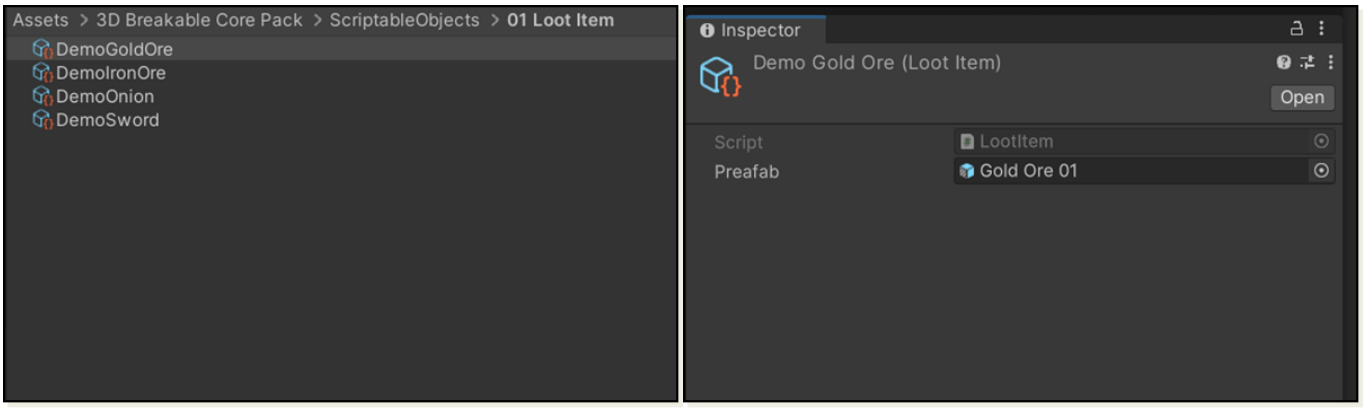
Loot

The 3D Breakables Core Pack offers an implemented solution for adding item drops to your breakable items. Solution is build around two types of ScriptableObjects: "**LootItem**" and "**LootTable**".

LootItem

A LootItem is a ScriptableObject that stores information about a lootable object.

Q: Where are some Examples?
LootItem ScriptableObject Instances created for purposes of included demo scenes, are located in 3D Breakable Core Pack → ScriptableObjects →01 LootItem

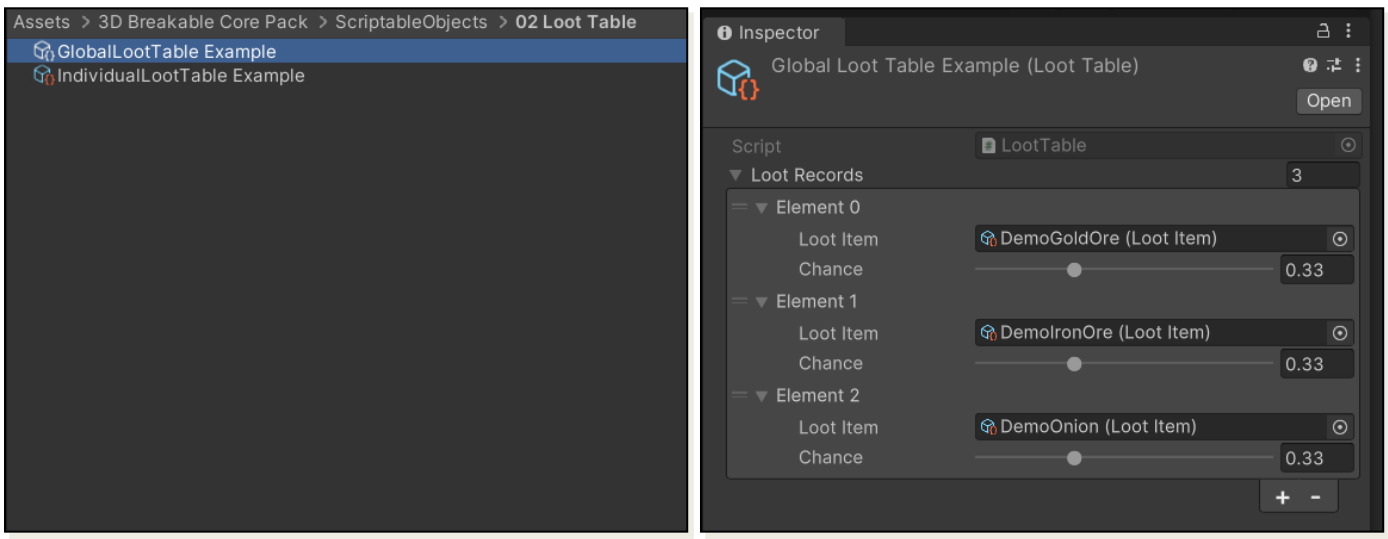


Q: How do I create a new "LootItem" ?
(1) Right-click in the selected project folder → Breakable Core Pack → Loot → LootItem (2) Select new added "NewLootItem"; (3) Change default name of new created "NewLootItem"; (4) Assign "Prefab" in Inspector Tab of new added LootItem;

LootTable

A LootTable is a ScriptableObject that holds an array of LootItems and their drop chances.

