This lesson follows the Construct 2 platformer tutorial found [here](https://www.scirra.com/tutorials/253/how-to-make-a-platform-game). This whole tutorial covers creating a platformer from beginning to end with no knowledge of Construct 2.

**Objective:**

At the end of this lesson, the students will be able to import images into Construct 2 that can be used for backgrounds, players, enemies, and world geometry. They will be able to control a playable character.

**Layout**

1. **Creating a New Project**
   1. Open Construct 2.
   2. Click the “File” button.
   3. Choose the “New empty project” option from the “Select Template or Example” window.
      1. This window show plenty of examples that the students can explore on their own time.
   4. There should now be an empty Layout and Event Sheet.
      1. You can add new Layouts and Event Sheets at any time. To do this, right click on the “Layouts” or “Event sheets” folder, then click “Add layout” or “Add event sheet”.
2. **Size the Layout**
   1. A Layout in Construct 2 is equivalent to a Level in a video game. Each level a student wants to create should be created in a new Layout.
   2. Changing the size of the Layout changes the size of the level as a whole. The “Layout Size”property is found in the “Layout Properties” section of the “Properties” window.

**Background**

1. **Adding a Background**
   1. Double-click in the empty area of the “Layout”. This will bring up the “Insert New Object” window.
   2. Choose the “Tiled Background” option.
      1. If, the student wants to use a single non-tiled they would then choose the “Sprite” option. All following instruction will be the same.
   3. Name the Background object.
   4. Click the “Insert” button.
   5. You will be returned to the Layout and the mouse cursor will change to a target. This allows you to place your background wherever you would like in the Layout by clicking one more time. (This won’t matter, because we will be changing the position and size shortly.)
   6. After choosing a position to add your background, you will be sent to the “Image Editor” window.
   7. Here you can upload or draw your own background.
   8. Click the “Open” button to select a background file to be used
      1. Have the students use the provided background for this class. They will be allowed to choose their own backgrounds later.
   9. Close the Image editor and the background will be added to the Layout. Now it needs to be resized to fit the Layout size.
      1. Make sure the background is selected.
      2. Set the “Position” property in the Common section of the background properties to 0, 0.
      3. Set the “Size” property in the Common section of the background properties to 4000, 2048. This should match the Layout size.
      4. You can use the mouse wheel to zoom in and out to see the entire Layout.
2. **Adding a Layer**
   1. With a background that covers the entire Layout, it will be difficult to add new sprites to our game. The background is blocking the Layout field so any mouse clicks will tend to select the background. Layers solve this problem and add some organization to the project.
   2. To add a Layer, select the “Layers” tab.
   3. Click the “+” at the top of this field to add a new Layer.
   4. You can name Layers to keep track of what is in each Layer.
      1. You can either double click the Layer name or right click and select the “Rename” option to name a Layer.
   5. Name the original Layer “Background”. Name the new Layer “Main”.
   6. You can lock the background layer by clicking the lock next to it’s name. This prevents any editing of a layer that is locked.
   7. The checkboxes allow to to hide and show Layers.
   8. Make sure that the “Main” Layer is selected before continuing.

**Tiles**

1. **Adding Tiles**
   1. Tiles are what the students will use as platforms for this platformer.
   2. Similar to adding a background double-click in the Layout to add a new object and name it “Tile”.
   3. Chooses “Sprite” and click on the screen again to select where the sprite gets added.
   4. The image editor will pop up again but with extra panes: “Animations” and “Animation Frames”. These panes allow for sprite animating.
   5. These Tiles will have different animation frames but they will not animate. To stop the animation, select the “Default” animation from the “Animations” pane. Set the “Speed” property in the “Properties” pane to 0.
   6. With 0 speed, the animation will not progress to the next frame.
   7. Right click any space in the “Animation Frames” pane.
   8. Highlight “Import frames” then click “From sprite strip...”.
      1. This allows us to load a grid of images and cut them up into individual animations.
   9. A file open dialog appears. Have the students use the tiles provided for now. They will be allowed to use their own images later
   10. After loading in the tile sprites, Construct 2 need to know how many tiles there are. The “Import sprite strip” pane will appear.
       1. Set the “Number of horizontal cells” to 3.
       2. Set the “Number of vertical cells” to 1.
   11. Once the new tiles have been loaded, delete the first empty tile in the “Animation Frames” pane.
   12. Close the Image Editor pane.
   13. You now see a single tile on the screen. To change the tile that is shown, change the “Initial frame” property in the Properties pane. This is a 0 based system so the first frame is 0 the second is 1 and so on.
2. **Creating Scenery from tiles**
   1. To copy the tile on screen, hold down control and then click and drag the tile. A duplicate will be created. Repeat as needed
   2. Change the “Initial frame” property to change which tile is shown.
3. **Enabling Snap to Grid**
   1. It will take forever to line up these tiles at this point. You can enable a grid that will make the job easier.
   2. Click the “View” tab from the top ribbon.
   3. Then click the checkbox next to “Snap to Grid”.
   4. You can change the size of the grid to whatever you need. We will keep the 32 x 32 fro now.

