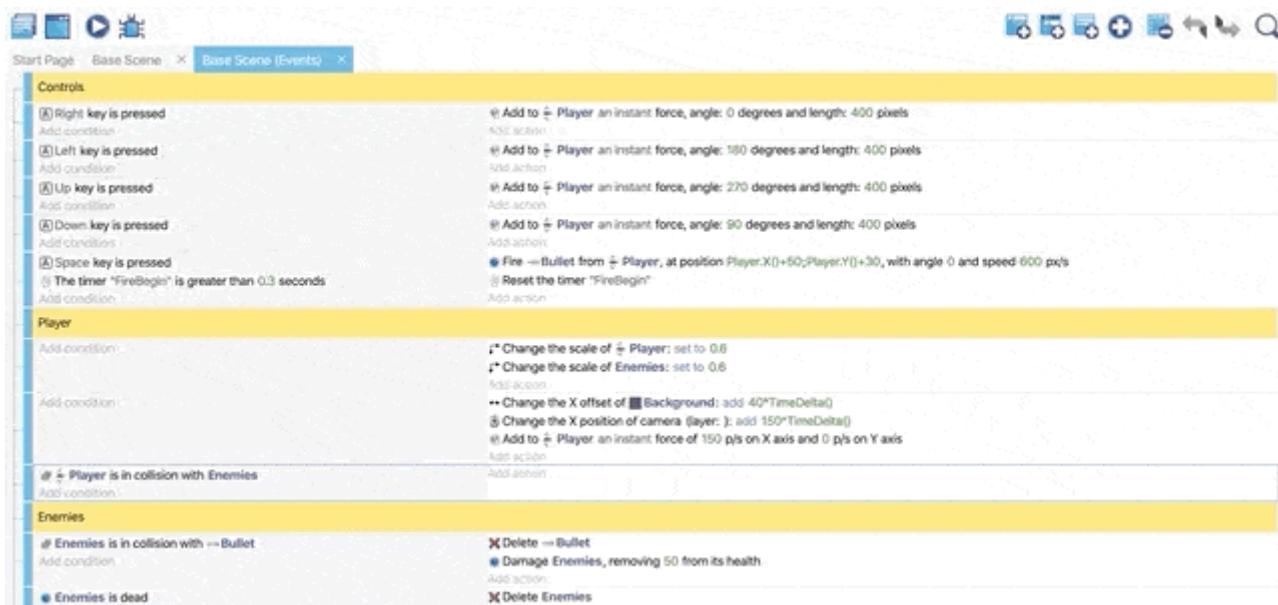
Add condition

Add a new event

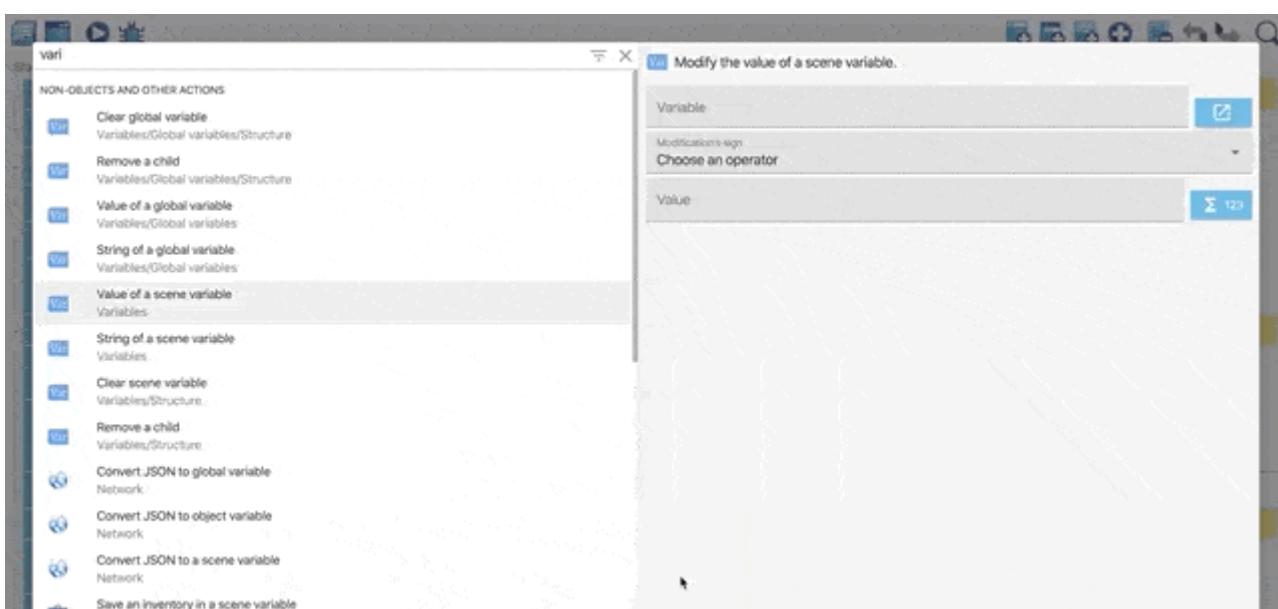
This time, we do not directly give the player damage because there will be different scenarios to damage the player. Hence, we are going to use the **Scene variable**.

