Platformer Tutorial, Part 1: Getting Started

This tutorial explains how to create a platformer game with GDevelop. If you've never created a game with GDevelop before, this is great place to begin.

You'll learn how to:

- Design a level with platforms, enemies, and collectible coins.
- Create a playable character that can move, jump, and kill enemies.
- Return the playable to checkpoints when they die.

Series

You are reading **Part 1** of the [Platformer Tutorial](#).

Step 1: Install GDevelop

If you haven't already, download and install GDevelop. You can find the full instructions on how to set up the software via the [Getting Started](#) page.

Step 2: Download the game's assets

The term *assets* refers to a game's audio and visual media, such as sound effects and sprites. Every game needs assets and, to make your life that little bit easier, this tutorial provides the assets you need to create the game.

You can download the assets via the following URL:

http://www.compilgames.net/dl/PlatformerTutorialResources.zip

> You're also more than welcome to use different assets. Naturally though, the game you see in the screenshots will look different from the game you end up creating.

Step 3: Create a new project

In GDevelop, a *project* is a container for the game's assets and logic. In this tutorial, the terms "project" and "game" are used synonymously.

To create a project:

1. Launch GDevelop.
2. Select **Create a new project**.
3. Choose a folder for the project.
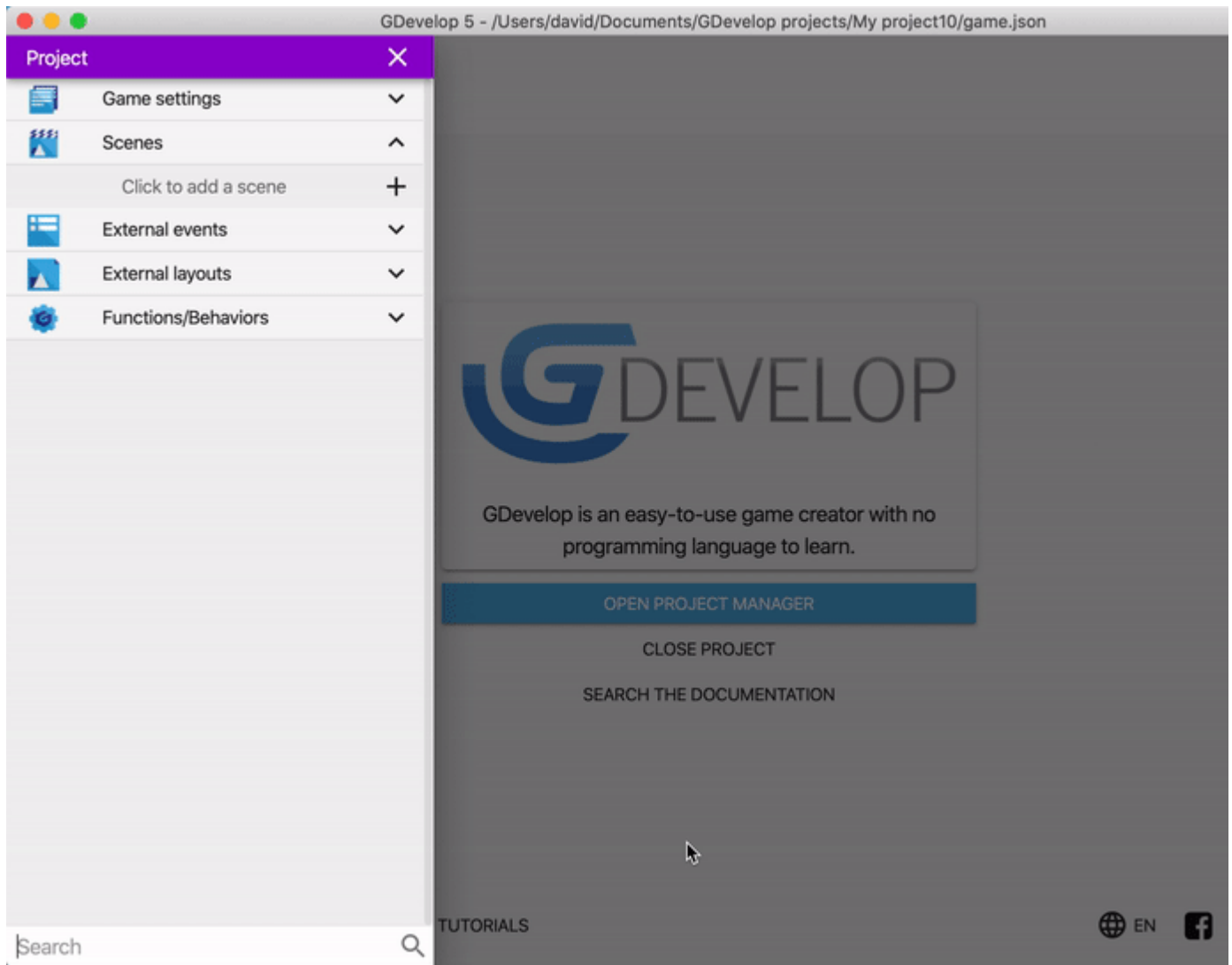4. Select **Empty game**.
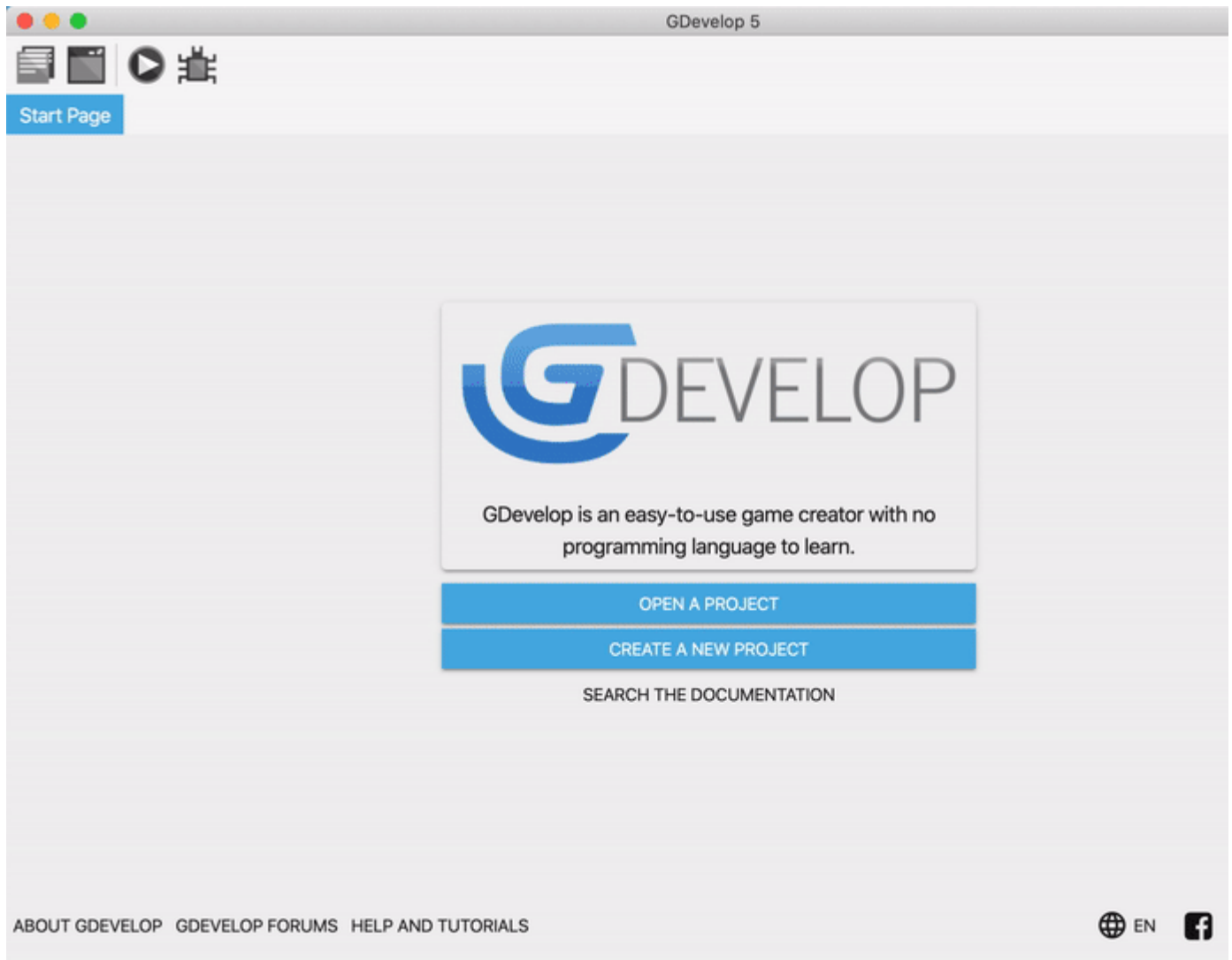
Step 4: Add a scene to the game

A game is made up of *scenes*. Typically, each scene is a different screen in the game. For example, the main menu, pause menu, and each level in a game would all be different scenes.

