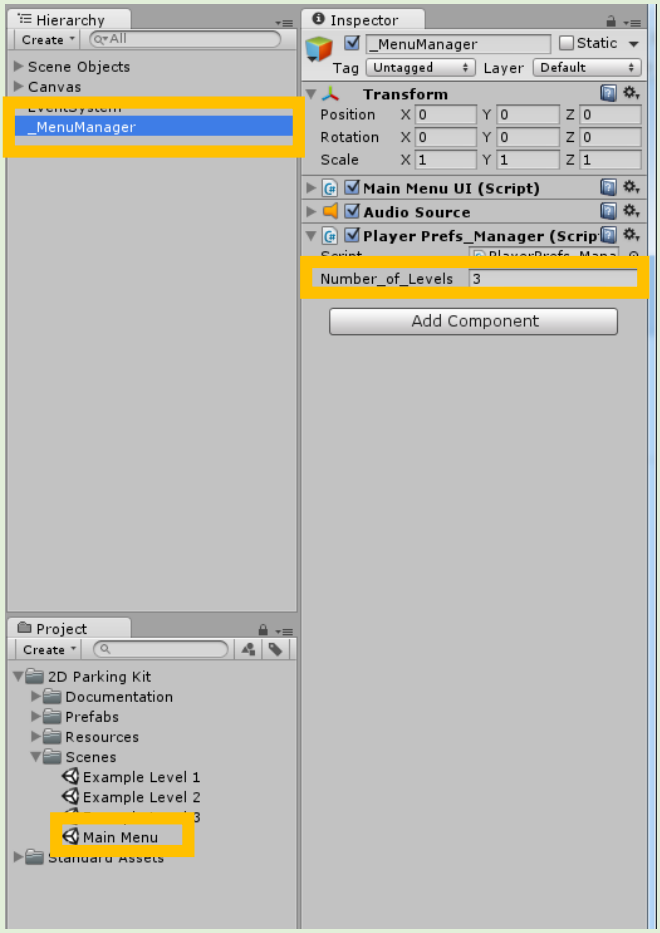
Creating your own levels with the 2D Parking Kit

This guide was created for beginners in the unity engine, a small amount of knowledge is required such as navigation and a familiarity with the interface, however most steps are explained with exact instructions. Although this guide looks long most of the developers with a decent amount of experience in the unity engine should be able to skim through and look at key points as some of the points are just explaining engine operations such as hotkeys and shortcuts.

Table of Contents

Step 1: Adjusting the main menu screen for the new level(s).



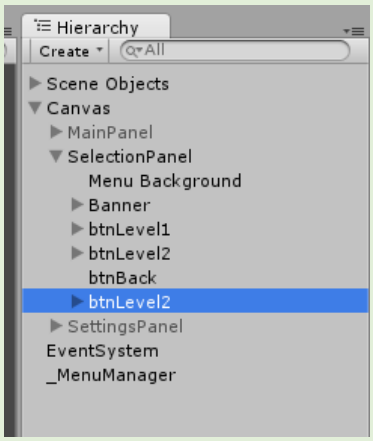
First select the 'Main Menu' scene from the Scenes folder.

Next select the _'MenuManager' GameObject from the hierarchy menu.

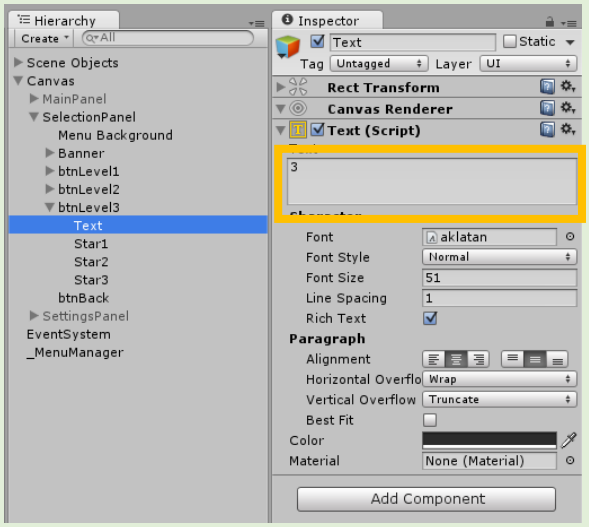
Next we change the number of levels variable on the 'PlayerPrefs_Manager' script to how many levels you are creating, I already have 2 levels and I am adding 1 more, so have entered 3.

Next we want to disable the 'MainPanel' and enable the 'SelectionPanel' we do this by selecting the MainPanel object (child of the Canvas object) and clicking on the tick box to the left of its name. Then selecting the 'SelectionPanel' object and checking the same box.

