

# Platformer Tutorial, Part 2: Create a Playable Character

This part of the tutorial explains how to add a playable character to the game.

You'll learn how to:

- Create an object that represents the playable character.
- Let the player control the object using the arrow keys.
- Follow the playable character with the camera.

## Series

You are reading **Part 2** of the [Platformer Tutorial](#).

1. [Platformer Tutorial, Part 1](#)
2. Platformer Tutorial, Part 2
3. [Platformer Tutorial, Part 3](#)
4. [Platformer Tutorial, Part 4](#)
5. [Platformer Tutorial, Part 5](#)
6. [Platformer Tutorial, Part 6](#)
7. [Platformer Tutorial, Part 7](#)
8. [Platformer Tutorial, Part 8](#)

## Step 1: Create an object

The first step is to create an object that represents the playable character. If you followed the first part of this tutorial, this should be a familiar process.

To create an object for the playable character:

1. Create a **Sprite** object named “Player”.
2. Use the “p1\_stand.png” asset as the animation's only frame.
3. Assign a name of “Idle” to the animation.
4. Drag an instance of the object into the scene.

## PROPERTIES

Object name

Player

≡ Animation #0 Idle



+ ADD

EDIT HITBOXES

EDIT POINTS



HELP



RUN A PREVIEW

517;623

[Platformer Tutorial, Part 4](#) explains how to animate the character.

## Step 2: Add a behavior to the object

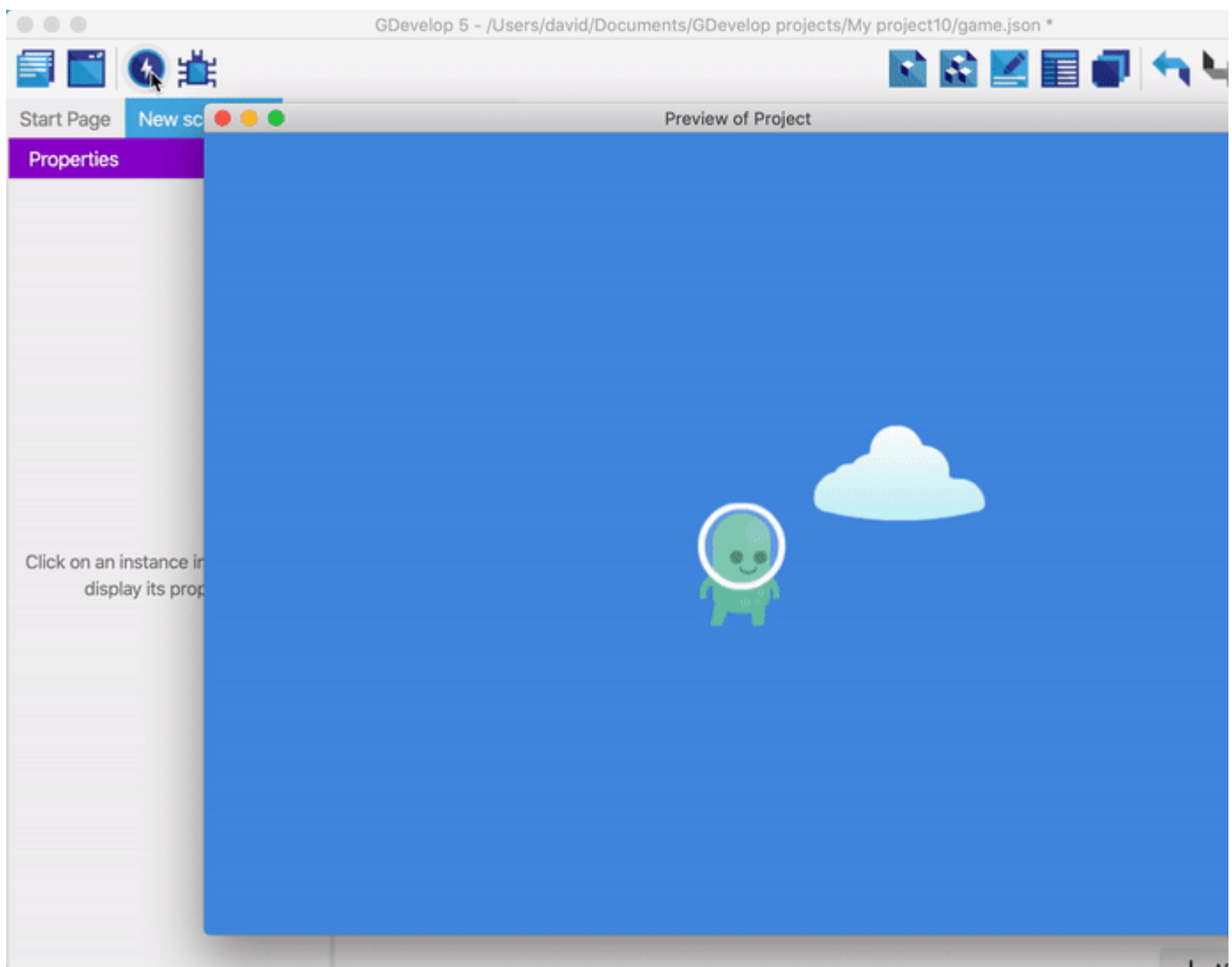
If you preview the game, you can't control the "Player" object. This is because, by default, objects don't do anything. To make objects do something, you need to assign one or more [behaviors](#) to them.

GDevelop comes with a large library of behaviors that you can add to objects in a matter of clicks. One of these behaviors is the **Platformer character** behavior, which lets you control an object as a playable character.

