

# 2D Map Generator Tool

---

Unity tool for procedural generation of two-dimensional maps.

## Content

Description.....	1
About tool.....	1
About map generation .....	1
Map examples .....	2
Usage.....	3
Editor.....	3
Data models.....	5
License .....	6

## Description

---

### About tool

The tool creates opportunity to generate maps in design mode and at the start of the scene. It has an editor that enable you to parameterize the generator and add new data models without programming. The parameterization of the generator has a big impact on the final effect of the generated map. It enable you to generate a map from the given seed of randomness, so it is possible to generate the same map for the same parameter.

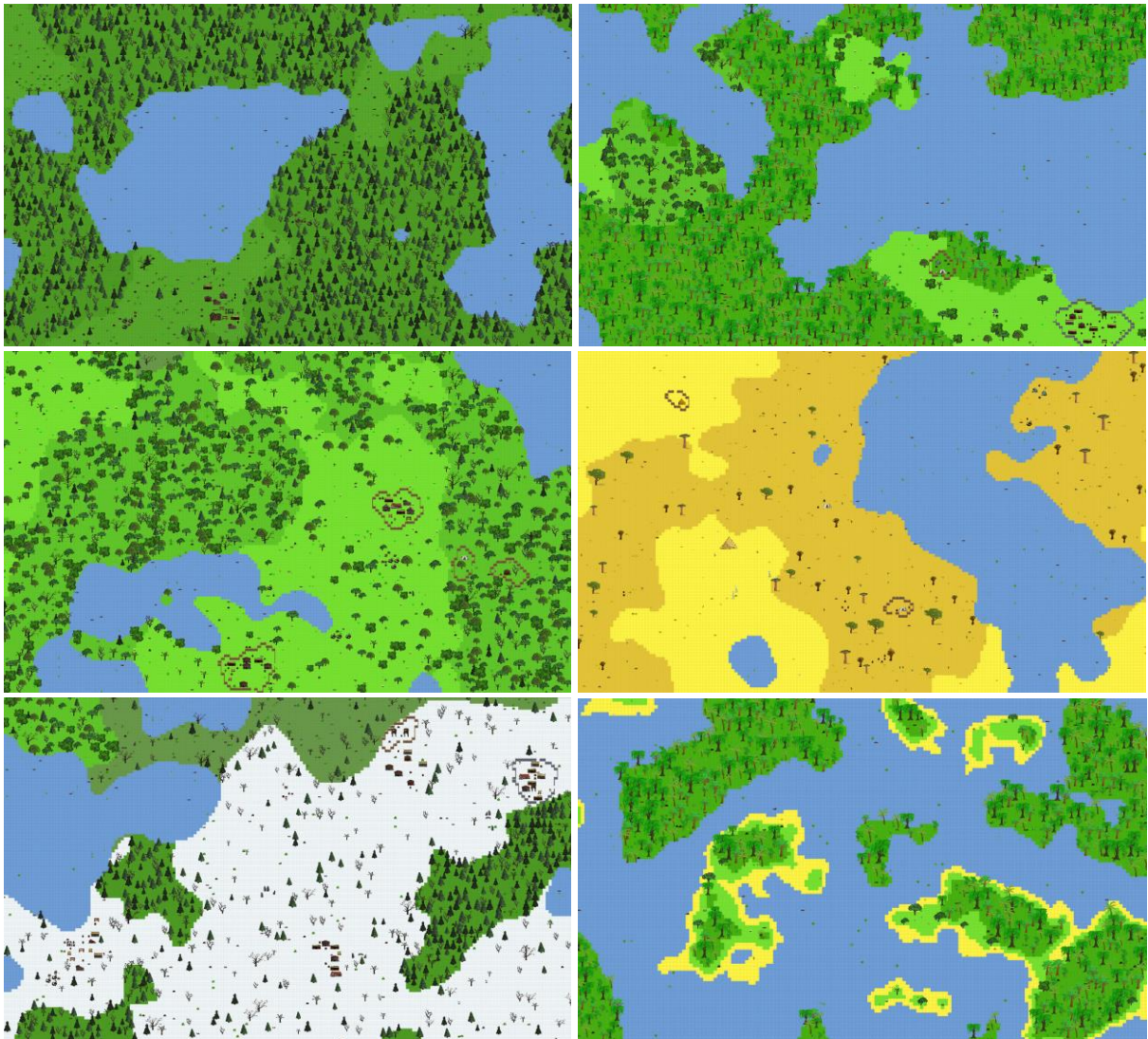
### About map generation

The generator's operation is based on [noise maps](#). First, height and temperature noise maps are generated. For each cell on the map, based on the values from these two maps, a particular biom is calculated from the prepared biomes diagram. Thanks to this, the transition between biomes on the map looks natural and diverse at once. For a generated map consisting of biomes, a water layer is generated, also created using a noise map. Finally, objects on the map such as locations, vegetation and other objects, with each biom having its own objects that will be generated on it.