Scene variables are useful because they are created at the beginning of the scene, and they are going to be destroyed when the scene is over.

Now, create an action and type “**variable**” on the search bar. Then, select **Value of a scene variable** and click on the button next to the action details. In this window, create a scene variable called “**IsDamaged**” and give its initial value **0**. Then click **apply**.



Then, select the **IsDamaged** as the variable, and change the scene variable's value to **40**.



Add the **delete action for the enemies** like in the above. Then, your actions look like in the image below.





Player IS IN COLLISION WITH ENEMIES

Add condition

Add a new condition to check whether **IsDamaged is greater than or equal to 1**. If the condition is fulfilled, then we will give damage to the player.

The image shows a Scratch interface. On the left, there's a vertical stack of blocks labeled "Stage" and "Backgrounds". In the center, there's a "Control" category tab at the top, followed by a "Variables" category tab. Below these are several other categories: "Motion", "Sound", "Operators", "Looks", and "Science". A condition block labeled "val" is currently selected. To the right of the stage, there's a list of conditions under the heading "NON-OBJECTS AND OTHER CONDITIONS". One condition, "Value of a scene variable" under "Variables", is highlighted with a red border.

val

NON-OBJECTS AND OTHER CONDITIONS

- Value of a global variable
Variables/Global variables
- Value of a scene variable
Variables
- Value of a scene timer
Timers and time
- Compare the value of acceleration
Sensors/Motion
- Compare the value of acceleration
Sensors/Motion
- Compare the value of acceleration
Sensors/Motion



