Random Spawnable Objects By Level

Spawning Enemies Concepts

Enemy Spawning Example

Catacombs Large Room 1

Level 1		Level 2		Level 3	
Enemy	Ratio	Enemy	Ratio	Enemy	Ratio
	10		5		5
	10		5		6
			5		6
			5		3
					3
					3
					6

The absolute value of the ratios doesn't matter, it's their value relative to the other enemy ratio values that's important (a ratio of 1:1 is the same as 10:10)

Spawning Enemies Concepts

Generic Spawnable Objects By Level

```
List<chanceBoundaries> chanceBoundariesList
```

```
private struct chanceBoundaries
{
    public T spawnableObject;
    public int lowBoundaryValue;
    public int highBoundaryValue;
}
```

chanceBoundaries					
T spawnableObject		int lowBoundaryValue	0	int highBoundaryValue	4
chanceBoundaries					
T spawnableObject		int lowBoundaryValue	5	int highBoundaryValue	10
chanceBoundaries					
T spawnableObject		int lowBoundaryValue	11	int highBoundaryValue	16
chanceBoundaries					
T spawnableObject		int lowBoundaryValue	17	int highBoundaryValue	19
chanceBoundaries					
T spawnableObject		int lowBoundaryValue	20	int highBoundaryValue	22
chanceBoundaries					
T spawnableObject		int lowBoundaryValue	23	int highBoundaryValue	25
chanceBoundaries					
T spawnableObject		int lowBoundaryValue	26	int highBoundaryValue	31

Spawning Enemies Concepts

Generic Spawnable Objects By Level

```
List<chanceBoundaries> chanceBoundariesList
```

chanceBoundaries	T spawnableObject	int lowBoundaryValue	int highBoundaryValue
chanceBoundaries		0	4
chanceBoundaries		5	10
chanceBoundaries		11	16
chanceBoundaries		17	19
chanceBoundaries		20	22
chanceBoundaries		23	25
chanceBoundaries		26	31

ratioValueTotal = 32

```
int lookUpValue = Random.Range(0, ratioValueTotal);
```

So the lookUpValue will be a random value between 0 and 31. This is then used to retrieve the item from the list whose range includes the lookUpValue.