CIS 350 – Assignment #2 – **Due Sunday before 11:55pm**

Step 1:

- Download the Prototype 1 assets by direct download or from the Asset Store
- Import the assets into Unity 2018.4.36 (do not use version 2019, 2020, etc.)
- Follow along with the Prototype 1 Follow Along Video to complete Prototype 1
- Note: There are additional required steps in the Prototype 1 video and lecture notes (slides) that are not on the Create with Code website.
- Be sure to include Prototype 1 with your submission of Challenge 1 (separate unity project folders is fine)

Step 2:

- Download the Challenge 1 assets by direct download or from the Asset Store
- Import the assets into Unity 2018.4.36 (do not use version 2019, 2020, etc.)
- If you want to, watch the video introducing Challenge 1 here, but **there will be additional requirements listed below**

Challenge 1 Requirements Part 1 (Requirements from Create with Code)

- The plane is going backward - Make the plane go forward
- The plane is going too fast - Slow the plane down to a manageable speed
- The plane is tilting automatically - Make the plane tilt only if the user presses the up/down arrows
- The camera is in front of the plane - Reposition it so it's beside the plane
- The camera is not following the plane - Make the camera follow the plane
- The plane's propeller does not spin - Create a script that spins the plane's propeller

Hints

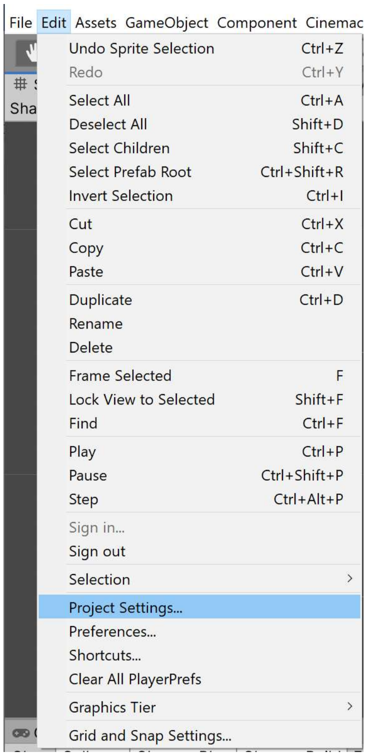
- **Make the plane go forward** *Hint: Vector3.back makes an object move backwards, Vector3.forward makes it go forwards*
- **Slow the plane down to a manageable speed** *Hint: If you multiply a value by Time.deltaTime, it will change it from 1x/frame to 1x/second*
- **Make the plane tilt only if the user presses the up/down arrows** *Hint: In PlaneController.cs, in Update(), the verticalInput value is assigned, but it's never actually used in the Rotate() call*

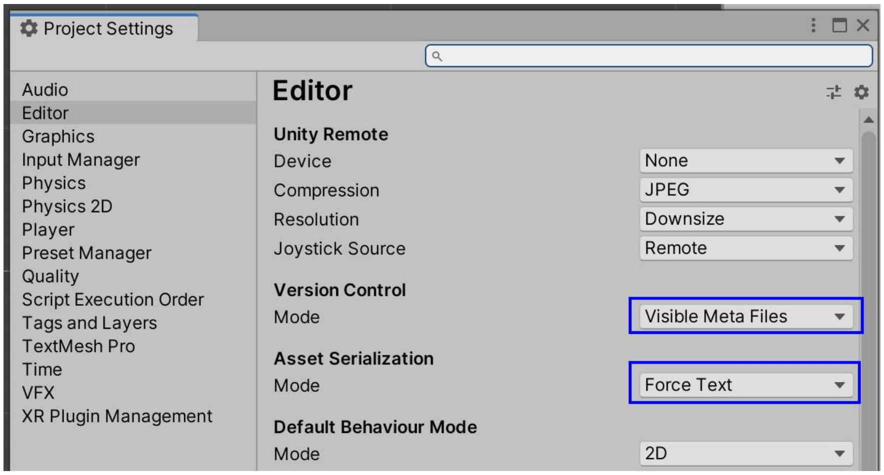
- **Reposition it so it’s beside the plane** *Hint: For the camera’s position, try X=30, Y=0, Z=10 and for the camera’s rotation, try X=0, Y=-90, Z=0*
- **Make the camera follow the plane** *Hint: In FollowPlane.cs, neither the plane nor offset variables are assigned a value - assign the plane variable in the camera’s inspector and assign the offset = new Vector3(30, 0, 10) in the code*
- **Bonus - Make the camera follow the plane** *Hint: There is a “Propeller” child object of the plane - you should create a new “SpinPropellerX.cs” script and make it rotate every frame around the Z axis.*

Challenge 1 Requirements Part 2 (Additional Requirements)

- Display a score (integer variable) at the upper left of the screen
- Add invisible trigger zones between the obstacles that increment score
- Add win condition: score >= 5
- Add loss condition: out of bounds (Hint: Test if player.transform.position.y is above 80 or below -51 and if so, display game over text.)
- Add ability to restart after game over (Hint: if the player presses a button such as R, and the game is over, reload the scene.)

Uploading to GitHub
If you are having difficulty uploading your unity projects to GitHub, see this video and this blog post





Editor

Unity Remote

Device	None
Compression	JPEG
Resolution	Downsize
Joystick Source	Remote

Version Control

Mode	Visible Meta Files

Asset Serialization

Mode	Force Text

Default Behaviour Mode

Mode	2D

Required Deliverable:

- Upload the following to Canvas under Assignments: a .txt text file with a URL web address linking to a git repository containing your project folders and files for both the completed prototype and challenge. You can include two github repository links or put both unity project folders into one repository.

- You must put the .gitignore file under Resources on Canvas in your git repository before adding your project files or creating a unity project in your git repository.

- To receive credit for this assignment, all script files that you write or change **must** include the header comment below with your name on it:

```
/*
 * (Student Name)
 * (Assignment)
 * (Brief description of the code in the file.
 *  For example: controls player movement)
 */
```