To add a scene to the game:

1. Expand the **Scenes** panel (if it's not already expanded).
2. Select **Click to add a scene**.

Then select the scene to open it.

If you add multiple scenes to a project, the first scene in the list is the scene that GDevelop loads when the game starts.

Step 5: Add objects to the scene

A scene is made up of objects. Every thing the user sees or interacts with is an object, including the playable character, enemies, and parts of the environment.

GDevelop provides different types of objects that you can add to a game. Each type of object has unique features and use-cases. The most common type of object is the Sprite object, which lets you create a graphic that you can (optionally) choose to animate.

To create an object:

1. In the **Objects** panel, click **Add a new object**.
2. Select **Sprite**.
3. In the **Object name** field, type "Cloud".
4. When the object editor opens, click **Add an animation**. Even though the label says **Add an *animation***, Sprite objects don't have to be animated.
5. Click **Add**.
6. Select the "cloud2.png" asset.

7. Select **Apply**.

Then drag one or more copies of the "Cloud" object into the scene.
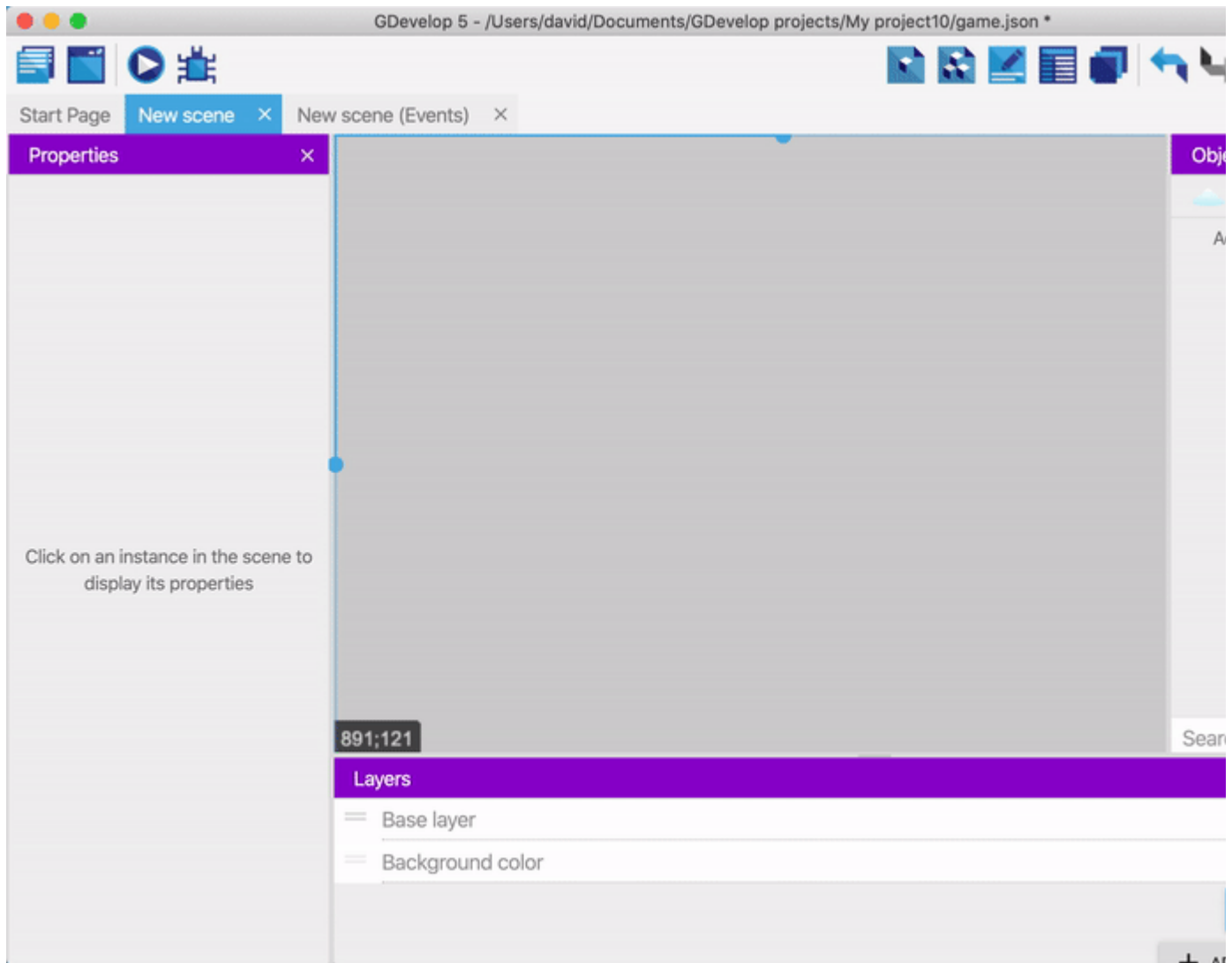
> Each copy of an object that appears in a scene is referred to as an *instance* of that object.
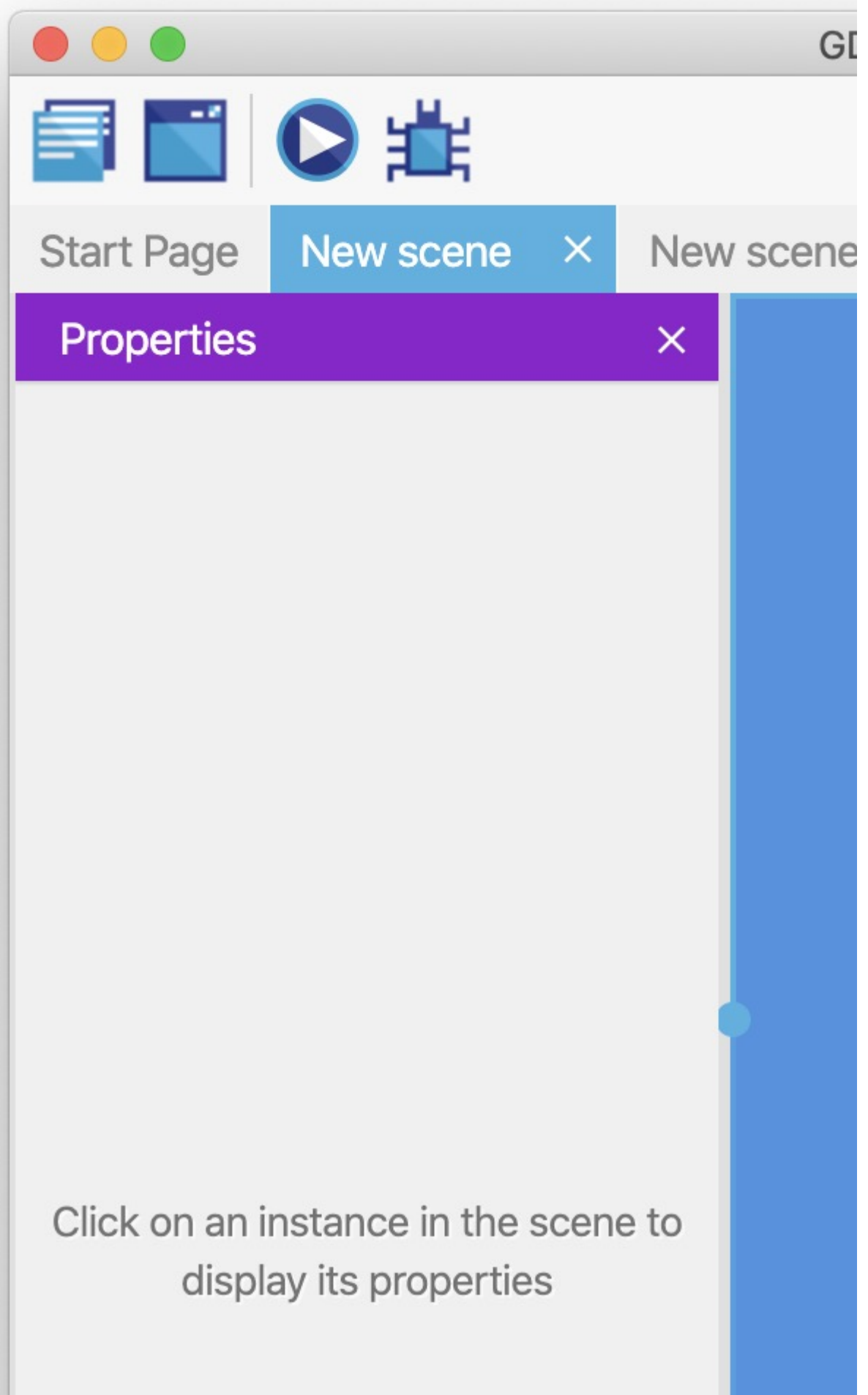
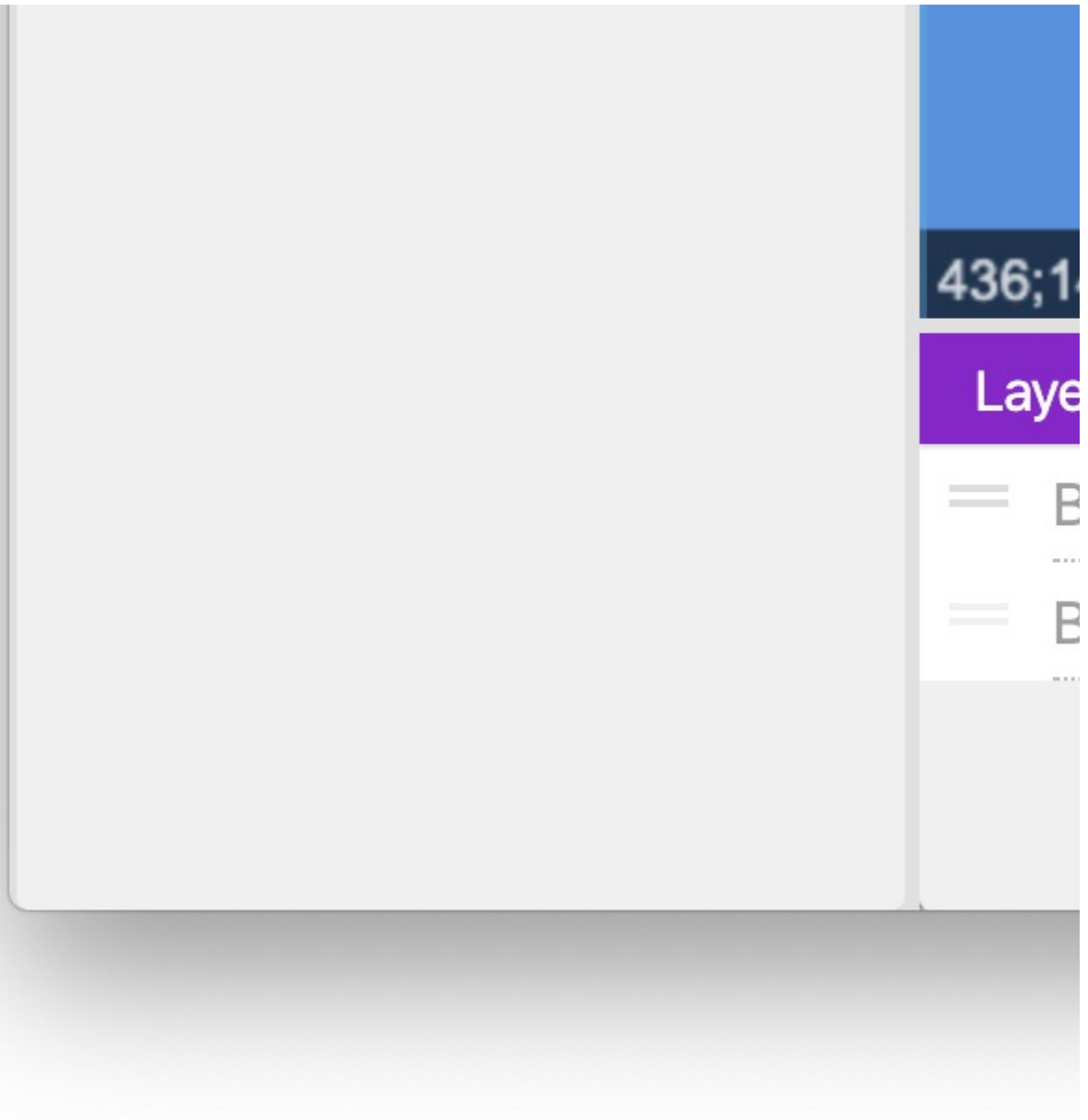# Step 6: Change the background color