Q: Where are some Examples?
Instances of LootTable created for the demo scenes can be found in: 3D Breakable Core Pack → ScriptableObjects →02 LootTable

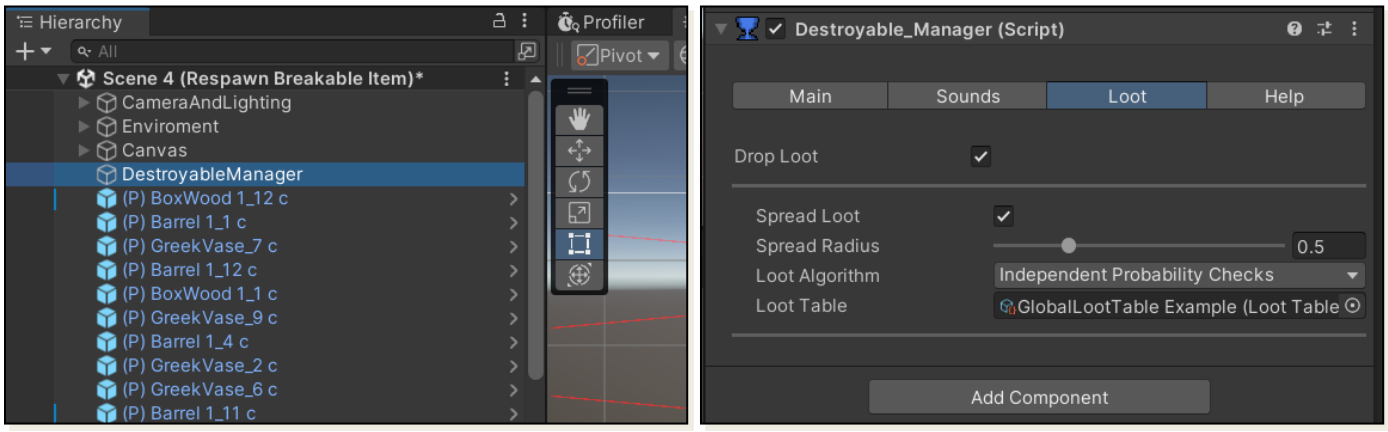


Q: How to add new “LootTable”?

(1) Right Click in selected Project Folder → Breakable Core Pack→Loot→LootTable;
(2) Select new added “NewLootTable”;
(3) Change default name of new created “NewLootTable”;
(4) Add required amount of Loot Records;
(5) In each Loot Record, assign a LootItem and its drop chance.

Global Loot Settings

Global Loot Settings are declared in “DestroyableManager” located in scene, under “Loot” Tab.



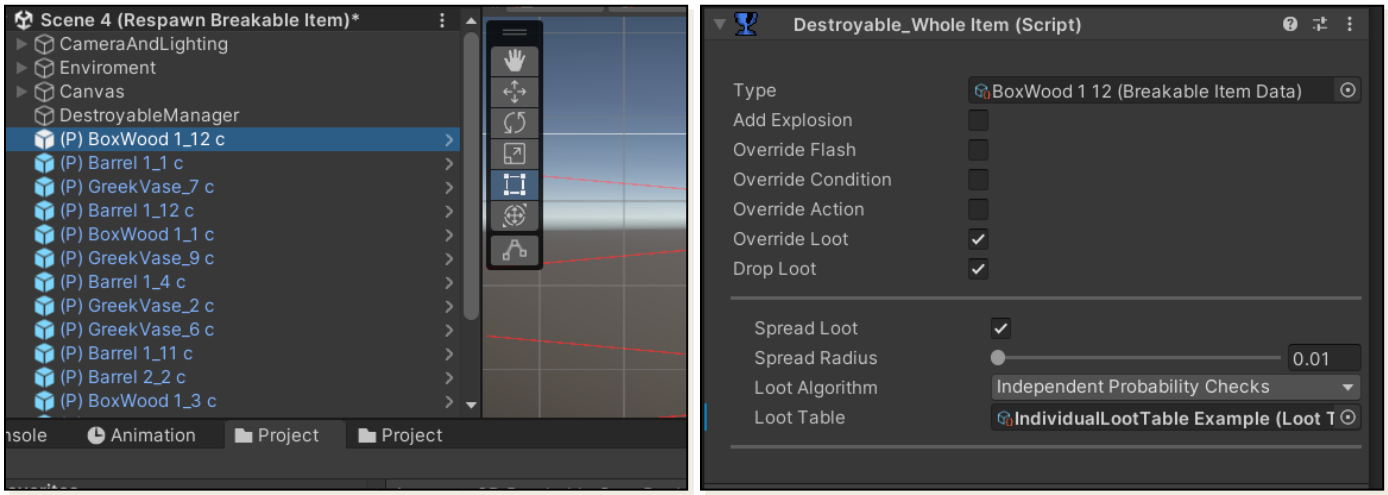
Q: What happens if there is no “DestroyableManager” in the scene?

