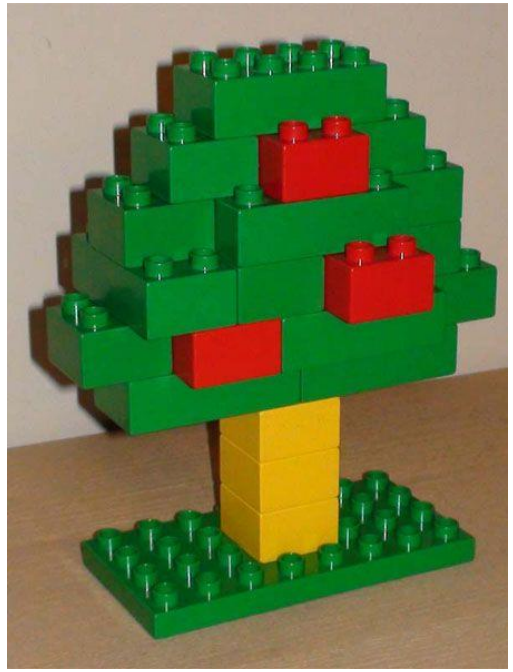
Tilemaps

What's a Tilemap in Unity?

- A Tilemap is a *component* in Unity (like Rigidbody2D and BoxCollider2D) that helps us design levels using tiles of sprite art

- The reason it's called a Tilemap is because the art used is all the same square dimensions (e.g. 32 pixels), and we place the tiles together one by one until we create a complete image or level

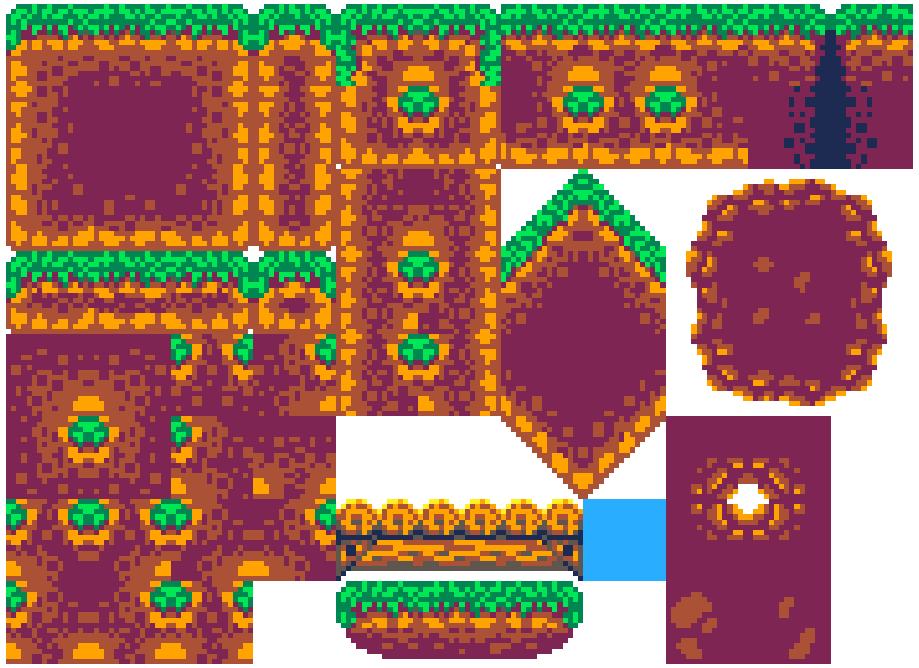
- Remember Legos? It's the same concept of using blocks to create a whole image or object