The white clouds don't look great on the grey background.

To change the background color of the scene:

1. Right click the scene's background.
2. Select **Scene properties**.
3. Click **Scene background color**.
4. Choose a nice shade of blue.

> You can enter a hex (Ex: #00000) or a name of the color (Ex: Black)

Start Page | **New scene** ✕ | New scene

Properties ✕

Click on an instance in the scene to
display its properties
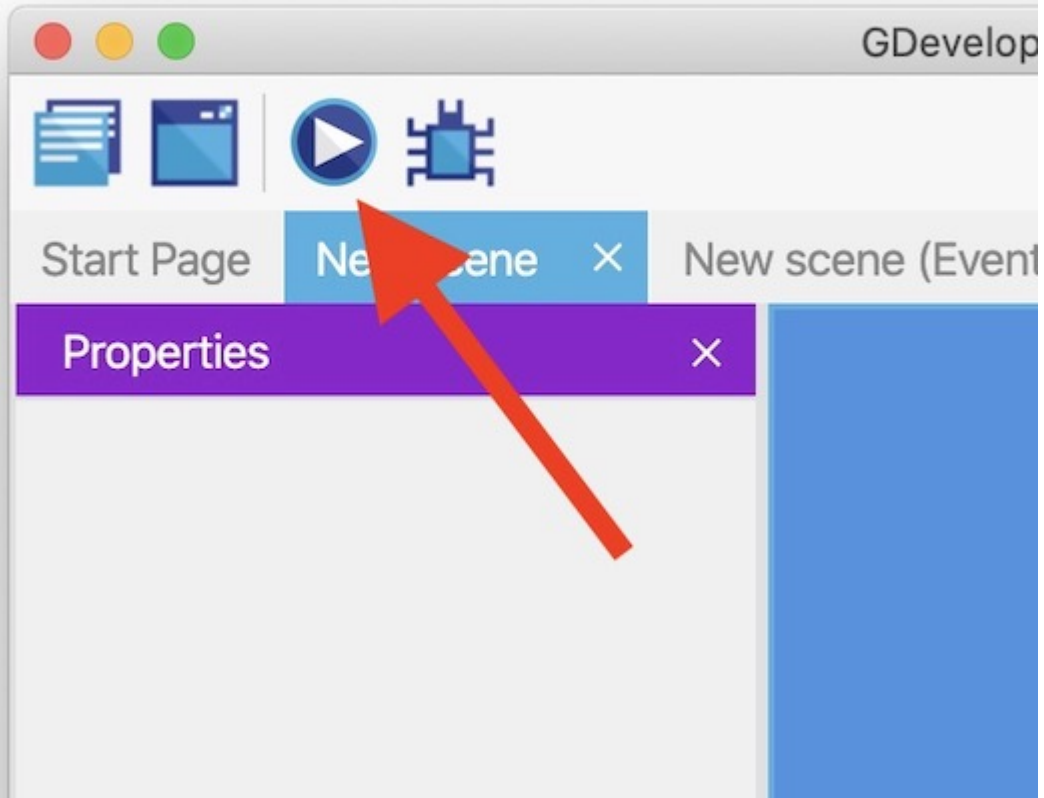
436;1

Laye

B

B

Step 7: Preview the game

It's good to get into the habit of regularly previewing your games. Then, when something breaks, it's easier to figure out the problem.

To launch a preview of the game, click the **Preview** button in the toolbar.

To stop the preview, close the preview window.

Step 8: Save the game

It's also good to get into the habit of regularly saving your games. You can save the game via the **File** menu or by using a keyboard shortcut:

- CTRL + S on Windows
- CMD + S on macOS

Next step

Read Platformer Tutorial, Part 2 and learn how to create a playable character.