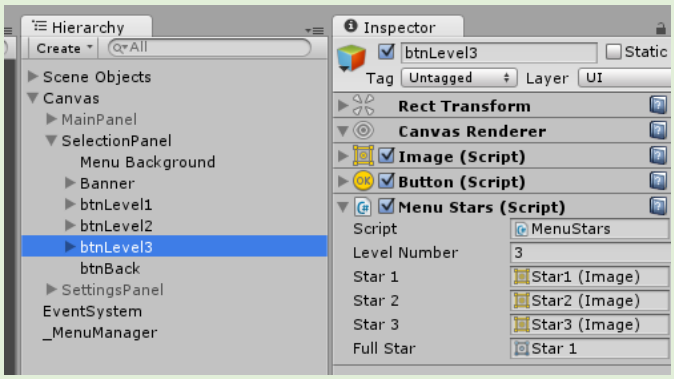
You should now see the selection menu, here I will add another button which the user can select to go to the new level we are adding.



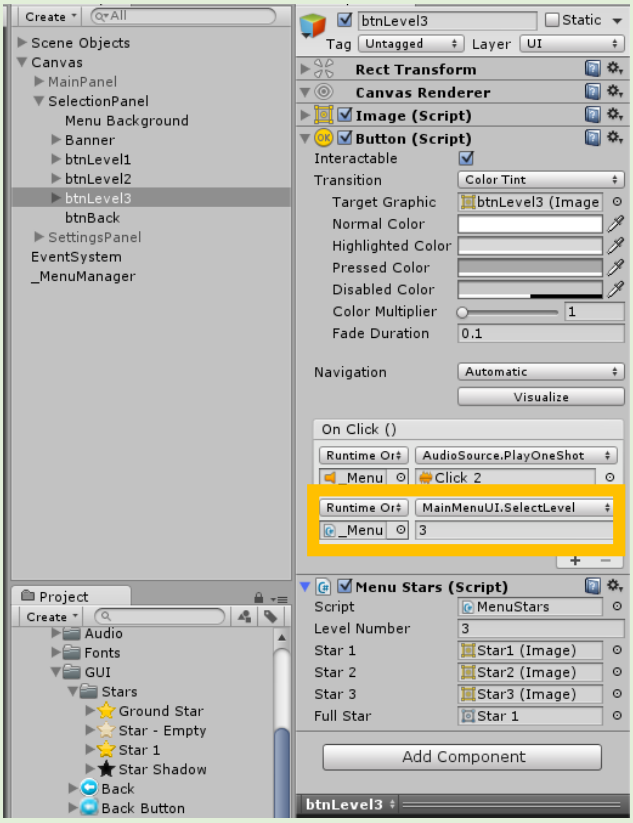
Click on one of the current buttons in the Heirarchy tab and press Ctrl + D on your keyboard (or CMD + D on mac) to duplicate this object. Move the new button so that it fits on in the menu panel.



Then change the name of the new button to its level number. In my case 3.
We now click on the drop down arrow to reveal the objects children, click on the “Text” object and change the Text value to the level number as well (again in my case 3).



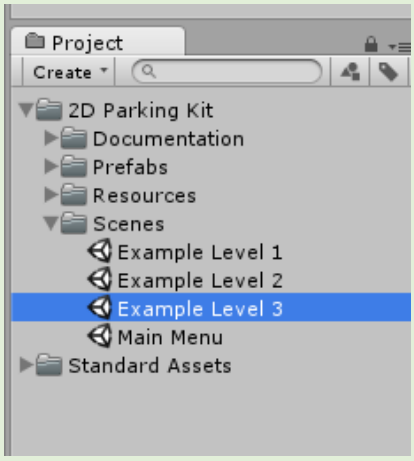
Next we click on the button parent object again and change the ‘Level Number’ value of the ‘Menu Stars’ script to the buttons level number. Again in my case this number is 3.



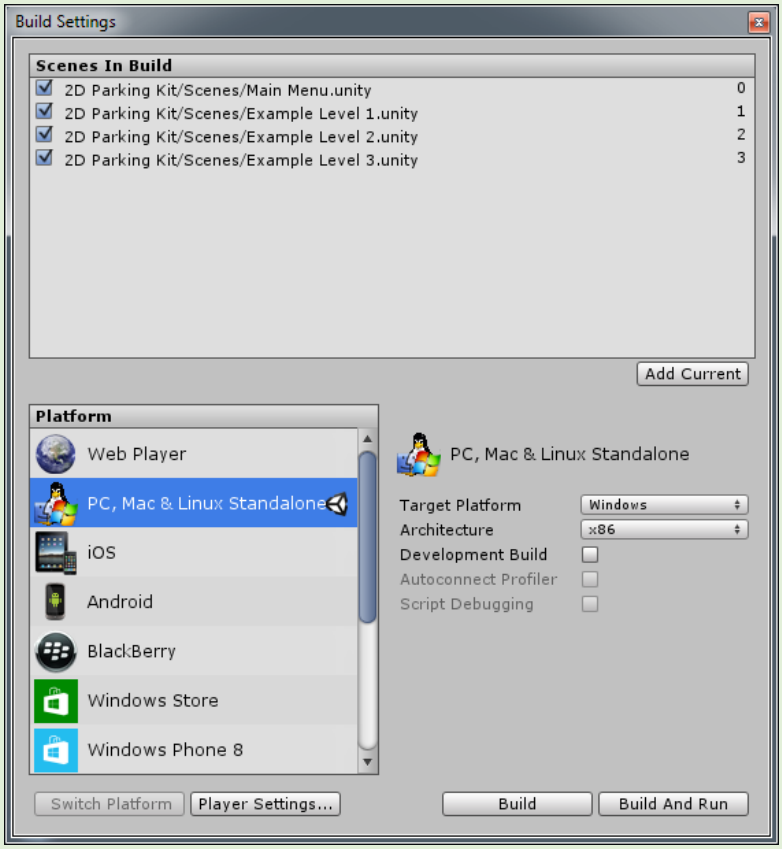
Finally, we go into the button component of the same object and change 1 value in the 'On Click ()' section of the component (as highlighted on the left) Again this value will correspond with all the previous steps, which is in my case 3.