=

Exercise #20 – Design Using Tilemaps

- Import a spritesheet containing the art you'd like to use. Something like this

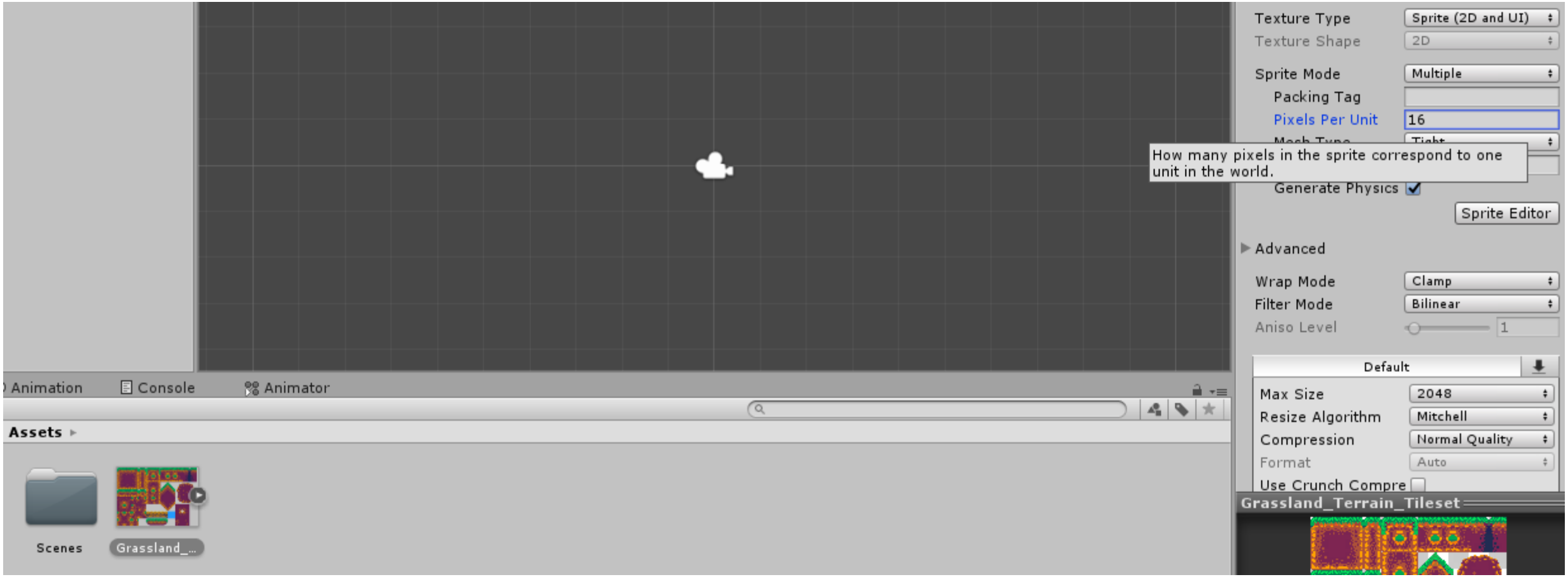


- You can find more tilesets like this on websites like OpenGameArt.org

Exercise #20 – Design Using Tilemaps (Cont.)

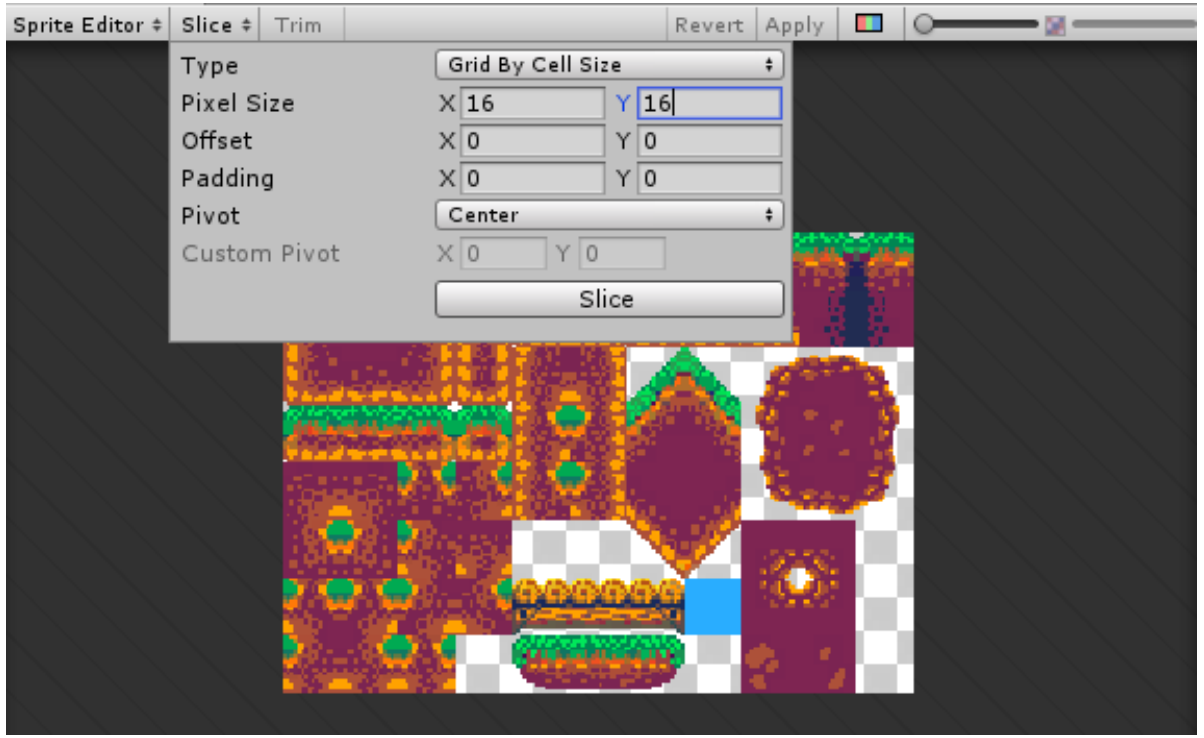
- Use a tool such as an online Pixel ruler or Adobe Photoshop to figure out the width of the smallest tile in your sprite sheet
- For this example specifically, the width is 16x16 pixels
- Select the spritesheet in the Project window
- Adjust the Pixels per Unit value as seen in the next page

Exercise #20 – Design Using Tilemaps (Cont.)