If there is no “DestroyableManager” located in scene, it will be added when needed (lazy initiation). Manager will be Instantiated as copy version located in “Resources” folder. As a result settings included in default version will be applied;

Q: How to override default settings for some items?

Each item with a “DestroyableWholeItem” component has an ‘Override Loot’ option. When enabled allows for changing global loot settings:

(1) Enabling/Disabling Dropping loot;
(2) Enabling/ Disabling Spreading Loot;
(3) Declaring different Spread Radius;
(4) Choosing different Loot Algorithm;
(5) Assigning different from global LootTable;



Loot Algorithm- Independent Probability Checks

This algorithm evaluates each element in the LootTable individually. For each element, the algorithm performs a random check using the following condition:

- Random.Range(0,1) < Chance

If the condition is met, the LootItem is added to the spawned loot list.

Q: How does this algorithm work step by step?

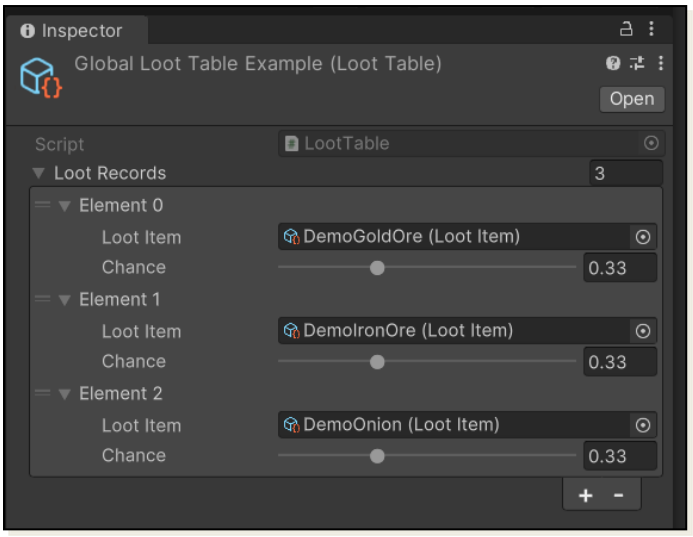
Let's conduct **theoretical** LootTable cast according to Independent Probability Checks.

(1) **LootItem "DemoGoldOre":** Random.Range(0, 1) resulted in 0.27. Since 0.27 is smaller than the Chance (0.33), the item **will be added** to the spawned loot list.

(2) **LootItem "DemoIronOre":** Random.Range(0, 1) resulted in 0.89. Since 0.89 is higher than the Chance (0.33), the item **will not be added** to the spawned loot list.

(3) **LootItem "DemoOnion":** Random.Range(0, 1) resulted in 0.07. Since 0.07 is smaller than the Chance (.33), the item **will be added** to the spawned loot list.

As a result of above **theoretical** cast, **2 items will be spawned**.



Loot Algorithm- Roulette Wheel Selection

This algorithm assigns each element in the LootTable a segment on a roulette wheel based on its Chance value. A single random number is generated to determine which segment is selected. The item corresponding to the selected segment is added to the spawned loot list. The process ensures that items with higher chances occupy larger segments, increasing their likelihood of being selected.

Q: How does this algorithm work step by step?

Let's conduct **theoretical** LootTable cast according to Roulette Wheel Selection.

(1) The algorithm calculates the total weight (sum of all Chance values): 0.33 + 0.33 + 0.33 = 0.99.

(2) A random number between 0 and 0.99 is generated, e.g., 0.75

(3) The algorithm iterates through the LootTable, summing the Chance values until the sum exceeds the random number:

   a. **LootItem** "**DemoGoldOre**" segment: 0.01 to 0.33
   b. **LootItem** "**DemoIronOre**" segment: 0.33 to 0.66
   c. **LootItem** "**DemoOnion**" segment: 0.66 to 0.99

(4) Since the random number (0.75) falls within the "**DemoOnion**" segment (0.66 to 0.99), "**DemoOnion**" is selected and added to the spawned loot list.

As a result of above **theoretical** cast, **always only one** item will be spawned.