# Map examples

---

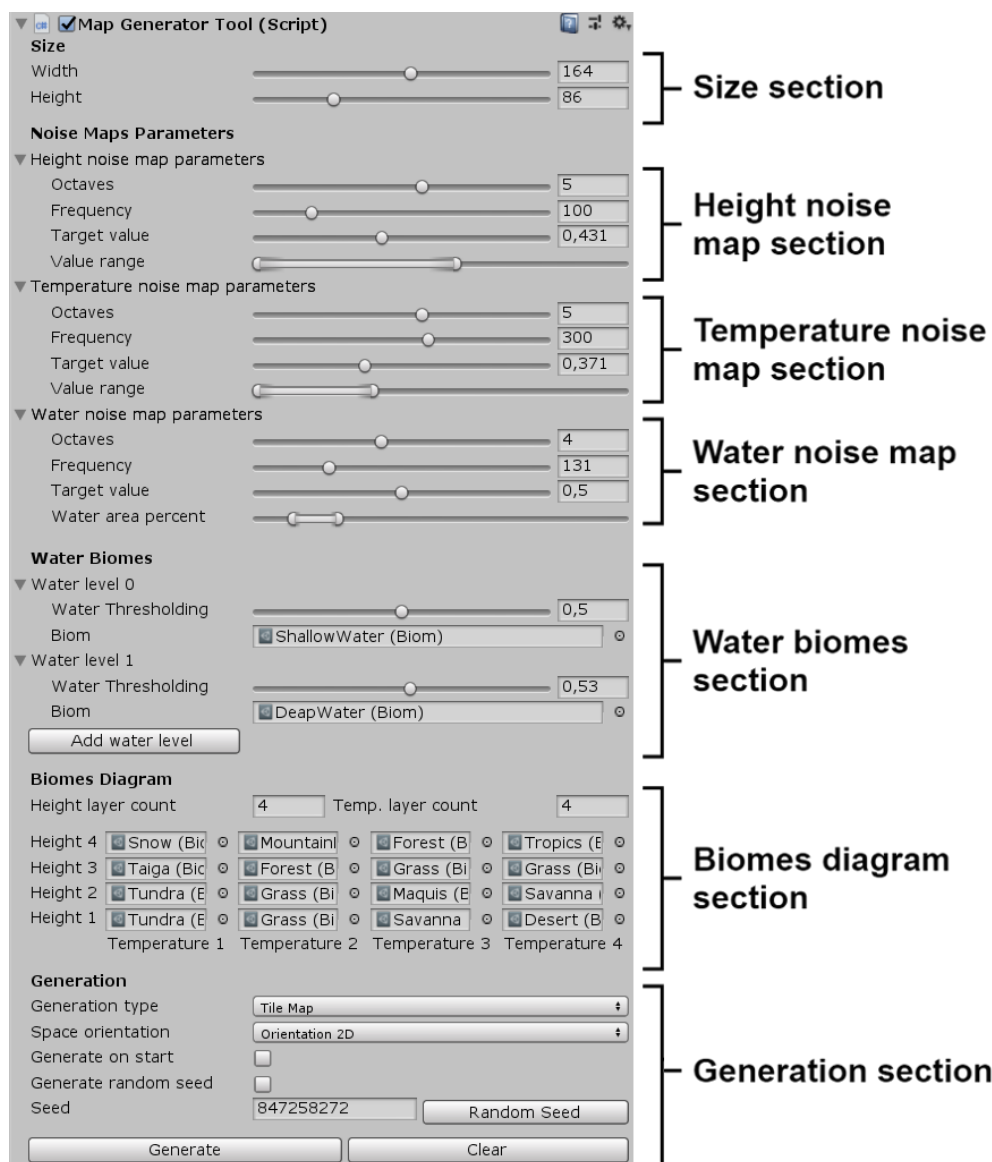
Below you can find some examples of generated maps for different generator parameters. The tool enable you to add new objects, define your own biomes and locations, change graphics or enter your own graphic style. I encourage you to experiment and create your own worlds.



# Usage

The tool has been designed in an easy way to use. Anyone can use it without necessity to edit the source code. The tool consists of: a generator editor that allows you to parameterize the generated maps and data models that represent specific objects such as biomes, locations or individual objects.

## Editor



### Size section

The tool enable you to generate maps in two-dimensional space. In this section you can set the sizes of the generated map.

### ***Height & Temperature noise map section***

In these sections you can modify the parameters of temperature and height noise maps. The generated values from these two noise maps will determine which biomes will be assigned to specific cells on the map.

**Octaves** affects on how blurry the adjacent biomes zones will be. The greater the value, the more they will be scraggy.

**Frequency** affects on how diverse the biomes generated will be. The lower the value, the more diverse they will be.

**Value Range** makes you possible to change the generated noise map values to the given range and to generate only specific biomes from the entire spectrum on the map.