To add the **Platformer character** behavior to the object:

1. Right click the **Player** object.
2. Select **Edit object**.
3. Switch to the **Behaviors** tab.
4. Click **Add a behavior to the object**.
5. Select **Platformer character**.
6. Click **Apply**.

If you preview the game, you can control the player with the arrow keys. There's a side-effect though: an object with the **Platformer character** behavior is affected by gravity, so in this case, the playable character quickly fall out of the frame.



## Step 3: Follow an object with the camera

To prevent the character from falling out of frame, the game needs platforms and a camera that follows the player. The [next part of the tutorial](#) is about platforms, so this section is about the camera.

To make the camera follow the player, you need to create an [event](#). In GDevelop,

events let you define cause and effect, so when something happens (cause) you can trigger something else to happen (effect).

Events are made up of two parts: conditions and actions. The condition is what determines when the event happens and the action is what happens when the event occurs.

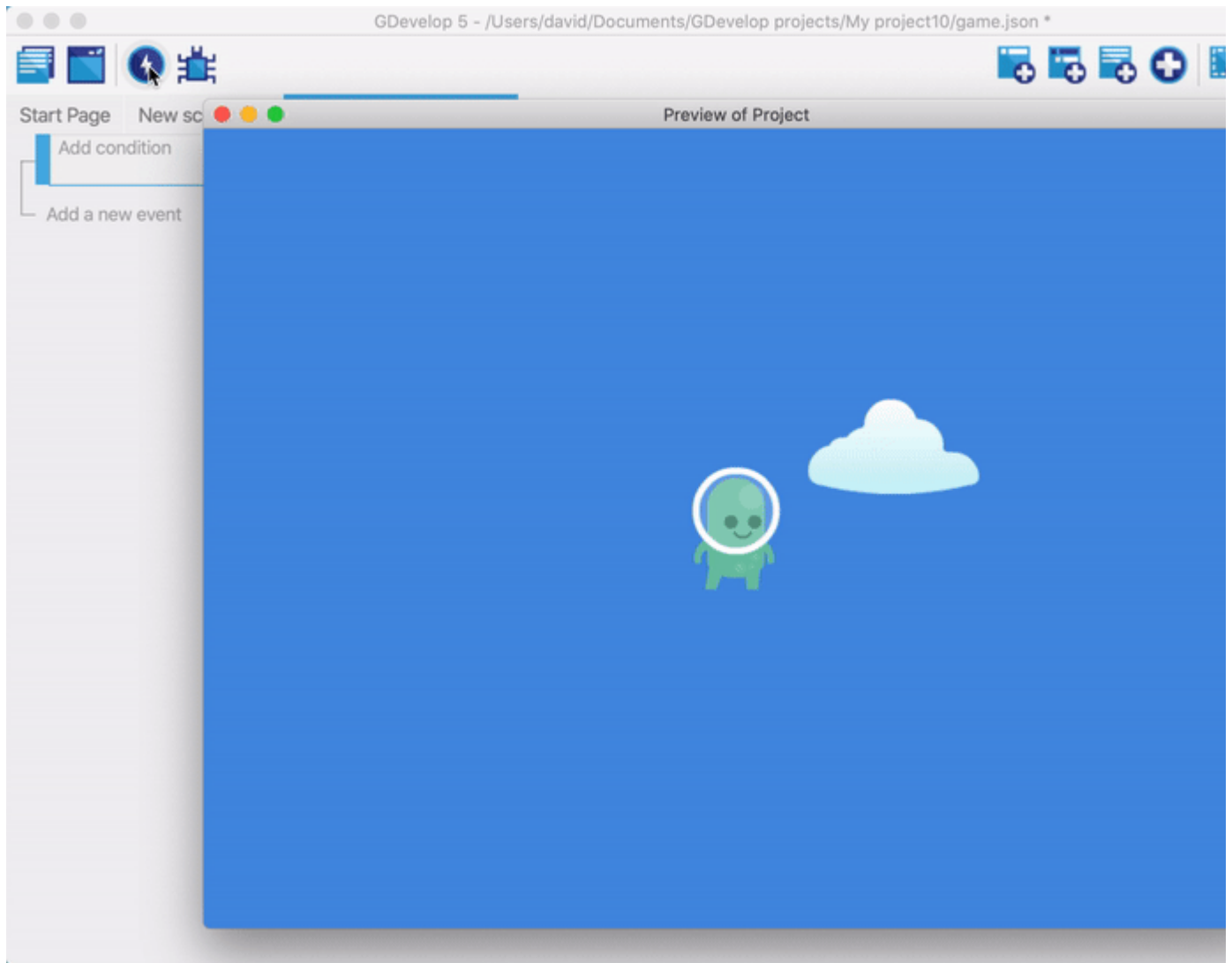
GDevelop has a large library of conditions and actions that add all sorts of features to a game. A big part of learning GDevelop is discovering the available options. In this case, the game needs the **Center the camera on an object** action.

To add the **Center the camera on an object** action:

1. Opens the Events editor.
2. Click **Add a new event**.
3. Without specifying a condition, click **Add action**. By not specifying a condition, the action runs in every frame. Therefore, if the game runs at 60 frames per second, the action runs 60 times per second.
4. Click **Other actions**.
5. Expand **Layers and cameras**.
6. Select **Center the camera on an object**.
7. Select the "Player" object.
8. Click **OK**.



If you preview the game, the player still falls – there's still no platforms, after all – but the camera also follows the player. As a result, the player remains within the frame.



## Next steps

Read [Platformer Tutorial, Part 3](#)