Sensors/Motion



Compare the value of orientation a
Sensors/Orientation



Compare the value of orientation k
Sensors/Orientation



Compare the value of orientation g
Sensors/Orientation



Compare the value of rotation alp
Sensors/Motion



Compare the value of rotation bet
Sensors/Motion



Compare the value of rotation gan
Sensors/Motion

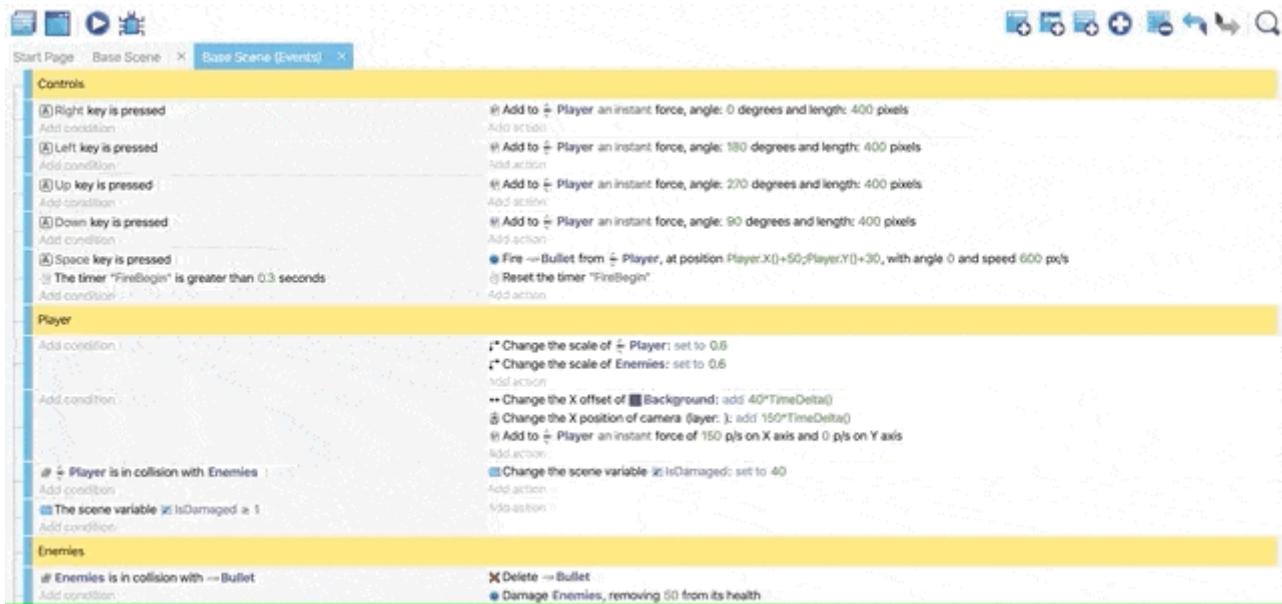


HELP FOR THIS CONDITION

Then, add a new action to damage the player. We did the same above, but this time enter the value of “**IsDamaged.**”

To enter the value of a variable, click the **number icon** next to the entered area. Afterward, choose **Variables** and the **Value of scene**

variable. There will be a new window where you can select our variables.



We also need to reset the **IsDamaged** to 0 because this will prevent our player from getting extra damages. Add a new action under the previous one and set **IsDamaged** to 0.

The screenshot shows the Construct 3 visual editor interface with the following elements:

- Scene:** The main workspace area.
- Variables:** A sidebar on the left containing lists of variables categorized by type (e.g., Number, String, Boolean).
- NON-OBJECTS AND OTHER ACTIONS:** A section header.
- Value of a scene variable** (highlighted with a red box): An action block with a "Var" icon.
- String of a scene variable**: Another action block with a "Var" icon.
- Clear scene variable**: A third action block with a "Var" icon.

-  Variables/Structure
-  Stop and go back to previous screen
-  Scene
-  Pause and start a new scene
-  Scene
-  Quit the game
-  Scene
-  Change the scene
-  Scene
-  Change background color
-  Scene
-  Pause a scene timer
-  Timers and time
-  Delete a scene timer
-  Timers and time
- Start (or reset) a scene timer
- Timers and time
- Unpause a scene timer
- Timers and time

