Scripting Documentation

Contents

Thank you for purchasing GenX 2D framework!

This documentation will guide you through everything you need to know to start creating your own procedural worlds.

When working with GenX don’t forget to add namespace ‘GenX2D’ in your script!

MapManager2D

Fields:

public static mapManager - Instance of 'MapManager2D' script in the scene for easier access.

public float squareSize - Size of 1 square in Unity units.

public bool useSeed - If unchecked - seed is random.

public static int realSeed –Real generation seed.

public static System.Random random - Global instance of System.Random class.

public List<Structure2D> allStructures - List of all structures.

public bool generateColliders - If checked colliders will be generated for the output map.

public bool allowDestruction - If checked - map will be destructible.

public bool smoothMesh - If checked the edges of the mesh will be smoothed.

public Material meshMaterial - Material of the output mesh.

Cellular Automata settings:

public int width - Width of the cellular automata map.

public int height - Height of the cellular automata map.

public int deathThreshold - If during a simulation, a living cell has less neighbours than this value - it dies.

public int birthThreshold - If during a simulation a dead cell has more neighbours than this value - it becomes alive.

public int chanceToStartAlive - Chance that during map initialization a cell will be declared alive.

public int outputGeneration - How many times should the simulation repeat.

public int caveAliveThreshold - Any cave that has less empty cells inside of it than this value will be filled.

public int islandAliveThreshold - Any island that is made up of less cells than this value will be destroyed.

public static bool[,] lastMap - Last generated 2-dimensional map, where false means cell is dead, true - it's alive.

Terrain settings:

public int chunkSizeX - X-axis size of the chunk.

public int chunkSizeY - Y-axis size of the chunk.

public int blockTextureSize - Texture size of every block.

public int chunkUnloadDistance - X-axis distance from the main camera to a chunk, at which that chunk is destroyed.

public int caveSpawnLimitY - Caves will not appear higher than this value.

public List<BlockType2D> allBlockTypes - List of all block types.

public List<Biome> allBiomes - List of all biomes.

Dungeon settings:

public int minRoomSize - Min size a room can have.

public int maxRoomSize - Max size a room can have.

public int minRoomCount - Min number of rooms to generate.

public int maxRoomCount - Max number of rooms to generate.

public int borderThickness - Thickness of room border.

public int passageWidth - Width of passageways.

public int passageBorderThickness - Thickness of the passageway border.

public int roomFullness - How full is the room 0 - 100%.

public int valueOffset - First generation value.

public int valueDifference - Second generation value.

Methods:

public void RegenerateSeed () - Set new random seed.

public static GameObject CreateCave(Vector3 position, BlockType2D blockType, string _seed = null)

- Create a cave at position using values from 'MapManager2D' in the scene.

position - Position of the cave

blockType - Block type that will make up this cave

_seed - Generation seed

Returns GameObject of the created cave.

public static GameObject CreateCave(Vector3 position, BlockType2D blockType, int _width, int _height, int _deathThreshold = -1, int _birthThreshold = -1, int _chanceToStartAlive = -1, int _outputGeneration = -1, int _caveAliveThreshold = -1, int _islandAliveThreshold = -1, string _seed = null, bool smooth = false)

–Create a cave at position using input values (see ‘Fields’for info about the values)

smooth –Should the output mesh be smoothed.

public static GameObject CreateCave(Vector3 position, BlockType2D blockType, int _width, int _height, GenerationPreset preset, string _seed = null, bool smooth = false)

–Create a cave at position from preset values.

public static bool[,] AddStep()

–Returns current map with one more simulation step.

public static GameObject PrefabMap(GameObject prefab, bool[,] inputMap, Vector3 centerPos)

- Makes a map of prefabs from 'inputMap' at 'centerPos'.

public static Mesh DestroyCell(int xPos, int yPos)

- Destroys a cell at position.

Terrain methods:

public IEnumerator GenerateChunk(int xPosition, int xRadius, float loadSpeed = 1F)

- Asynchronously load chunks at x-axis position and radius.

public static IEnumerator UnloadChunk(int xPosition, int xRadius)

- Asynchronously unload chunks at x-axis position and radius.

public static void SpawnStructures(Vector2 offset, bool[,] map, List<Structure2D> structures, int[] peaks = null, Transform parent = null)

- Spawn structures on 'map' with offset.

peaks - Peaks of the chunk (If spawning on terrain)

parent - Parent transform of every structure

public static GameObject CreateDungeon(Vector3 pos, BlockType2D blockType = null, string seed = null)

- Create a dungeon at position ‘pos’, that is made up of ‘blockType’ blocks.

public static GameObject CreateDungeon(Vector3 pos, int _minRoomSize, int _maxRoomSize, int _minRoomCount, int _maxRoomCount,

int _roomFullness, int _valueOffset, int _valueDifference, int _roomSmoothness, int _minDistanceBetweenRooms, int _maxDistanceBetweenRooms, BlockType2D blockType = null, string seed = null, bool smooth = false)

- Create a dungeon at position ‘pos’ using input values (see ‘Fields’for info about the values).