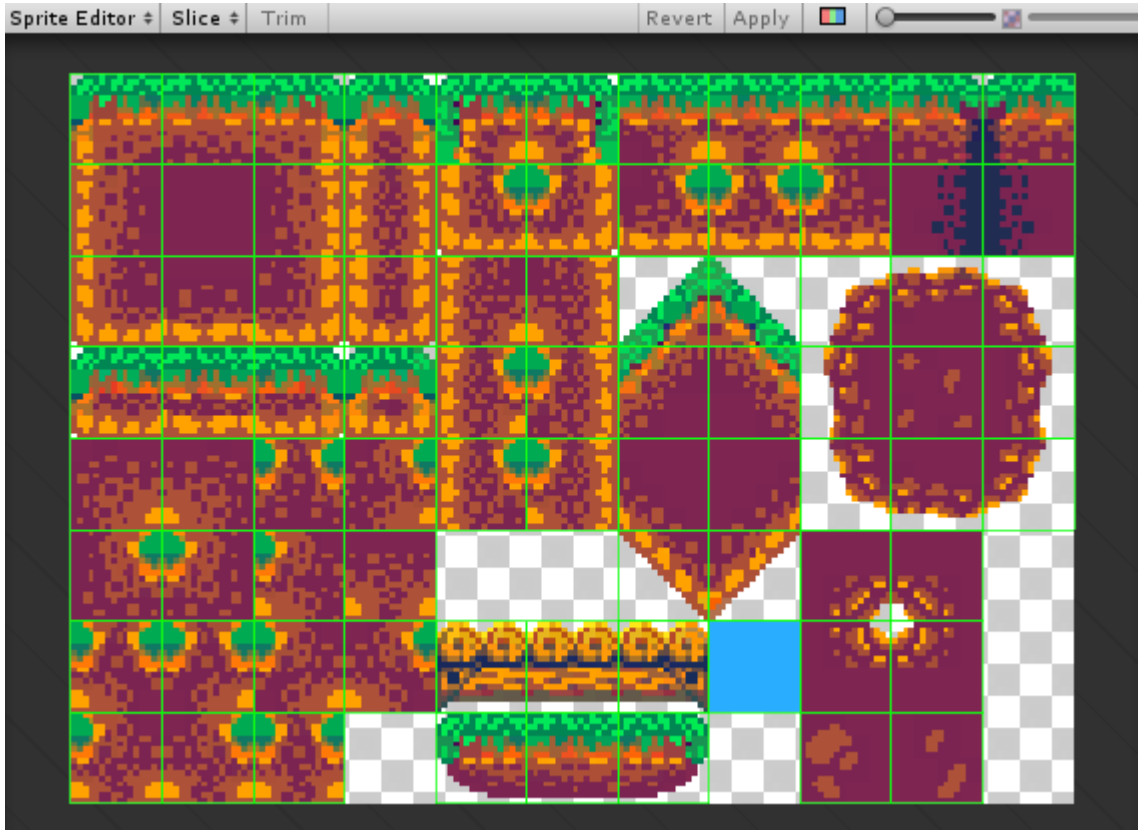
Exercise #20 – Design Using Tilemaps (Cont.)

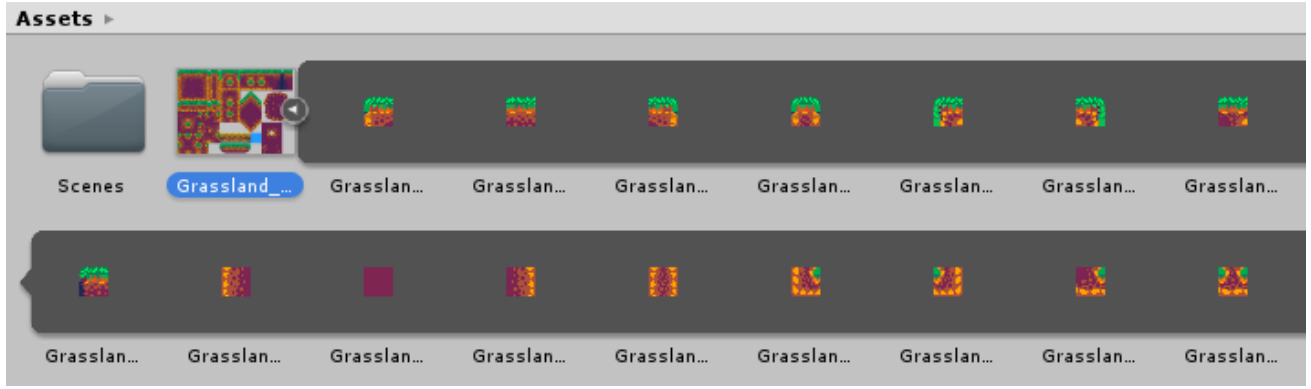
- Set the Sprite mode to Multiple and use the Sprite Editor to cut up the sprite sheet as shown in Lab 1. However there will be a few adjustments:
  - Change Type from Automatic to Grid by Cell Size
  - Change the pixel size of X and Y to 16, then click Slice, then Apply



Exercise #20 – Design Using Tilemaps (Cont.)

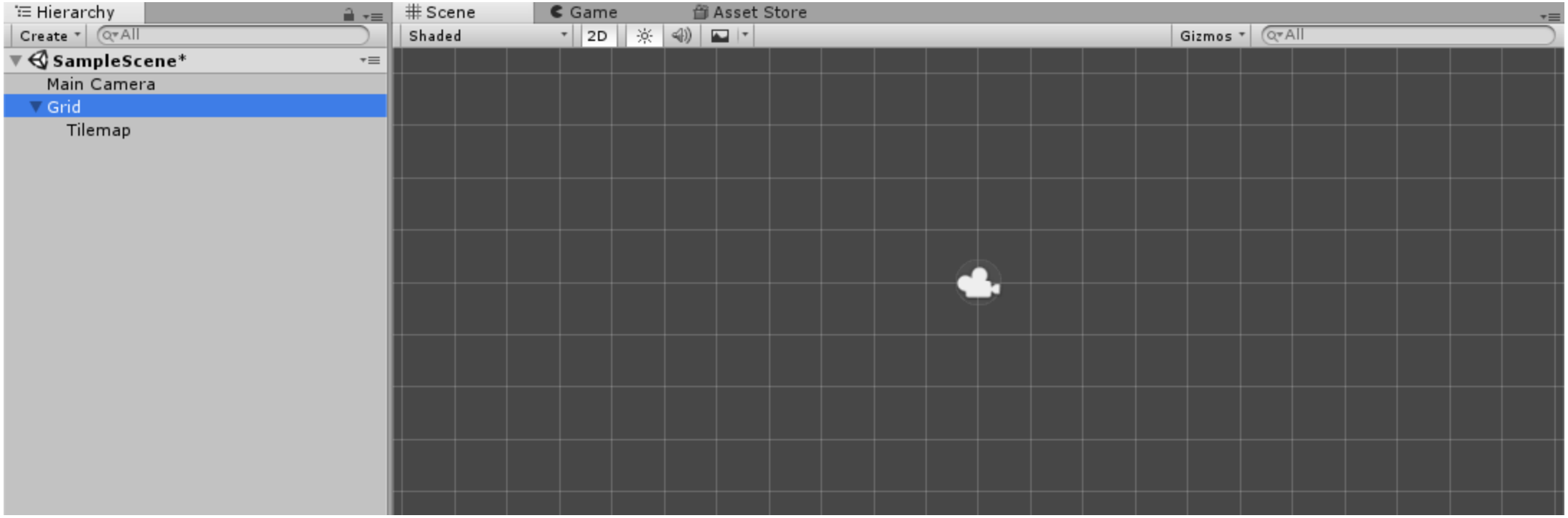
- Note the gridlines all over the sheet after slicing:





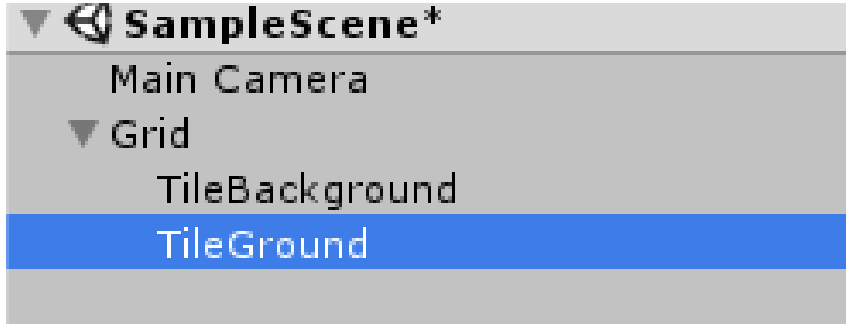
Exercise #20 – Design Using Tilemaps (Cont.)

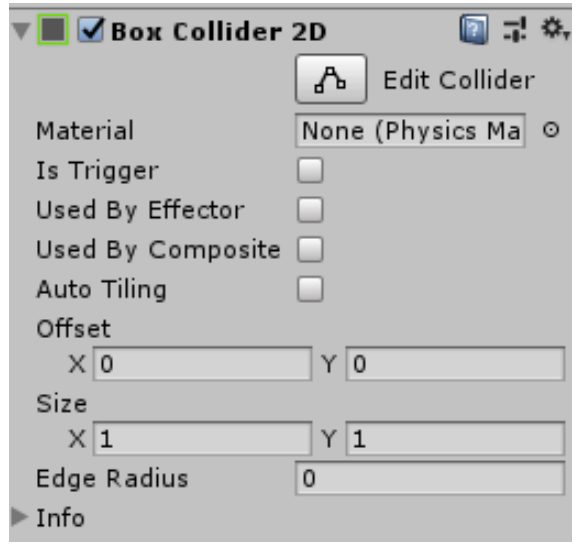
- Now that the tiles are ready, we will create a Tilemap
- Tilemap is by default a child of a gameObject called *Grid*
- To create a tilemap, right-click in the Hierarchy>2DObject>Tilemap



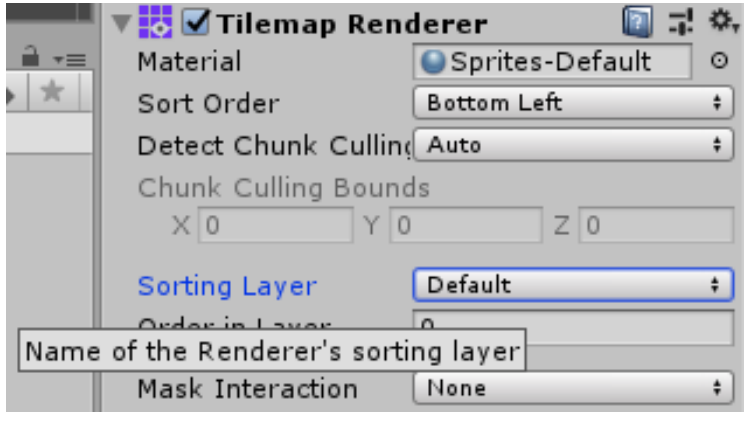
Exercise #20 – Design Using Tilemaps (Cont.)

- A Grid can have many tilemap children in order for you to create layers and add tiles on top of each other as your design requires
- Create 2 Tilemaps underneath the grid, one called TileBackground and the other TileGround
- Add BoxCollider2D to TileGround only to make it solid. You may need several, to cover all floors and walls
- Make the Layermask Ground so player can jump without issue