You should now have something similar to that on the left.

Step 2: Creating the new scene:

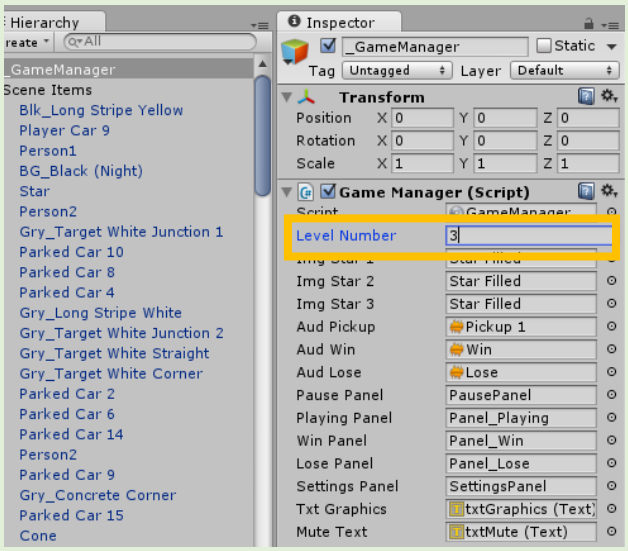


Now we need to create the new scene for the level. Firstly select one of the Current example levels and press Ctrl + D again (or CMD + D) to duplicate the level, this will also duplicate the scripts and user interface in that level and is much easier and less time consuming than starting from new.

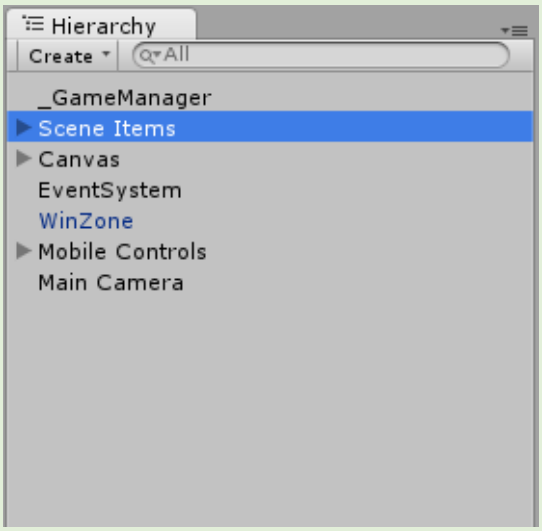


Next press CTRL + SHIFT + B to open the Build Settings Menu. (mac: CMD + SHIFT + B) when on this window drag and drop the new scene from the project view into the “Scenes in Build” window.

NOTICE: Make sure the build settings follows this structure, with the ‘Main Menu’ being the first (the 0 slot) and then all following levels being in the correct order after this, (Level 1 in the 1 slot, 2 in the 1 slot…) This is important and the script managing the level scripting relies on this order.



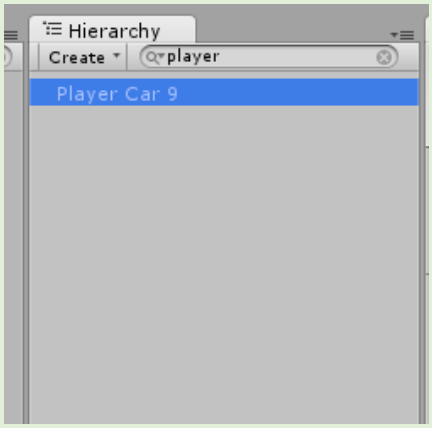
Now open the scene you created and click on the _GameManager object, change the Level Number in the script to correspond with the scene number (Again in my example the number is 3).