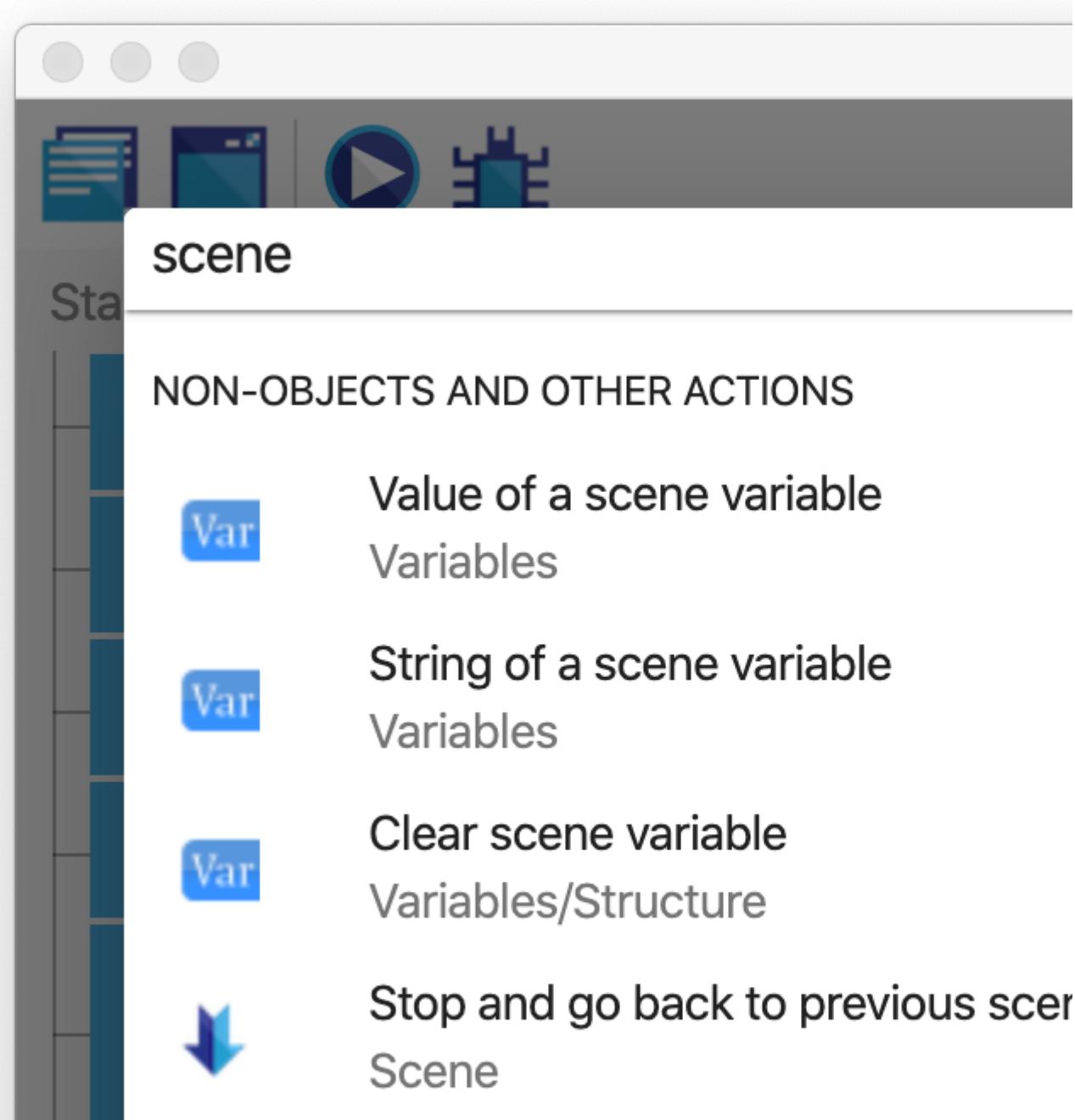


HELP FOR THIS ACTION

Add a new event

Restarting the game

Our game is over when the player is dead. Add a new condition to **check the player is dead or not**. Then, add an action to restart the game. Type “**scene**” on the search bar and select **Change the scene**. Just select the “**Base Scene**” or your scene’s name.





Pause and start a new scene