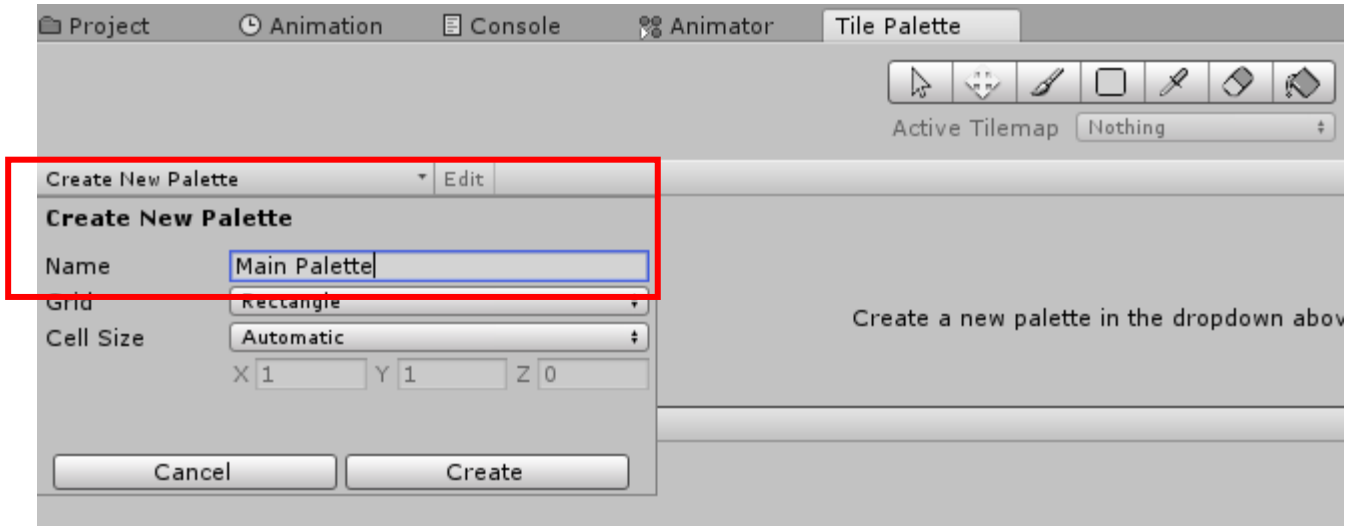


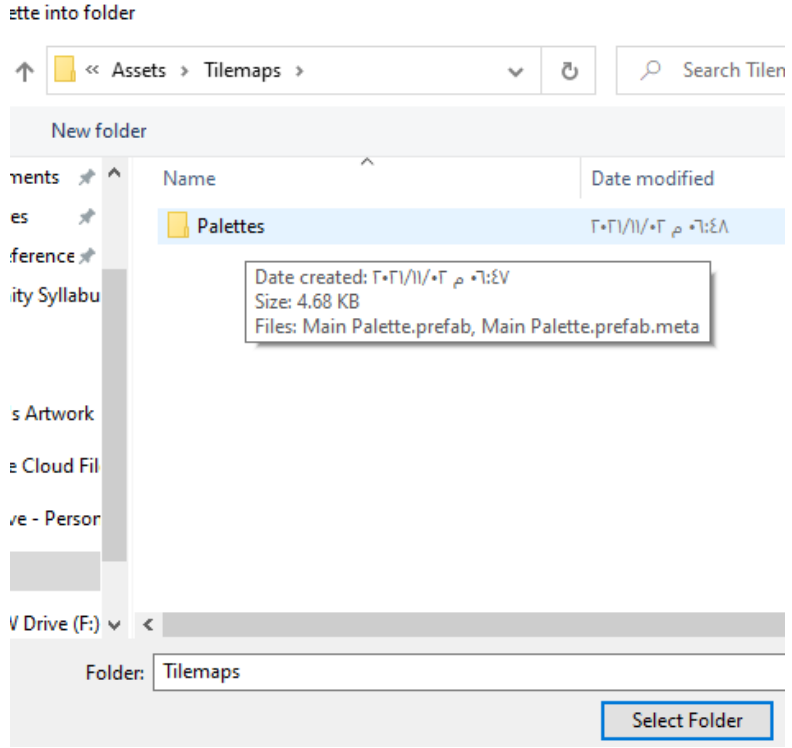
- In the Tilemap Renderer component in the Inspector window, you can create sorting layers and order layers as done with ordinary sprites



Exercise #20 – Design Using Tilemaps (Cont.)

- Go to Window>2D>Tile Palette
- Click Create New Palette, name it, and create a folder called Tilemaps. Within Tilemaps, create another folder called Palettes and save your palette in there