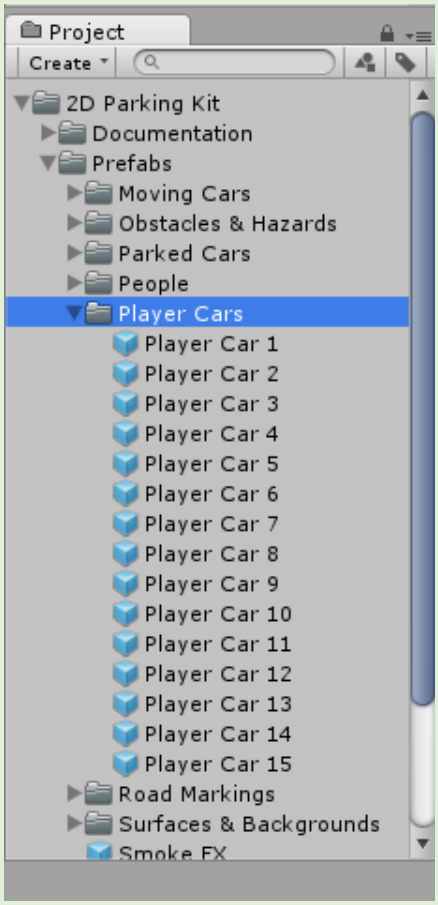
If you want to have a completely blank level with no player, obstacles, cars, stars etc. brought over from the previous level, firstly double click the 'Scene Items' GameObject to bring the camera view into focus, then Highlight the 'Scene Items' GameObject and press the delete key to remove it.

Step 3: Populating the level

The Player



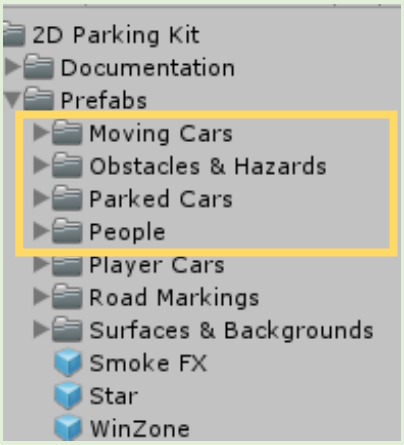
To begin populating the scene you will want to create the player, in earlier steps we copied a previous scene so unless you chose to delete the 'Scene Items' in the last step, this one will already have a player, to delete it search for *player* in the hierarchy tab, select the result (something similar to the left) and press the 'Delete' key on your keyboard to delete it.



Next, in the project tab navigate to 2D Parking Kit > Prefabs > Player Cars and select one of the types of car (there aren't any differences in the cars aside from the sprite that they use, so just one that you like the look of.

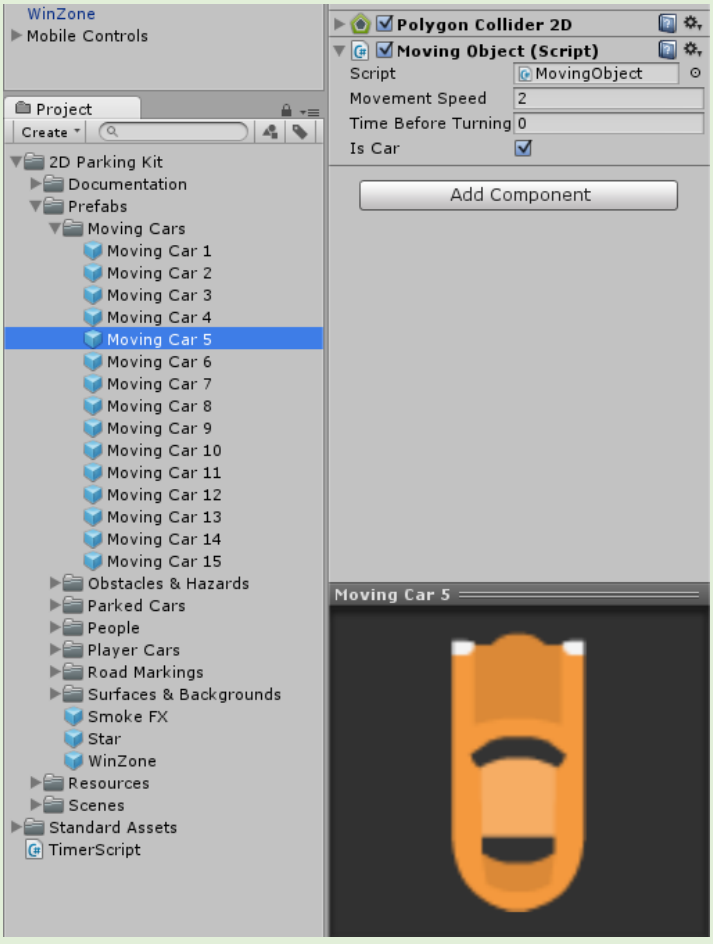
Drag and drop this prefab into the scene view window or Hierarchy tab.

