**Particle System**

**What is a Particle:**

* A particle is a small individual 2D Sprite or 3D mesh which undergoes various changes during its lifetime.
* Ex: In a rain effect each raindrop can be considered as a particle.

**Particle System:**

* Particle system emits a collection of particles within a specified shape.
* It is a system of multiple modules working together to create some particle effect.
* Particle systems are useful when we want to create dynamic objects like fire, water, smoke etc., because it is difficult to depict these kinds of objects with a Mesh or Sprite.
* Creating a particle system in Unity: GameObjectMenu-Effects-Particle System.

**Main Module in ParticleSystem Inspector:**

* **Duration:** for how many seconds the particle system should emit particles.
* **Looping:** Determines if the particle loops or plays only once.
* **Prewarm:** Only used when looping is enabled and the particle system acts as if it already completed a full cycle. Indicates whether a particle system should be simulated from the beginning or end of the loop.
* **Start Delay:** The delay in seconds before the particle system starts emitting.
* **Start Lifetime:** The initial lifetime in seconds for the particles. The particle will get destroyed after the elapsed time.
* **Start Speed:** The initial speed of the particles. The greater the speed of the particles the more spread out they will be.
* **Start Size:** The starting size of particles in all axes.
* **3D Start Size:** Indicates whether particle size should be controlled in each axis separately or not.
* **Start Rotation:** Indicates the initial rotation of particles in all axes.
* **Flip Rotation:** Setting the value to 1 flips the rotation angle along the horizontal axes.
* **Start Color:** Indicates the initial color of each particle.
* **Gravity Modifier:** Indicates the scale value for the gravity of the world. If set to 0 then gravity will be disabled.
* **Simulation Space:** Indicates whether particles should be animated in the object’s local space, in the world space or in the custom object space.
* **Simulation Speed:** the speed at which the particle effect simulates the particles.
* **Delta Time:**
* **Scaling Mode:** Indicates the mode of scaling particles; Hierarchy scales particles with respect to the scale of its parent. Local scales particles with respect to particle system scale.
* **Play On Awake:** Indicates whether the Particle system should be simulated automatically when the object is created or not.
* **Emitter Velocity:** Allies us to change the emitter velocity from RigidBody to Transform.
* **Max Particles:** Indicates the maximum number of particles allowed in the system at once. As the limit is reached, some particles will be deleted from the system.
* **Auto Random Seed:** Indicates whether the Particle system should look different or the same each time it is played.
* **Stop Action:** Determines if the Particle System will disable or destroy itself when the particle system is stopped.
* **Culling Mode:**
* **Ring Buffer Mode:** Keeps particles alive until they reach the Max particles count, at which point new particles recycle the oldest ones, instead of removing the particles when their lifetime elapses.

**Emission Module:**

* Emission module controls the rate at which particles are emitted and burst events are triggered.
* **Rate over Time:** Indicates after how many particles should be emitted per second.
* **Rate over Distance:**Indicates how many particles should be emitted per unit distance moved. (Works only in world simulation space)
* **Burst**: A burst is an event which emits n number of particles at once.
  + Time: Indicates after how many seconds it should trigger the burst event.
  + Count: Number of particles that should be emitted after ‘Time’ that was set.
  + Cycles: Count indicates a definite number of cycles the particles should be emitted after each time interval.
  + Interval:
  + Probability:

**Choosing particle system solution:**

* There are two types of particle system solutions are there in Unity
* **Built-in Particle system:**
  + It gives us full read/write access to the system through c# scripts.
  + We can use Particle System API to create custom behaviours for our Particle systems.
  + Particle systems can use Unity’s underlying physics system and thus interact with Colliders in the Scene**.**
* **Visual Effect Graph:**
  + It can run on GPU to simulate millions of particles and create large-scale visual effects.
  + The visual effect graph provides a visual graph editor to help us author highly customisable visual effects.

**OnParticleCollision() function:**

* OnParticleCollision is called when a particle hits a Collider.
* This can be used to apply damage to a GameObject when hit by particles.
* This message is sent to scripts attached to Particle Systems and to the Collider that was hit.