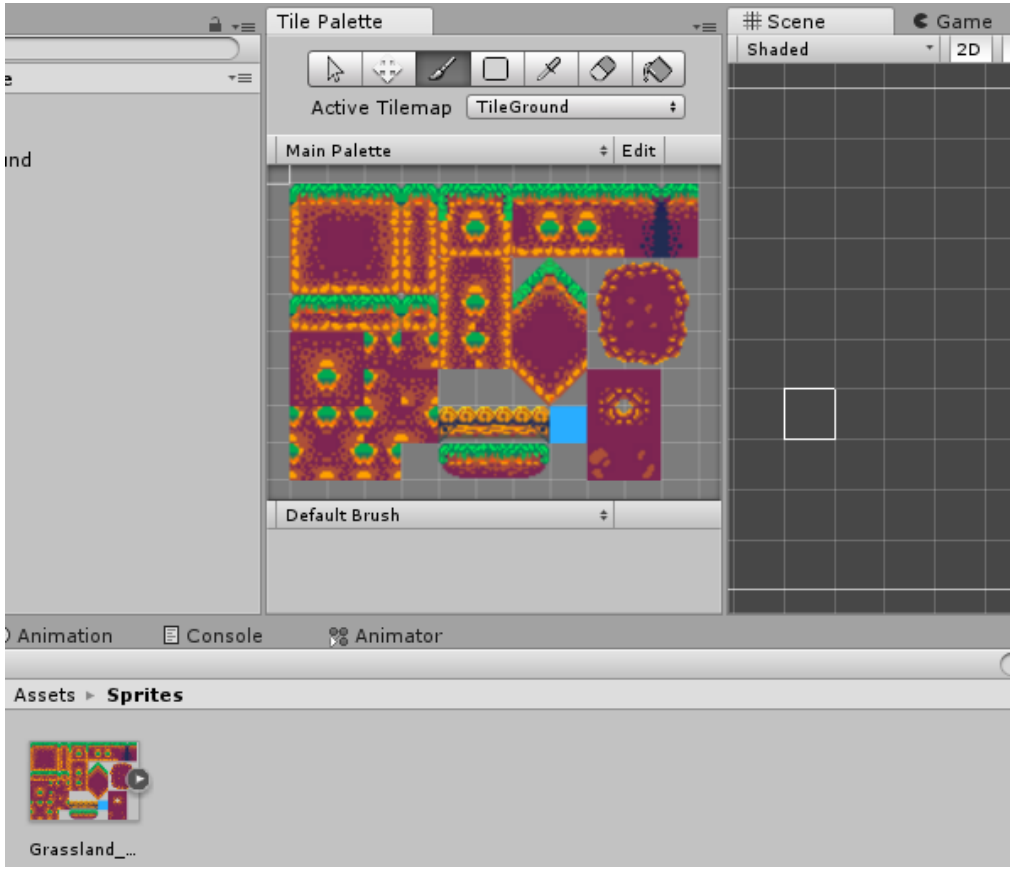




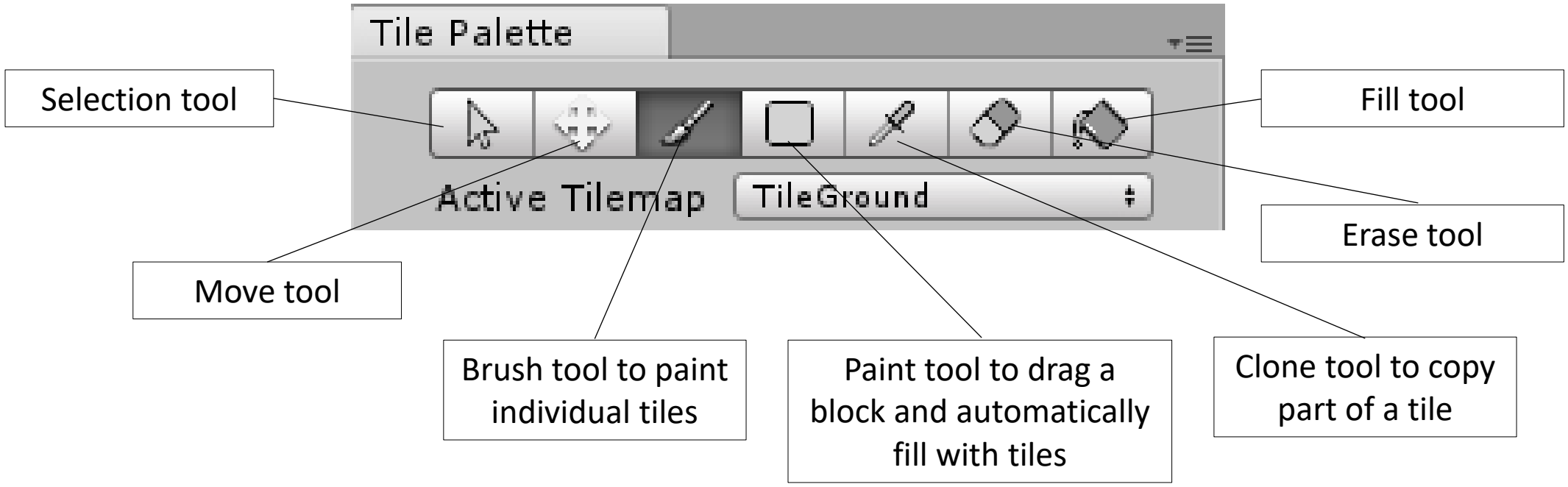
Exercise #20 – Design Using Tilemaps (Cont.)

- Drag the sliced up image and drop it into the palette window
- Unity will ask you to create a new folder. Name it Tile Assets and place it in the Tilemaps folder

- Refer to this screenshot of the tiles successfully copied into the Tile Palette