The Obstacles



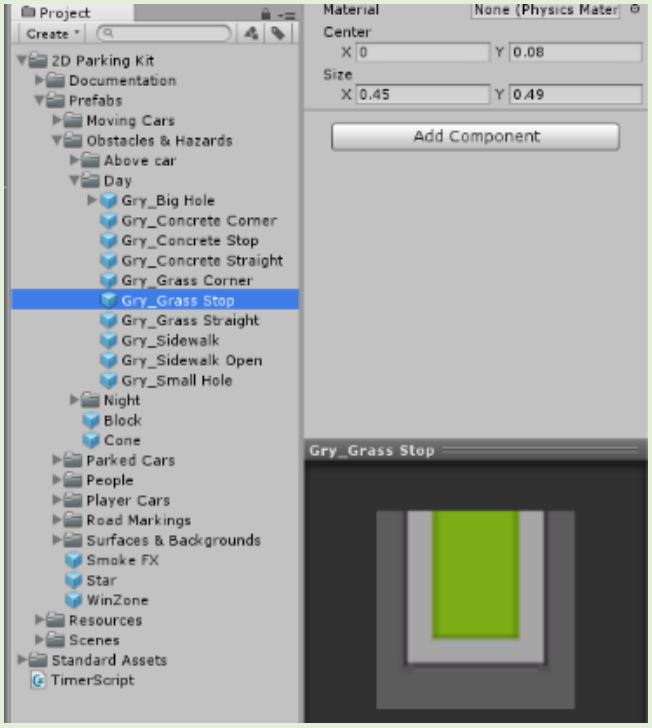
-'Obstacles' refer to several different sets of prefabs and can be found in the 'Moving Cars', 'Obstacles & Hazards', 'Parked Cars' and 'People' folders. All of the obstacles found in these folders are prefabs, which means they are already scaled and have the appropriate components (colliders, scripts etc.) to work with little-to-no adjustments needed. You can just drag and drop the items into the scene and after positioning them, in most cases they are ready to go.

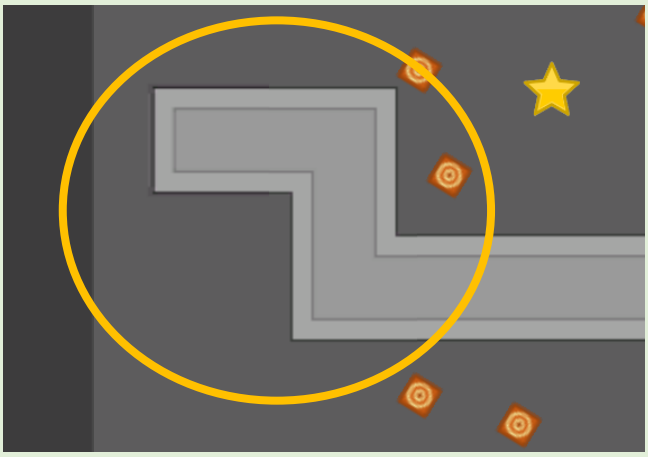
i) Moving Cars:



These obstacles are just like the player, or the parked cars in appearance, however they have a script attached called "Moving Object". This gives the car a special purpose in that it drives around in a loop. The 'Movement Speed' value is the speed at which the car travels and the 'Time Before Turning' value is the amount of time (in seconds) that the car takes to turn around and travel the other way. The car turns around based on its direction, so initially the car should be on the right side of the road when traveling up the screen, and on the left side when traveling down the screen, this allows the script to turn the car to the other side of the road.
Getting these cars to work can be fairly tricky at first, balancing the movement speed as well as the time before turning values to make sure the cars only turn when they are off screen, and don't overlap each other can require a bit of trial and error. The main menu scene has an example of moving cars, it may be a good idea to observe the values on each car.

ii) Obstacles & Hazards:

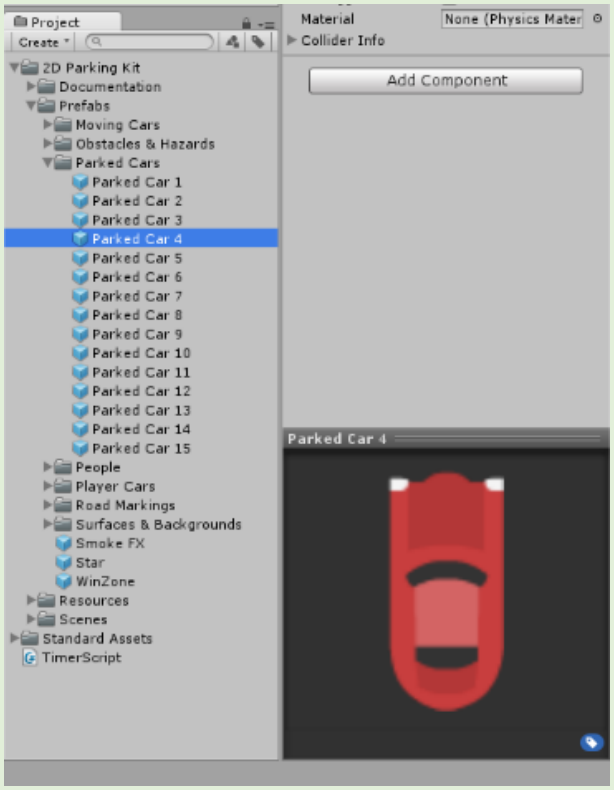




In the “Obstacles & Hazards” folder you will find prefabs that cause a ‘Game Over’ when the player collides with them. This includes Blocks, cones street lamps, trees / bushes, sidewalks and walls / dividers. The majority of these come in DAY or NIGHT versions, aside from those that do not have backgrounds (such as trees, cones, blocks and street lamps) day obstacles have a light grey backgrounds, night have black backgrounds.