**Creating a Playable Character**

1. **Adding the Player Sprite**
   1. Double-click an empty space in the Layout.
   2. Choose Sprite and change the name to “Player”.
   3. Click on the Layout where you want the player to be added.
   4. Select the “Default” animation and rename it to “Idle”
   5. Right-click in the Animation frames pane.
   6. Highlight “Import frames” and select “From files...”
   7. Select the Player’s Idle frame.
   8. Set the Speed to 0 and Loop to No.
   9. Cropping Extra Whitespace
   10. You will see that the sprites all have extra space around them. This is bad for things like collision. If we left the extra space, then objects will interact with the player without actually touching the player. Cropping this space fixes the issue.
   11. The quickest way to do this is to hold shift, then click the “Crop” button from the top ribbon of the Image Editor pane. This removes all whitespace from the image.
   12. Repeat for all frames
   13. Close the Image Editor.
   14. Select the Player Sprite and change the “Initial animation” from the Properties pane to “Idle”
2. **Setting the Origin**
   1. The Origin of the image is where the position and the rotation originate from. If the origin is in the middle, the sprite will rotate around the center and position to that center. For platformers we do not want that. The origin in our game should be placed at the feet of the sprite so it will land on platforms correctly.
   2. If the Image Editor pane is not open, double-click on the sprite to open it.
   3. Select the Default animation from the Animations pane.
   4. Click the “Set origin and image points” button. The Image Points pane will appear.
   5. Make sure the “origin” point is selected. There are multiple ways to adjust this point
      1. With “Num Lock” on, click 2 on the Numpad. This works for 4(left), 5(center), 6(right), 8(top) and all number in between.
      2. Click and drag the origin point in the Image Editor pane.
      3. Right-click on the origin point in the Image points pane. Highlight Quick assign. Select the option you would like.
   6. You can do this for each frame, but that would be tiresome. In the Image points pane, right-click the origin and click “Apply to whole animation”.

**Behaviors**

Construct 2 comes with a lot of built in functionality for making games. We will utilize some of these to make our games.

1. **Adding Player Box**
   1. Double-click on the Layout and create a new Sprite
   2. Name it PlayerBox. Click on the Layout again to place it.
   3. In the Image Editor pane, select the fill tool.
   4. Change the color to a semi transparent color by setting the Alpha value to 125.
   5. Use the fill tool to fill the entire sprite with that color.
   6. Set the Origin point to match the Player sprite.
   7. Close the Image Editor pane.
   8. Move and resize the new PlayerBox sprite to match the size and position of the Player sprite.
   9. Set the “Initial Visibility” in the Properties pane to “Invisible”. This will prevent us from seeing the square once the game is being played.
2. **Adding Platform Behavior**
   1. his Behavior provide controls for moving left, right, and jumping. It works best on an object that doesn’t move.
   2. Click “Behaviors” by “Edit Behaviors” of the PlayerBox Properties pane.
   3. Click the “+” button from the pane that appears. This will bring up the Add Behaviors pane.
   4. Find the “Platform” behavior under the Movement section. Select it and then select “Add”.
   5. We also want the camera to follow the player so click the “+” button and add the “Scroll To” Behavior.
   6. Close the Behaviors pane.
   7. You will see new properties for platform movement appear in the PlayerBox Properties pane. Set the “Jump strength” to 1100 and the “Gravity” to 2500.
3. **Add the Solid Behavior**
   1. The Player needs to know what objects it can land on. Any object with the “Platform” Behavior will fall through any object that does not have the “Solid” Behavior attached to it.
   2. Click on the Tile sprite.
   3. Click “Behaviors” by “Edit Behaviors” of the Tile Properties pane.
   4. Click the “+” button from the pane that appears. This will bring up the Add Behaviors pane.
   5. Find the “Solid” Behavior in the “Attributes” section. Select it and click the “Add” button.

**Event Sheet**

Events are the code you write to control every aspect of the game. Events work by testing if a series of conditions have been met. If they have, the actions are run. Events are created inside of Event Sheets. Select Event Sheet 1 now.

1. **Updating Player Position**
   1. Currently, the PlayerBox is what is moving around the Layout because is has the Platform Behavior on it. We need to update the Player sprite every frame to follow the PlayerBox movement.
   2. Double-click any empty spot on the Event Sheet or click the “Add event” button. This will bring up the Add Event pane.
   3. Double-click the System Event.
   4. Find the Every tick Event under the General Section. Double-Click on this event to add it to the Event Sheet.
   5. Click the “Add action” button next to the newly created event. The Add Action pane will appear.
   6. Double-Click the Player object.
   7. Double-Click the “Set position to another object” action.
   8. For Object, click <click to choose> and pick PlayerBox.
   9. Leave Image point as 0.
   10. Click Done.

Now you should have a controllable character That can move and jump. You should also have multiple platforms that you can navigate with your player.