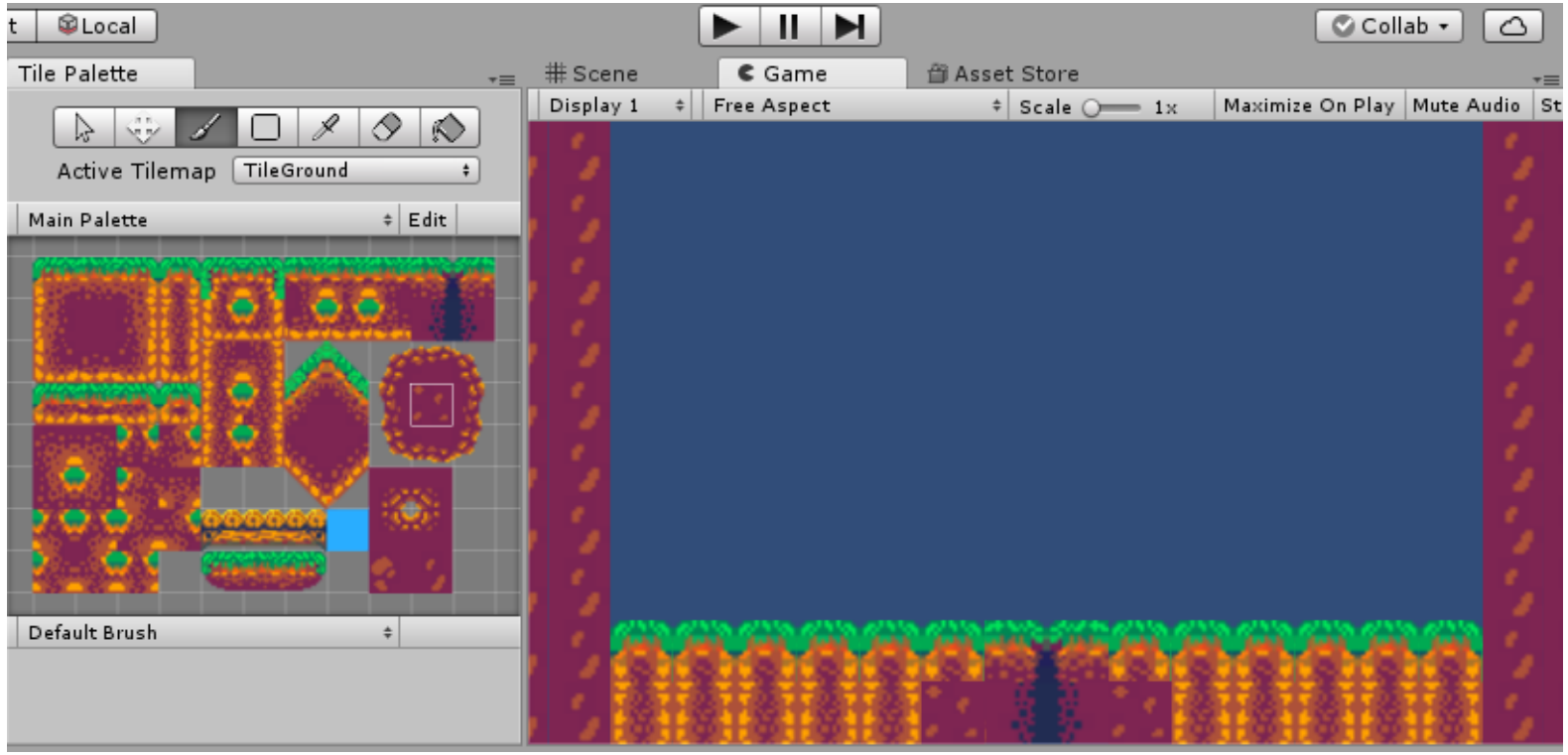


Exercise #20 – Design Using Tilemaps (Cont.)



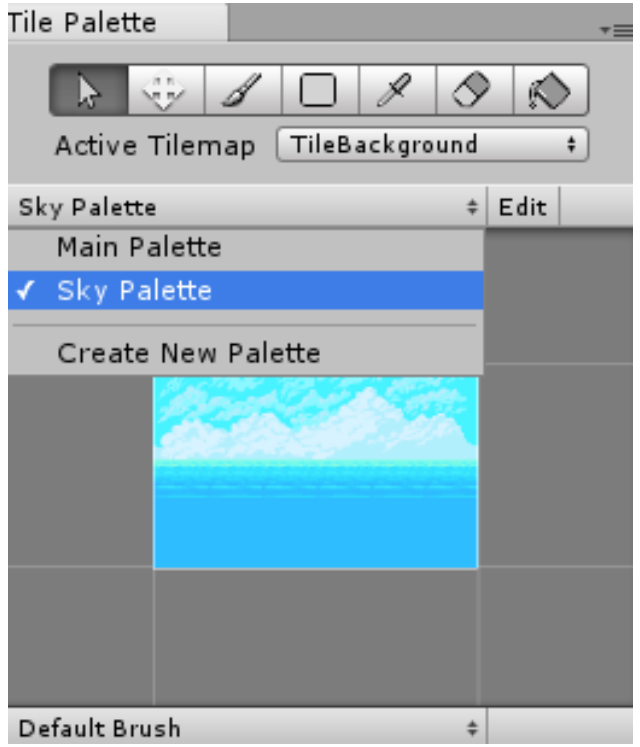
Exercise #20 – Design Using Tilemaps (Cont.)

- A quick example of tiles placed on TileGround
- These tiles represent walls and floors, so the added BoxCollider2D makes them solid for the player to stand on



Exercise #20 – Design Using Tilemaps (Cont.)

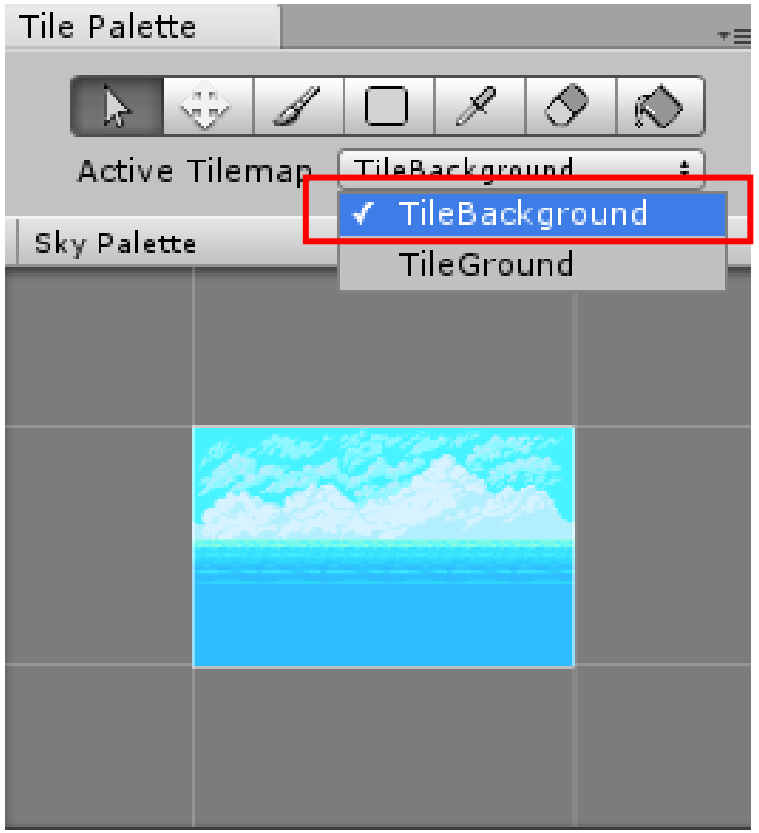
- Now let's create the sky behind these walls and ground
- This time, for variation, we're using a whole sky image not indiv. tiles
- Create a new palette and call it 'Sky Palette' for example. Save it in the Palettes folder, the same location as the previous main palette



- Now drag the sky image into the palette window, and the window pop-up will appear. Save this tile in the Tile Assets folder you created earlier

