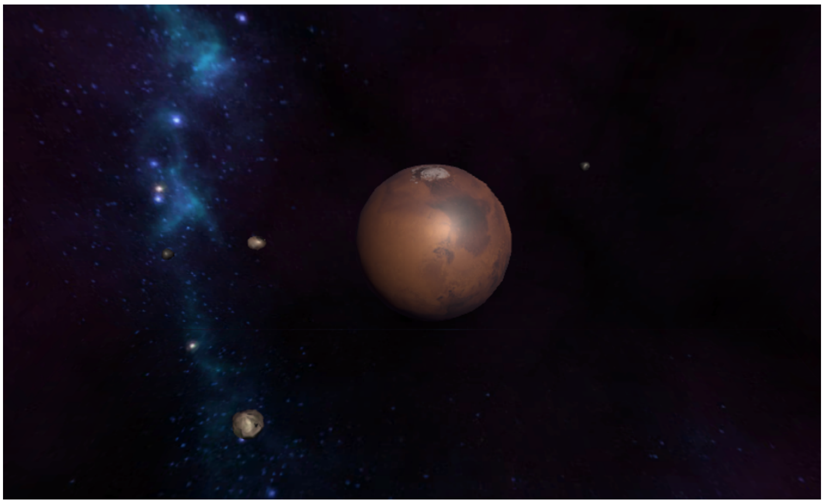
CT3535 Unity3D Lab 3
Asteroids Approaching Mars. Oh no!



Step 1 – Add Assets
Import the 'asteroid1' model and texture (download link is on Canvas)

Step 2 – Colliders & Rigidbody
Make a prefab containing the asteroid model. Add a sphere collider to the asteroid1 prefab, the mars prefab, and the phobos and deimos prefabs, so that we will receive notifications when they collide. Also add a Rigidbody to the asteroid1 prefab, so we can use the physics engine to move it. And turn off gravity for it!

Step 3 – make a new script
In order to handle collisions between asteroids with the planet and moons, we will need a script to attach to asteroids. Call it something like 'AsteroidScript'. Add it in your game's Assets and then attach it to your asteroid prefab.

Step 4 – spawn asteroids randomly
In the GameManager script's Update() handler, use **Random.Range()** to decide whether to spawn a new asteroid (don't make it too likely each time, since Update() is called 60+ times per second!) When you want to spawn an asteroid, use **GameObject.Instantiate** to do so (see lecture notes, section 2).

Step 5 – initialise the asteroid
In the Start() method in AsteroidScript, choose an appropriate starting position to initialise its transform.position with.. perhaps somewhere to the left of mars?
Set the asteroid moving (using physics) using **GetComponent<RigidBody>().AddForce()**

Step 6 – collisions
Implement the OnCollisionEnter() method in AsteroidScript, and in this simply destroy the asteroid using **GameObject.Destroy(this.gameObject)**

Step 7 – removing offscreen asteroids
Also destroy asteroids if they have passed offscreen (which will happen if they didn't collide with anything). You can decide a good way to do this.. sample solution next week.