Scene



Quit the game
Scene



Change the scene
Scene



Change background color
Scene



Pause a scene timer
Timers and time



Delete a scene timer
Timers and time



Start (or reset) a scene timer
Timers and time



Unpause a scene timer
Timers and time



HELP FOR THIS ACTION

Add condition

Add a new comment

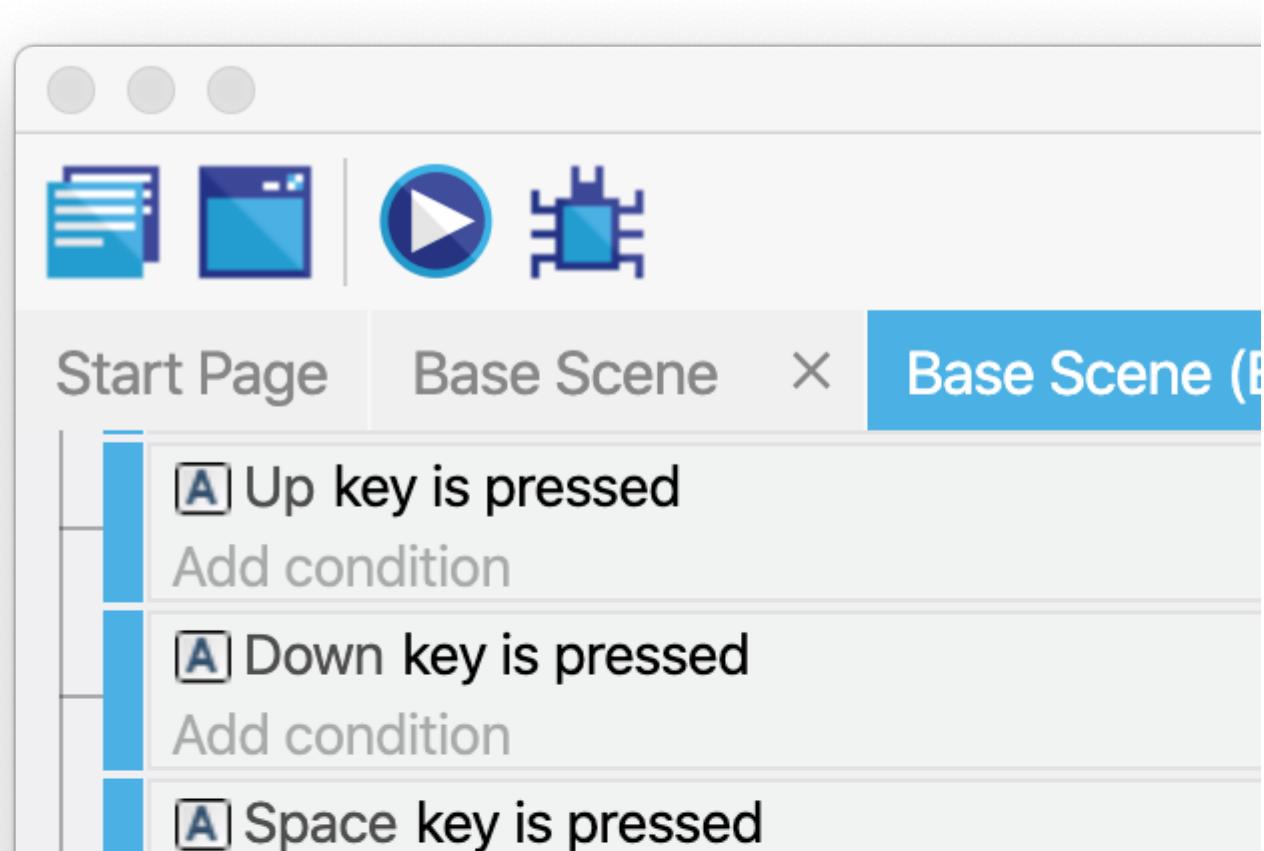
For this chapter, the player's actions are ready.