**Target Value** enable you to shift the generated values to one side of the spectrum in such a way that certain values appear more often.

### ***Water noise map section***

In this section you can modify the parameters of the water noise map. It has the parameters mentioned above. The appearance of the water layer will depend on the values generated here.

**Water area percent** affects on the minimum and maximum percentage of water on the map.

### ***Water biomes section***

In this section you can define further layers of water depth on the map. The generator makes it possible to create separate layers for shallow and deep water or even to simulate islands.

**Water Thresholding** defines the threshold for the values from the water noise map after which this layer will be selected. Each subsequent layer of water must have a higher threshold value.

**Biom** assigned here will represent a given water layer.

### ***Biomes diagram section***

In this section you can define a biomes diagram, set its size and assign biomiom models to the appropriate cells in the diagram. The values assigned here have an influence on which biomes on the map will be adjacent. Biomes on the map are selected from the biome diagram based on the values from height and temperature noise maps.

## Generation section

In this section you can set the details of the map generation and generate it via the button.

**Generation type** enable you to choose whether the objects are to be generated on the tile map or maybe as sprites.

**Space orientation** enable you to choose whether the map will be generated in 2D or 3D space.

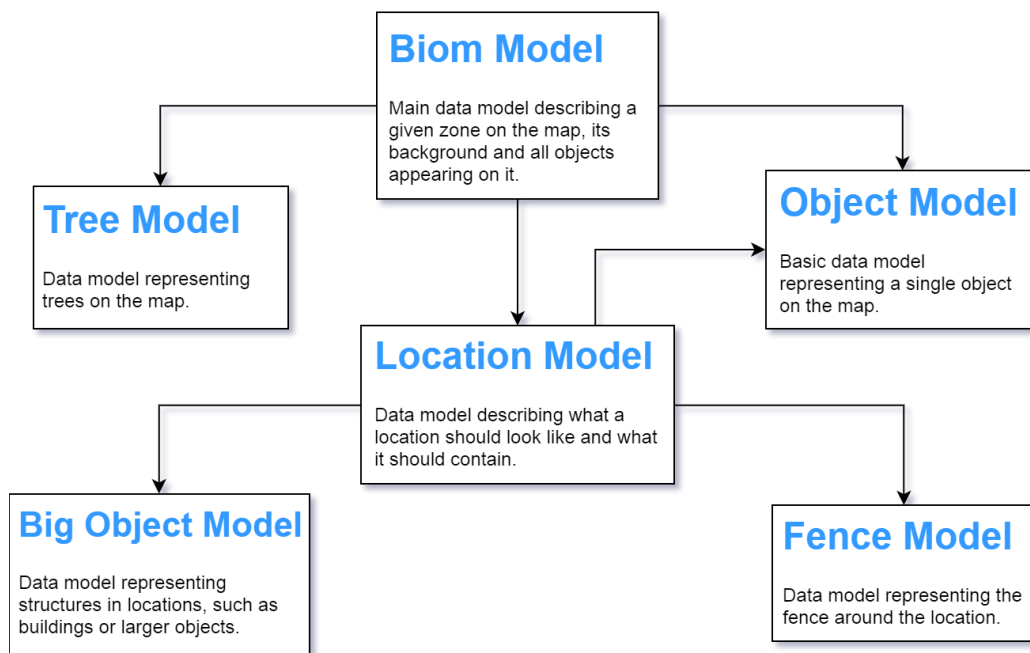
**Generate on start** allows you to choose if the map should be generated at the start of the scene.

**Generate random seed** allows you to choose if the map should be generated on the basis of a random seed or if it should be entered.

**Seed** allows you to enter a specific seed of randomness, thanks to which an identical map can be generated for the same parameters.

## Data models

Data models are used to keep information on which the map generator operates. They are saved in Unity using the [Scriptable Object](#) mechanism, thanks to which one prepared data model can be used many times. The tool has several types of data models that can be presented in a hierarchical scheme.



To create a new data model in the Unity project, press PPM in the project view, select Create -> Map Generator from the context menu, then the specific data model type. In the uploaded project you can see examples of the use of such data models.

**Intensity** determines the chance of generating an object for each individual cell on the map, if conditions allow for it. It does not specify covering the map itself with a given type of objects.

**Priority** defines the chance for this object to be generated from the entire collection of objects. The greater the priority, the greater the chance. The roulette wheel selection mechanism is used to draw an object from the collection.

**Max Count** specifies the maximum possible number of occurrences of a given object under the parent's object.

## License

---

The author of this tool is **TukanHan** and it is released under the **MIT** license. You have permission to do what you want with it without asking me for permission, but if you appreciate my contribution, please acknowledge the authorship.

If you like this project, support my work by sharing the project with other users so that they can also use it.

You can also view this project on [my GitHub](#).