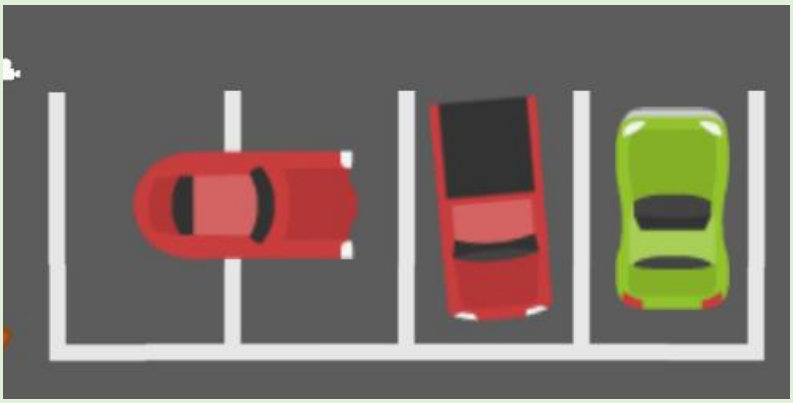
Again, to place these objects in the scene just drag and drop them, they have no scripts with values to be adjusted so after positioning them they are good to go.

iii) Parked Cars:

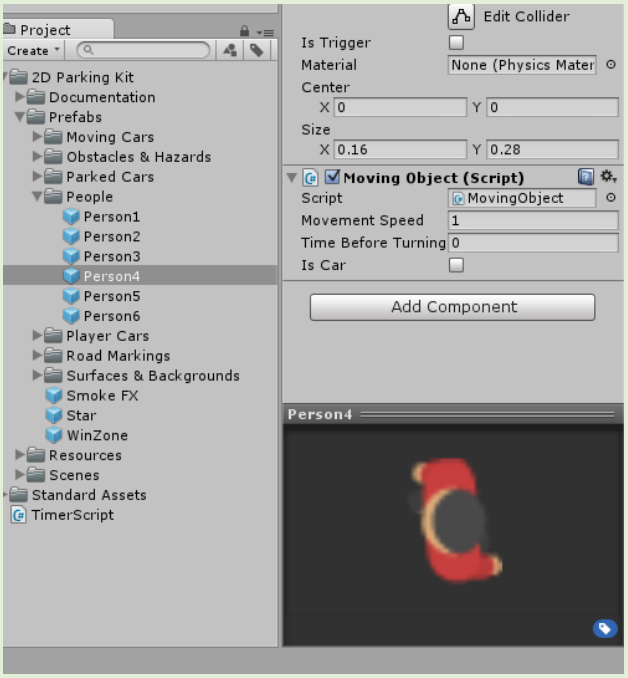


There are 15 different types of visual car, similarly to the moving cars and the player cars the differences are only visual. These are obstacles that are parked in spaces that player must avoid, hitting them will result in a Game Over.

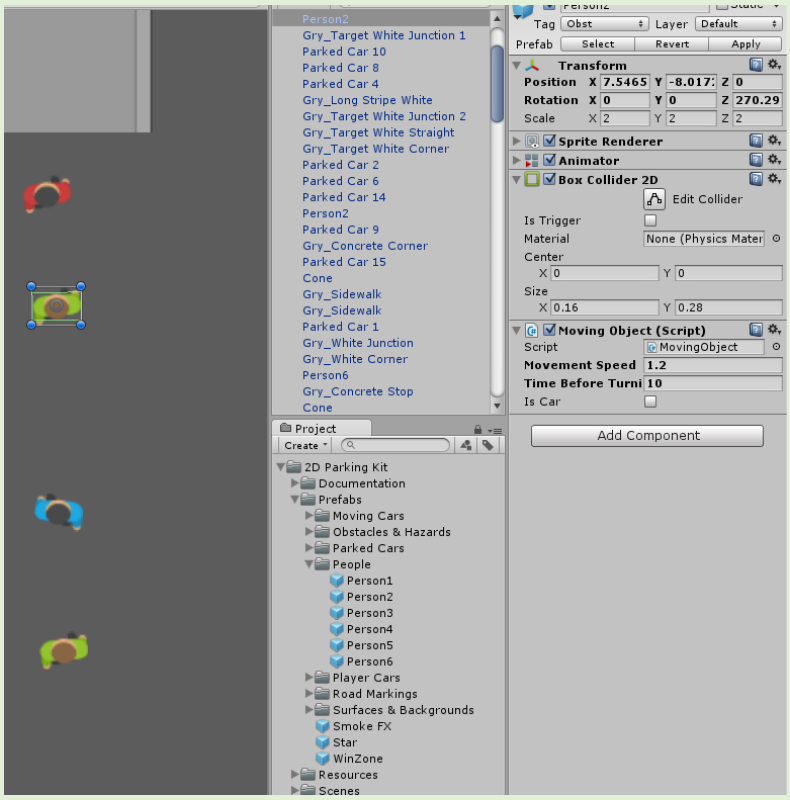
Just drag and drop the prefab into the scene view and after positioning it is good to go.



