3.

a. Time.deltaTime is a variable used in Unity that makes the frame rate separate from the speed across all computers, whether fast or slow. It measures the interval between frames and tells the developers this information, and makes it so the game is frame-independent, which means the game will run at the same speed regardless of the frame rate.
b. Forza Horizon likely uses Time.deltaTime when determining the rate of acceleration on the car. A slower computer with a lower frame rate may lag behind when accelerating, so the rate is most likely determined by an independent source like Time.deltaTime. It may also be used when transitioning from gas to brake. Since braking is important to maintain your control and stay on the track, it is important that all users have the ability to brake with urgency if needed, which is where Time.deltaTime would work well. If it wasn't used, as stated above, users with faster computers would have an advantage over users with slower computers, as the fast players would be able to accelerate or brake very fast, while the slow players would experience choppiness or lag and be unable to make precise movements.

5.

a. The **mesh renderer** allows 3D objects to be visible to the camera by rendering them into the scene. It's used to render GameObjects like characters and environments, along with other 3D objects.
b. The **box collider** is used to detect physical interaction between GameObjects. It does this by creating a box-shaped area around an object, and defines the collision area for objects. This box can also be manipulated in size to increase the collision area for the object.
c. The **Input.GetAxis method** returns values of a virtual axis, which come in the form of the names "Horizontal" and "Vertical" that are used to map inputs to actions in-game. It can be used to detect input from external devices, like game controllers and keyboards.
d. **Rigid body** allows an object to act under Unity's physics engine, which enables behaviors like gravity, collision, and force. Gravity determines if the object is affected by the gravity in-game, collision detection determines how collisions are recognized, and mass determines the space an object takes up, which affects how forces act upon it.