CellularAutomata

Methods:

public static bool[,] Generate(int _width, int _height, int _deathThreshold, int _birthThreshold, int _chanceToStartAlive, int _outputGeneration, int _caveAliveThreshold, int _islandAliveThreshold, string _seed = null)

- Returns a generated 2D map using input values (See MapManager2D/Fields or editor documentation for info about the values).

public static bool[,] Simulate(bool[,] inputMap, int _deathThreshold, int _birthThreshold)

- Returns 'inputMap' simulated 1 more time.

public static bool[,] LargestCaveInArea(bool[,] inputMap)

- Returns the largest closed empty area in 'inputMap'.

public static bool[,] RemoveCavesAndIslands(bool[,] inputMap, int _caveAliveThreshold, int _islandAliveThreshold)

- Removes all caves smaller than '_caveAliveThreshold' and islands smaller than '_islandAliveThreshold'.

public static int GetNeighbourCount(int x, int y, bool[,] inputMap)

- Returns the number of neighbours of the input cell on the input map.

public static bool IsOutOfMap(int x, int y, bool[,] inputMap)

- Checks if input coordinates are out of the 'inputMap'.

Dungeon

Methods:

public static Room2D[] GenerateDungeonRooms(int _minRoomSize, int _maxRoomSize, int _minRoomCount, int _maxRoomCount, int _roomFullness, int _valueOffset, int _valueDifference, int _roomSmoothness, string _seed = null)

- Generates a dungeon and returns an array of roooms in it. (See MapManager2D/Fields or editor documentation for info about the values).

public static bool[,] GenerateDungeonRooms(int _minRoomSize, int _maxRoomSize, int _minRoomCount, int _maxRoomCount, int _roomFullness, int _valueOffset, int _valueDifference, int _roomSmoothness, string _seed = null)

- Generates a dungeon and returns the whole map of it. (See MapManager2D/Fields or editor documentation for info about the values).

public static List<Vector2> GetLine(int x1, int y1, int x2, int y2)

- Get a line between point x1,y1 and point x2,y2.

MeshMaker

Fields:

public static Mesh lastMapMesh

- Last generated mesh.

Methods:

public static Mesh MakeMesh(bool[,] map, float squareSize, bool smooth)

- Generate a 2D mesh from the input map.

public static Mesh MakeMesh(Vector2[] smoothedPeaks, float size, int yDepth)

- Generate a smooth 2D mesh from the input peaks.

public static Mesh MakeMesh(Block2D[,] map, float _squareSize, bool smooth)

- Generate a 2D mesh from the input map.

public static Mesh MakePlane(int width, int height, float _squareSize, int[] peaks)

- Make a plane below peaks.

public static void MakeCollider(bool[,] map, Transform parent = null)

- Make colliders from the input map.

public static Mesh DestroyFace(int xPos, int yPos, Chunk2D inputChunk = null)

- Destroy a face on the chunk or the last map.

MeshTextureGenerator

Methods:

public static IEnumerator GenerateTextureAsync(Block2D[,] map, List<BlockType2D> blockTypes, MeshRenderer targetChunk, float loadSpeed)

- Set texture of a chunk async.

public static Texture2D SetTerrainInside(Texture2D backgroundTexture)

- Set background texture of the terrain.

public static Texture2D GenerateTexture(Block2D[,] map, List<BlockType2D> blockTypes, float squareSize)

- Generate texture for a map.

public static Texture2D GenerateTexture(Block2D[,] map, BlockType2D tile, float squareSize)

- Generate texture for a map with only one block type.

public static Texture2D Scale(Texture2D source, int newWidth, int newHeight, FilterMode filterMode)

- Resize the input texture.

Terrain2D

Fields:

public static int currentBiome

- Current biome index

Methods:

public static int Noise(int x, int y, float scale, float mag, float exp)

- Get noise at position using the input values.

public static void SetBiome(int biome)

- Manually set current biome index.

public static Block2D[,] GenerateTerrain(int posX, List<BlockType2D> blockTypes)

- Generate block 2D terrain with x-offset.

public static int[] GeneratePeaks(int globalPositionX, int width, List<NoiseLayer> noiseLayers)

- Generate only height values from the input noise.

public static Vector2[] SmoothPeaks(int[] peaks)

- Returns smoothed points of the input peaks.