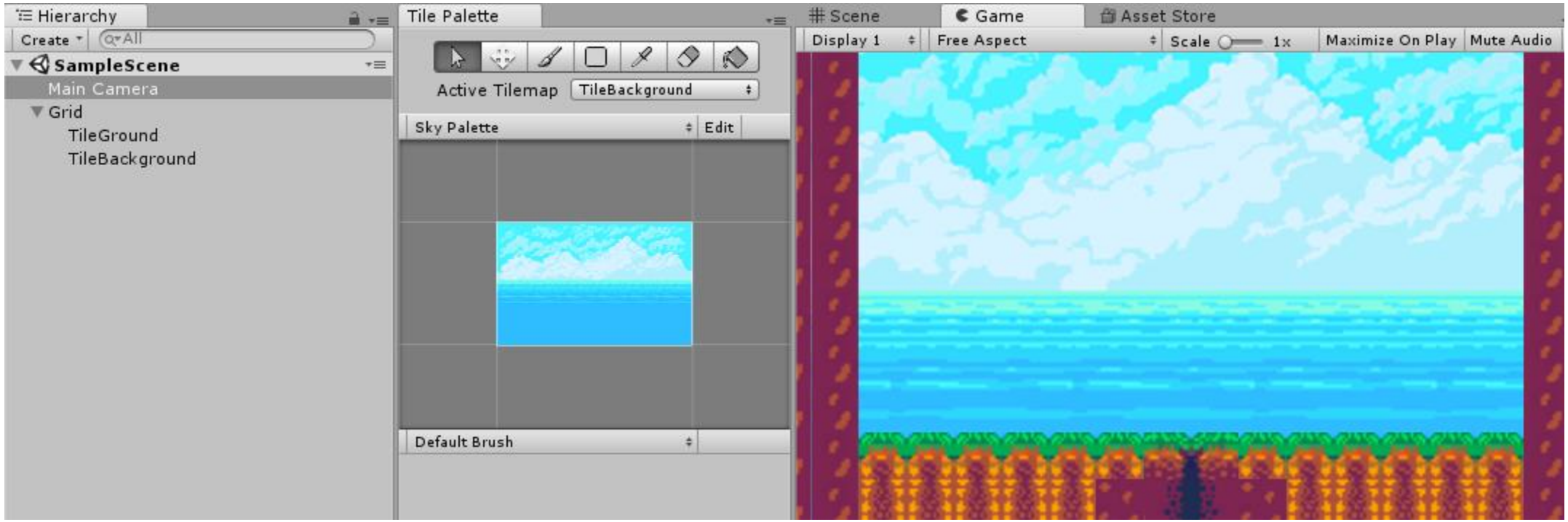
Exercise #20 – Design Using Tilemaps (Cont.)

- Now change the ‘Active Tilemap’ option to be TileBackground instead of TileGround, since this is the tilemap we want to design



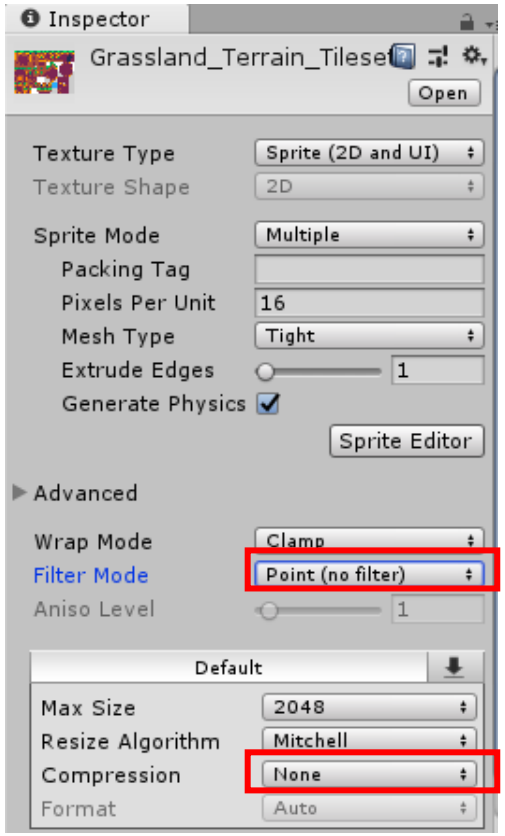
Exercise #20 – Design Using Tilemaps (Cont.)

- Now use the brush tool to adjust the sky however you want in your scene to get something like this:



Exercise #20 – Design Using Tilemaps (Cont.)

- **What if there are gaps between the tiles?**
- To remove them, select the original sliced tilesheet from the Assets folder to display properties in Inspector



- Change these settings:
- Compression -> NONE
- Filter Mode -> POINT
- (Optional) Pixel Per Unit -> 15.9 instead of 16, 31.9 instead of 32, etc.

Exercise #20 – Design Using Tilemaps (Cont.)

- Now go to Edit > Project Settings>Quality
- Select anti-aliasing -> DISABLED

- (Anti-aliasing is basically techniques to get *rid* of jagged edges on your display, especially if the pixel is not entirely rectangular in shape. That jaggedness is, therefore, known as aliasing.)

Resources

- **Unity 2D Platformer Tutorial 19 - Tilemaps + Tile Collider.** https://www.youtube.com/watch?v=iGQqVgNYxGM&t=91s
- **Tilemaps in Unity by Brackeys. URL retrieved from:** https://www.youtube.com/watch?v=ryISV_nH8qw&t=424s
- **How to make Tilemaps in Photoshop. URL retrieved from:** https://www.youtube.com/watch?v=Ie4TZc5W5Os
- **Free Tilemap Creator. URL retrieved from:** https://thorbjorn.itch.io/tiled