Small enemies

We already made the player smaller, but we did not make it for the enemies. Please copy and paste the same action for making it small and change the **Player** with **Enemies**.



This should be the **final version of the events**.



 The timer "FireBegin" is greater than 0.3 s

Add condition

Add condition

Add condition

Player

  Player is in collision with Enemies

Add condition

 The scene variable  IsDamaged ≥ 1

Add condition

  Player is dead

Add condition

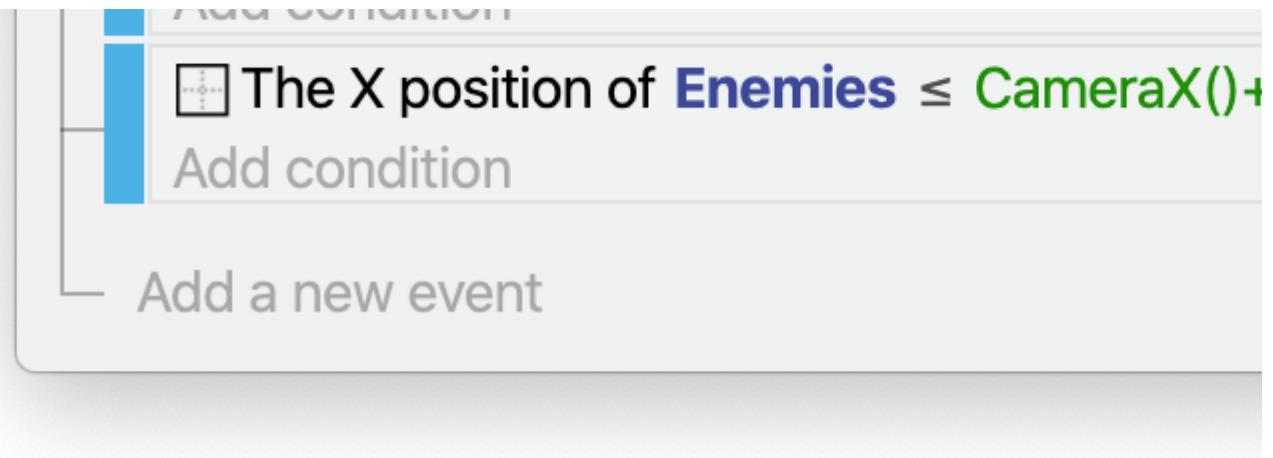
Enemies

 Enemies is in collision with  Bullet

Add condition

 Enemies is dead

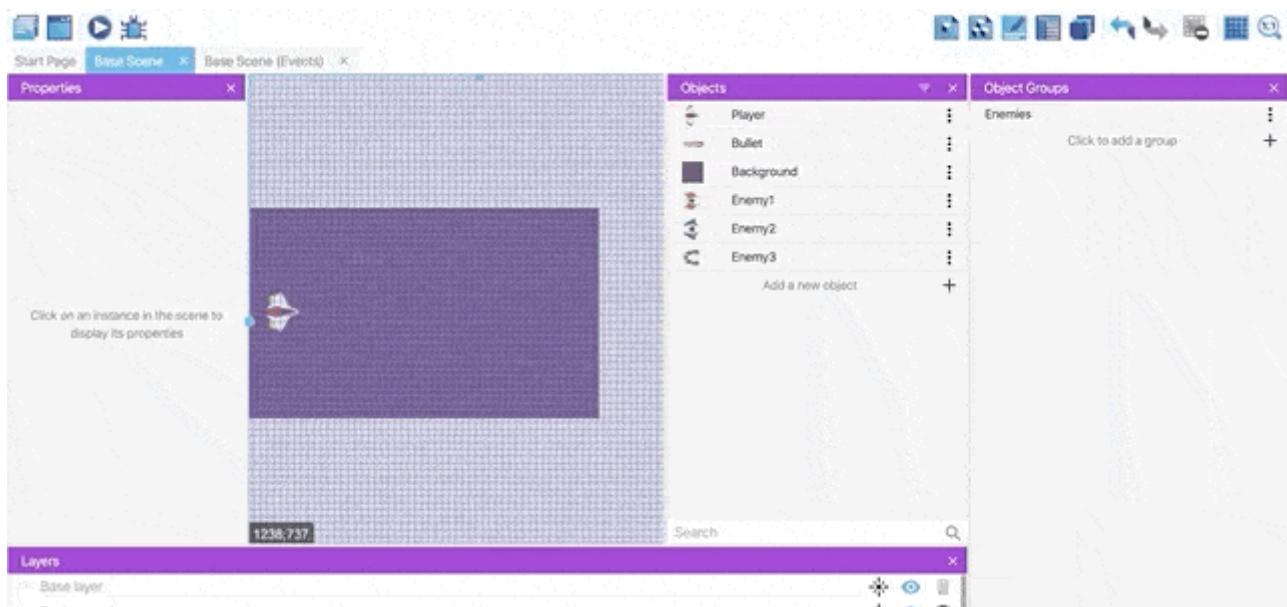
Add condition



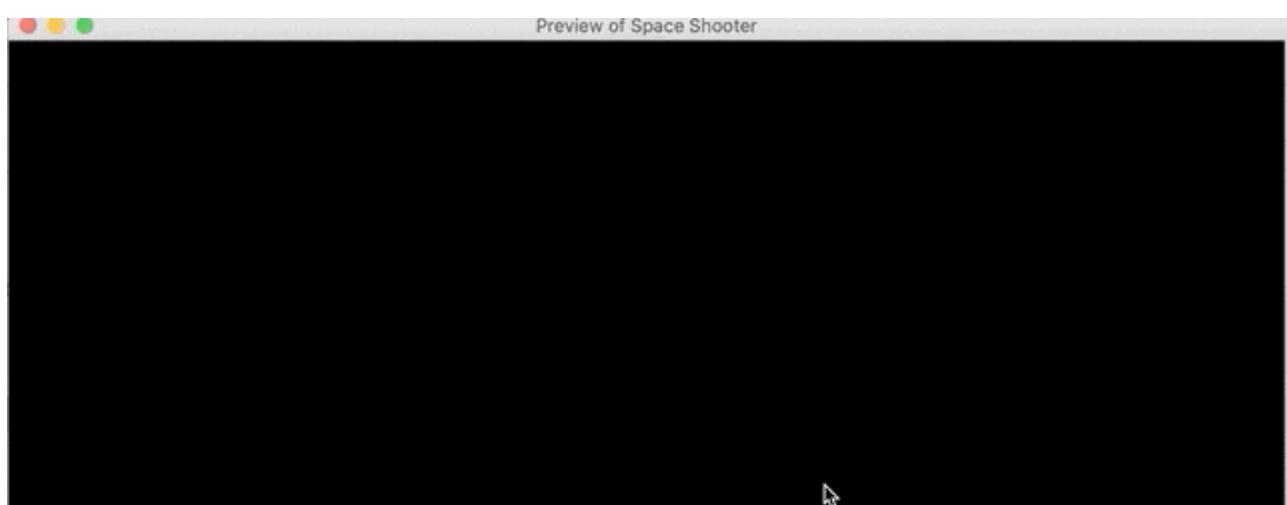
Testing out new mechanics

Before testing out new mechanics, we need to put some enemies on the scene.

Drag some enemies and drop them at different places where horizontally the same level as the scene.



Now, we are ready to play our game. We can shoot the enemies, and the player can get damage due to the enemy's collision.





Next step

Space Shooter, Part 6