Space Shooter

This game is an implementation of Space Shooter in Unity using C# for scripting. The target platform is Windows. It's a simple 3D game, but with a 2D perspective. The UI is minimal: in the Start scene there is a `PLAY` button which loads the Game scene. In the Game scene, named Level01, there is a UIText that displays the score which increases when the player shoots an asteroid.

Player

- `Spaceship` object represents the logical layer for player. It has a Rigidbody, a Capsule Collider and a Spaceship Movement script. The script has some public variables which are set from the Inspector.
- `SpaceshipVisual` child object represents the visual layer, which is a Quad with a material on it.
- The Spaceship rotates itself to face the mouse and it is movable using Arrow or W, A, S, D keys. For shooting the player has to left click.

Game Controller

- The Game Controller is an empty object which holds a script for updating the score, for spawning asteroids, and to load the GameOver scene when the player collides with an asteroid. This script has a reference to the asteroid to spawn it somewhere between spawnValues after a number of seconds given by waitSecondsForNewSpawn. There is also a reference to the UIText for score and also a static variable to be accessed from GameOver scene.
- `SpawnAsteroids()` function represents the main loop. It is a coroutine so it returns an IEnumerator object, using 'yield return' so the next spawning waits `waitSecondsForNewSpawn` seconds.
- When the player collides with an asteroid the loop will be broken because the `GameOver` scene is loaded.

Asteroids

- Asteroids have also a logical layer named `Asteroid` and a visual layer named `AsteroidVisual`.
- Asteroid is a prefab which is randomly spawned on top of the screen and moves down towards the player.
- `Asteroid` has as components a Rigidbody, a Capsule Collider and two scripts: Asteroid Movement and Destroy By Contact.
    - The first script sets the `velocity` property of the Rigidbody component to the forward property of transform multiplied by forwardSpeed and sets the `angularVelocity` property of the Rigidbody component to a new Vector3 with a random value on Y axis, multiplied by rotationSpeed.
    - The second one destroys the asteroid and the other object which collied with it. If the other object is a bullet than the score is increased and if the other object is the player than the game is over.

Bullet

- Bullet is also a prefab and it also has two layers: the logical one and the visual one.
- The bullet has a Rigidbody, a Capsule Collider and a Bullet Movement script.
- When a bullet is instantiated, the script sets `velocity` for that bullet to move forward.

Boundaries

- `Boundaries` is a cube GameObject which doesn't have Mesh Filter nor Mesh Renderer and surrounds the game area.
- This GameObject has a Box Collider and a Destroy By Boundaries script. When an object leaves these Boundaries, meaning that the object leaves the game area, that object is destroyed so the RAM will not be full with unused objects. Because the player has some restrictions on moving outside the game area, he will never be destroyed by the boundaries. However, bullets and asteroids are flying around and they are spawned every second so when they are leaving the game area they are being destroyed by the boundaries.