Name: Sofia Enriquez

HW 2 Written Questions – Game Development

3. a) Explain Time.deltaTime: Time.deltaTime informs the developer about how long each frame took to execute. To make the game “frame rate independent”, the developer can multiply something by Time.deltaTime. The game will act the same on slow and fast computers so the game will be executed at the same speed regardless of the fps. This allows for uniform experience across the developer community.

b) Based on your understanding of Time.deltaTime, please try to explain where it may have been used in the game (Forza Horizon). Also, explain how the user experience would vary depending on the machine if it weren’t used:

Time.deltaTime could have been used to make sure the car's speed is consistent no matter how often the game updates the frame. This would give players a uniform gaming experience regardless of what machine they are playing on. A faster or slower machine wouldn’t affect the car’s speed.

The user experience would vary depending on the machine if Time.deltaTime wasn’t used. If the game is run on a slower machine with a lower frame rate, then the car would move a lot slower. The position wouldn’t update as quickly so the car would move less in a given amount of time compared to a car in a game that’s running on a faster machine. The speed of the car would vary depending on the machine.

5. Explain the following:

a) Mesh Renderer: The mesh renderer component renders 3D assets such as characters and environment objects. When the mesh renderer is on, it renders a 3D mesh in the scene that is visible to the camera. When it’s off, it masks the view of the 3D mesh to the camera.

b) Box collider: The box collider component defines a box shaped collision area for characters and GameObjects in the scene. It is responsible for collision detection and physical interactions of GameObjects.

c) Input.GetAxis method: The Input.GetAxis method returns the value of a virtual axis. This allows it to detect input from devices like keyboards (ex. the arrows), game controllers, and joysticks. The horizontal and vertical axis are used to map input controls to actions in a game. It’s used to move the player around in different directions.

d) Rigid body: The rigid body component causes a GameObject to be controlled by Unity’s Physics engine. This allows for gravity, forces, collision and other realistic physical behaviors.