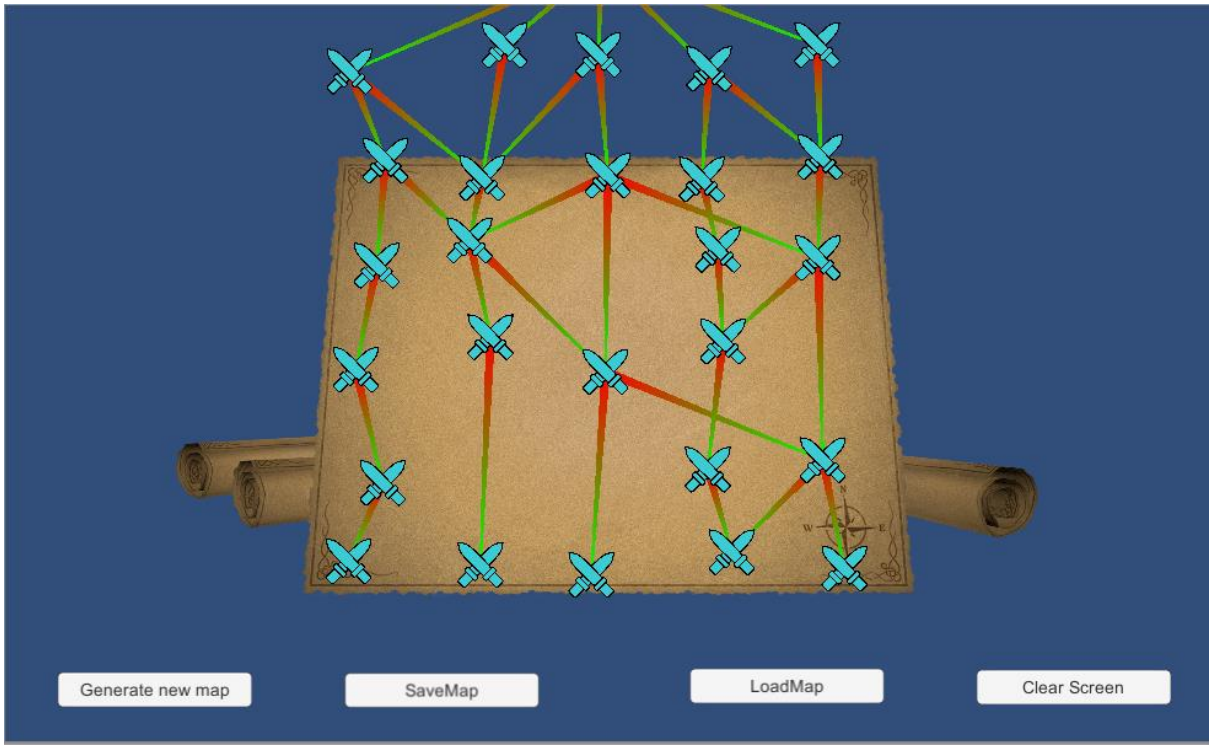
# World Map Generator


By Rare Camel Studios

RareCamelStudios.com

RareCamelStudios@gmail.com




# Contents

# 1.0 Getting started

To start using the asset, simply drag the 'Map' prefab, found in WorldMapGenerator/Prefabs onto your scene. Add two tags "Line" and "Node" into your project. This is all that is required! You can now generate maps by clicking on the map object and clicking 'Generate' at the bottom of the script. You can also drag 'ButtonFunctionality' found in the prefabs folder onto the scene for extra functionality, including Generating, saving, loading and clearing the map. You may need to reassign the buttons. If so, link the map object to your buttons by adding a listening (+ button). Drag the Map object onto each listener. I have listed which button should link to what function below. Change the settings on the map object to fit your map style.

Generate Button: MapGeneration.BuildNew

SaveButton: MapGeneration.SaveMap

LoadButton: MapGeneration.LoadSavedMap

ClearButton: MapGeneration.ClearNodesAndLines

# 2.0 The fields

There are over 27 fields you can change in the map to create the perfect map for your game. This section will detail each variable and how they effect the map generation.

## 2.1 Transforms

**Node prefab:** The prefab that will spawn at each node. You can swap this out with your own and write extra functionality for your own prefabs.

**Gen Start Pos:** The point where the map starts generating. Move this to move the location of your map. Put this at the bottom left of where you want your map to be.

**Nodes parent:** The Transform that nodes are parented to upon creation.

**Lines parent:** The transform that lines are parented to upon creation.

## 2.2 Map size, distances and positions

**Center top nodes:** A bool that determines whether or not the top row of nodes will be centered. Recommended to be set to true.

**Squash rows in:** If you have manually set a row size that is bigger than there are columns, this bool will ensure the nodes are squashed inside the map and not run across the screen to the right.

**Min/Max columns:** The minimum and maximum amount of columns that the map will generate. Columns are vertical. Set the min and max to the same to generate a fixed amount of columns. The amount of columns determines the amount of nodes in the rows.

**Min/Max rows:** The minimum amount of rows the map will generate. Rows are horizontal. Set the min and max the same to generated a fixed amount of rows.

**Row dist:** The amount of distance between each node horizontally. Set higher for the horizontal rows to be further apart.

**Col dist:** The amount of distance between each column. Set higher for the vertical columns in the map to be further apart.

**Min/Max X/Y Deviations:** The amount that each node will deviate from its designated position. Put all 0 to line up all the nodes perfectly. Recommended to include some deviation as it looks more dynamic.

**Line Z position:** The z position of the line renderers.

## 2.3 Node culling

**Cull nodes chance:** The default chance that any given node is culled.

**Bottom nodes cull chance:** The chance that each bottom node (row 0) is culled.

**Top nodes cull chance:** The chance that each top node is culled.

## 2.4 Node connections

**Always connect upwards:** Set this true to ensure all nodes connect to nodes above on the same column is there is one available.

**Nodes always have at least one forward connection:** Ensures all nodes lead upwards.

**Node connection chance default:** The default chance a node connections to any given node a row above it.

**nTh Column away connection chance:** Use this to manually set the connection chance based on its column distance away.

Example:

Size: 2

**Element 0**

Column number away: 1

Chance to connect: 25

**Element 1**

Column number away: 2

Chance to connect: 10

A setup like this means that given a node, nodes above it that are one column distance away have a 25% chance to connect, and nodes above it that are two column distances away have a 10% chance to connect. Use ‘nTh Plus’ for all distances greater than a certain amount.

**nTh Plus:** Does the same thing as above. However it will include the set column distance and any column distance greater than the column number away.

Example:

Column number away: 3

Chance to connect: 5

This will mean that all nodes on the above row that are 3 columns away or more have a 5% chance to connect

### 2.5 Row sizes

**Top row nodes:** Sets the amount of nodes in the top row. Ignore ‘RowNum’ field. Set Node num to 0 to generate the top row like a normal row. Use ‘TopNodesCullChance’ in conjunction with this.

**User Entered row amount:** Manually set the amount of nodes in a row.

Example:

Size 1

Row num: 3

Node num: 5

Cullable: False

This will generate five nodes on row 3 and stop them from being culled.

## 3.0 Classes

**Node:** The class which holds all the information about a node so it can be built later.

**RowInput:** A class that allows users to manually enter row input.

**NThAway**: A class used to allow the user to manually enter the connection chance by column distance.

**MapState:** Holds a list of nodes that determines the map state. Also used for saving and loading.

## 4.0 Scripts

**MapGeneration.cs:** Where all the magic happens. All fields and functions are commented. If you need further help, contact me (details on the first page).