Infinite Spawn Asset

For detailed instructions on how to set it up, code documentation, and feature list, please visit [this website](https://drakconianproject.wordpress.com/infinite-spawn-asset/).

Under InfiniteSpawnPlane -> DemoScene -> Prefabs, there is a prefab with the spawn plane attached. Simply drag and drop it into your game and go through the link above (Quick Start section) to set the parameters.

Alternatively, you can also open up the included demo scene and simply press play to see the spawner in action (as well as how things are set up). The included demo scene is a great tool for debugging the spawn settings.

**Quick Start Guide**

Open the InfiniteSpawn\_Demo scene; this will be used as an aid to optimize the spawn settings.

The follow objects are included in the demo scene:

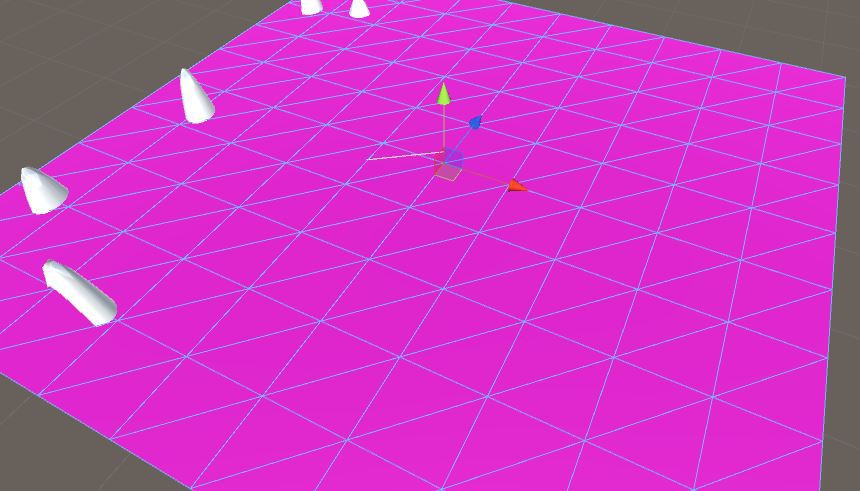
* **aiSpawnPlane:** The spawner itself, in-scene game object that has randomGenerationScript.cs attached.
* **Terrain:** The in-scene Terrain; ensure your games terrain has a Terrain layer set, and that the randomGenerationScript also includes this Terrain layer under LayerMaskRaycast.
* **Player:** Drag and drop the in-scene player into the PlayerToFollow spot in the inspector so that the plane knows who to follow.
* **Enemy Purple, Yellow, and Blue:** These are the enemies to spawn. Use these to visually see where the spawner is spawning things in the demo scene.
* **Obstacle 1, 2, 3:** These represent rocks, tree’s, or any other obstacle in your game that you don’t want spawns to occur in. Make sure Obstacles are set to an Obstacle layer, and that LayerMaskNoSpawn is also set to that same layer.

Quick explanation of settings:

* **PlayerToFollow:** The game object the spawn plane follows.
* **aiPrefabsToSpawn:** An array set by you (using the size parameter) to the number of various enemies you wish to spawn. WARNING: Make sure if you set this array to a certain size, that you set all the arrays to that same size in the inspector. The script uses the size of this array to determine many parameters.
* **SpawnTimer:** The time in seconds between each spawn. Increasing this will decrease the rate at which each particular enemy spawns. The element in the array corresponds to the element in the aiPrefabsToSpawn array. (For example, #3 in the aiPrefabsToSpawn array will use the #3 Timer)
* **Spawn Ceiling:** The maximum height (using Y-Axis) that spawns will occur.
* **Spawn Floor:** The minimum height (using Y-Axis) that spawns will occur. (Useful for preventing underwater spawns)
* **Instantiate Height:** How high off the Terrain the spawn will occur. Useful if you have enemies of varying sizes and you are finding some are spawning halfway underground.
* **LastSpawnTimer:** Debug tool, use this to verify spawns are actually occurring.
* **LayerMaskRaycast:** This is looking for your Terrain layer. Ensure your Terrain has a layer, and that this is set to it. (Where you want spawns to happen)
* **LayerMaskNoSpawn:** Obstacles, rocks, tree’s, etc. that you don’t want spawns to occur on/in. Ensure your Obstacles have an Obstacle layer and this is set to that same layer.
* **Set Distance (Player Buffer Zone):** A float value used to prevent spawns from occurring a certain distance from the player. Raise this to force spawns away from the player, decrease to have spawns occur near the player.
* **SteepnessBufferAng (Terrain Steepness):** Use this to prevent spawns from occurring on steep surfaces/cliff-sides. A lower value means spawns can occur on very steep surfaces, a high value means a surface must be extremely flat.

Next, run the demo scene for a little while. It should populate with spawns and show you visually how your settings are impacting the spawn rate/spawn locations.

For debugging, you can also set the mesh renderer for the spawn plane to active and be able to see visually the random spots that are being selected along the plane:

When the mesh renderer is activated for the spawn plane, you can visually see the random spawn calculations as they happen. Each point where the ray impacts the plane, is a point where a ray is casted downward towards the terrain.

You can reach me at this [thread post](https://forum.unity3d.com/threads/dragon-survival-game-the-drakconian-project.448306/#post-2900926) in the Unity Forums.

If you would like to support my work, find me [on Patreon](https://www.patreon.com/skyedragon).

Thank you for using the Infinite Spawn Asset!

SkyeDragon

Email:

Skyebound404@gmail.com