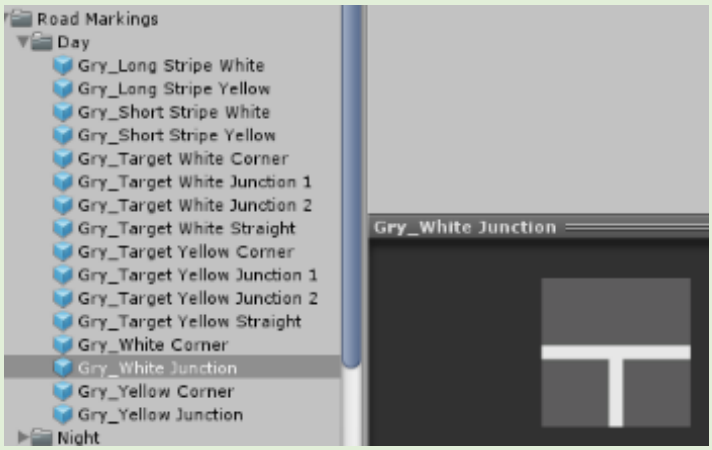
iv) People:



The 'People' obstacles work in a similar way to the moving cars, however they are usually positioned on sidewalks. They also have the 'Moving Object' script attacked to them. As with the moving cars, it will be necessary to adjust certain settings so that the people turn outside of the view of the camera, giving the illusion it's a different person walking from the opposite way. The first example level uses the people obstacles so you may want to observe the settings used there to perfect those used in your own project.

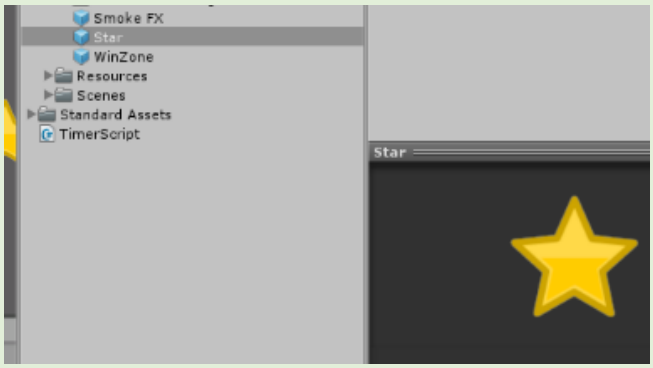


Road markings



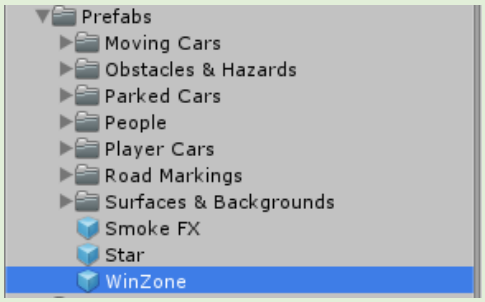
Different road markings are positioned together to create the spaces that parked cars are placed in. There are also markings that are ‘Target’ markings, these have an orange sections that shows the player where they can park.
These markings are also available in DAY and NIGHT varieties. With day having a light grey floor and night having a black floor.

Stars

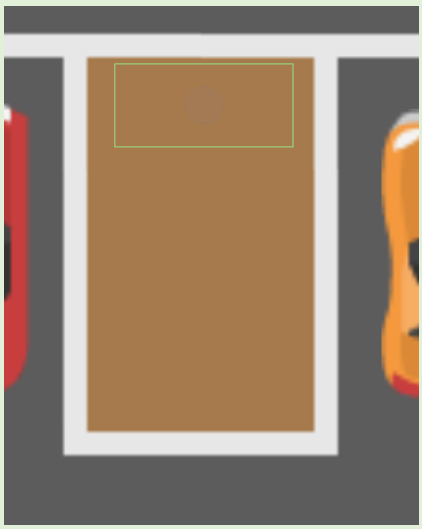


The star objects are what the player collects for the star system to work, it is not mandatory for the player to collect any stars and they could just park in the target spot without doing so. However on the menu it will show how many stars have been collected for each level.
As with all other prefabs the star can just be dragged into the scene view / hierarchy tab and positioned.
NOTICE: There must only be 3 stars per level for the star system to work correctly.

Win Zone



The Win Zone is a prefab that is essentially just a box collider with a tag (“WinZone”).

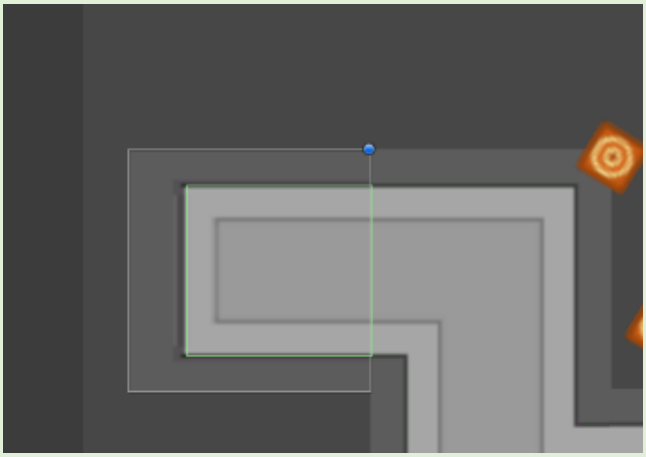


Drag and drop this prefab into the scene and position it so that the collider box is at the very end furthest away from the entrance of the target space. As shown in the image to the left (green outline rectangle is the collider component on the WinZone GameObject.

Step 4 : Positioning tips

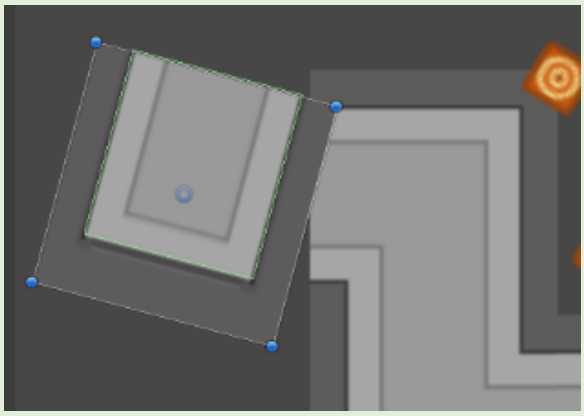
These tips require the rect tool to be switched on, this is found in the top left hand corner of the unity window.

Vertex Snapping



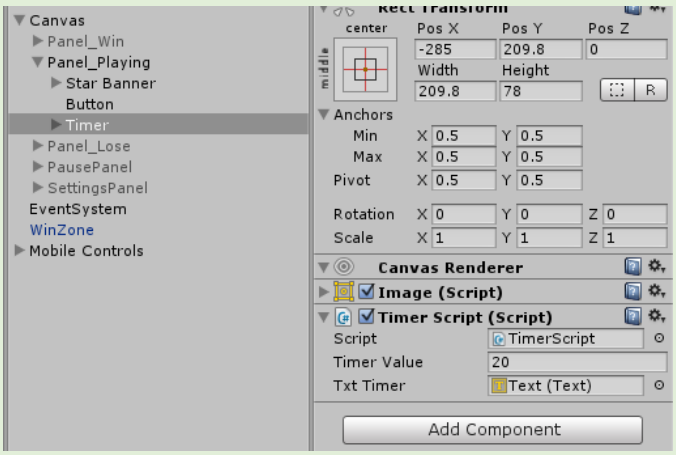
By holding the V key on your keyboard you activate the vertex snapping feature, this can save you a great deal of time where working on projects such as these. With V held down hover your mouse over one of the corners of the sprites you can then snap this corner (or vertex of the sprite) onto any other vertex in the scene on any other object. The image on the left shows me snapping the top right vertex of the end wall onto the top left vertex of the corner wall.

Angle Snap Rotation



Activate Angle snap by holding the ‘Shift’ button. Hover the mouse slightly away from one of the corners and rotation will become available. With angle snap the rotations will snap to a certain amount of degree’s rotation, this means that you can rotate more accurately, differing from the free rotate which means you probably won’t be able to rotate to perfect 90 degree / 180 degree angles.

Step 5: Timer Settings



After creating your level you will want to adjust the timer settings to reflect the difficulty of the level. The 'Timer' GameObject is automatically used if timed mode is selected from the main menu, and is automatically hidden if casual mode is selected. Find the 'Timer' GameObject by either searching for it in the Heirarchy tab, or by navigating to it through Canvas> Panel_Playing >Timer The only value you need to change is the 'Timer Value' on the Timer Script. This will be up to you, play through the level a few times and see how many seconds it takes you, then take off a second or two to make it more difficult.

Contact:

Please feel free to contact me at megruda@hotmail.com With any questions, or to suggest features you wish to see in future updates.

Credits:

Thanks to:

kenney.nl

1001.com

Photoshopwizard

Hamza Trabelsi

From opengameart.org for providing the majority of the art.

All other aspects developed by ElectricLoft.