**Welcome to Day #2 of CGCC!**

Every day we will have a GitHub repository page that outlines each day and the activities that we will complete. We will also provide all homework on these pages.

Feel free to browse the other days to see what is coming up!

As always, let us know if you need any help or have any questions.

*Link to Camp GitHub*: <https://github.com/paigerodeghero/ClemsonGameCodingCamp/tree/master/2021>

**Links for Camp Day #2**:

* GoDot Engine
  + <https://godotengine.org/>
* GitHub Overview Video
  + <https://youtu.be/w3jLJU7DT5E>
* GitHub Classroom
  + TODO
* How to use GitHub?
  + <https://docs.github.com/en/github/authenticating-to-github/creating-a-personal-access-token>
  + <https://github.com/git-guides/>
  + <https://youtu.be/SzrETQdGzBM>
* Game engines
  + <https://youtu.be/DKrdLKetBZE>
  + Game1: <https://youtu.be/Kf4lVuTYdeM>
  + Game2: <https://youtu.be/E9SnpPXg8hw>
* ProfessorPlatypus
  + <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.1>
  + <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.2>
  + <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.3>
  + <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.4>
  + <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.5>
  + <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.6>
  + <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.7>
  + <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.8>

**Optional Resources**:

* What is open source software (OSS)?
  + <https://www.youtube.com/watch?v=a8fHgx9mE5U>
* Tutorials:
  + FlappyBird
    - <https://youtu.be/8_ThGJG9Kqg>

**Day 2: GitHub, Godot, and creating first game**

**SCHEDULE**:

* Homework review
* Game Elements Continued
* Instructors play “Minesweeper” and narrate the story
* GitHub introduction
* “ProfessorPlatypus” story
* Godot game engine
* Making “ProfessorPlatypus”
* Homework: Make one change to “ProfessorPlatypus” and commit to GitHub

**ACTIVITY**: Homework Review (20 minutes) [James/Paige]

Homework Review:

* Remind students where their journals are.
* Four student present their findings from the game they choose to play (each student gets 2-4 minutes) (Sharing screen is optional. Sharing gameplay is optional).
* Discuss the following game elements from the game.
  + Goal
  + Story
  + Rules
  + Players
  + Player interactions

**INSTRUCTION**: Game Elements Continued… (approximately 20 minutes) [Paige]

* Players
* Intention/Motivation
* Relationship to other players
* Player/enemy
* Characters (“Super Mario Bros.” example)
  + Mario: Plumber
    - *Link*: <https://cdn.vox-cdn.com/thumbor/Yt1avchDkHqEqJuhYZ3YjKF3kFc=/0x0:1700x960/1200x675/filters:focal(714x344:986x616)/cdn.vox-cdn.com/uploads/chorus_image/image/57514059/mario.0.jpg>
  + Luigi: Mario’s brother, his sidekick, also a plumber
    - *Link*: <https://upload.wikimedia.org/wikipedia/en/7/73/Luigi_NSMBUDX.png>
  + Goomba (Mushrooms): They walk around and kill Mario if he touches them. He can jump on them to kill them.
    - *Link*: <https://upload.wikimedia.org/wikipedia/en/c/ce/Goomba.PNG>
  + Koopa Troopas: Birds turned into turtles. They want to kill Mario.
    - *Link*: <https://upload.wikimedia.org/wikipedia/en/b/b2/Koopa_Troopa_NSMBU.png>
  + Princess Toadstool: Ruler of the mushroom kingdom, the goal is to rescue her
    - *Link*: <https://upload.wikimedia.org/wikipedia/en/thumb/d/d5/Peach_%28Super_Mario_3D_World%29.png/220px-Peach_%28Super_Mario_3D_World%29.png>
  + Toad: Protects Princess Toadstool
    - *Link*: <https://upload.wikimedia.org/wikipedia/en/d/d1/Toad_3D_Land.png>
  + Bowser (King Koopa). A dragon who kidnapped Princess Toadstool and is guarding her. Defeat him to win the game.
    - *Link*: https://mario.wiki.gallery/images/7/7d/MSOGT\_Bowser.png
* Player Interaction
  + Player/player
  + Player/computer
* Art /Music
  + Show difference in art.
    - “Super Mario” vs. “Super Mario 64”
    - Show clipart for each character for both games and show them side by side.
      * “Super Mario” (*Link*: <https://www.mariowiki.com/Gallery:Super_Mario_Bros>.)
      * “Super Mario 64” (*Link*: <https://www.mariowiki.com/Gallery:Super_Mario_64>)
* Story
* Essential Conflict

**ACTIVITY:** Think-Share (approximately 13 minutes)

Solo (3 minutes):

* Pick a character from a game of choice
* Find an image of the character ready to share with the camp.
* Remind students where their journals are. Write down the URL in your journal.
* Write down one distinctive part of the character’s graphical costume.

Share (10 minutes:

* Campers will share their character’s image via sharing the URL/screenshare one at a time. We will go in alphabetical order by first name.

**INSTRUCTION**: Instructors play “Minesweeper” and narrate the story (approximately 5 minutes) [Paige]

* “Minesweeper” (*Link*: <https://minesweeperonline.com/>)
* Show off elements of the game that support the story. (E.g., colors of numbers selected)

**ACTIVITY:** Team-Share (approximately 20 minutes)

Pick an existing game, play it and narrate the story.

* Campers join a breakout room with one partner
* Campers pick a game and narrate it to their partner
* Show off elements of the game that support the story
  + Remind students where their journals are.
  + Both partners should write down the game and the elements in their own journals.

Campers come back to main room and volunteers can share their game and narration. If no one volunteers, we go in reverse alphabetical order.

10 MINUTE BREAK

(James will display a 10-minute countdown timer and present his screen).

**INSTRUCTION**: GitHub Introduction (30 min) (Put it in context with game development)

* GitHub intro
  + What is GitHub? (3.5 minutes)
    - <https://youtu.be/w3jLJU7DT5E>
  + Create a token (4 minutes)
    - <https://docs.github.com/en/github/authenticating-to-github/creating-a-personal-access-token>
    - <https://youtu.be/SzrETQdGzBM>
  + How to use GitHub?
    - <https://github.com/git-guides/>
    - Commit
    - Pull
    - Push
    - Merge

**INSTRUCTION**: Creating a game: Story (10 min)

* Demo ProfessorPlatypus game (James)
* Discuss the story of ProfessorPlatypus
* What is the goal?
  + To go through as many obstacles as possible
* How can we make the game more interesting?
  + Add enemies in the game

**ACTIVITY**:How to make ProfessorPlatypus more interesting? (5 min)

* Students share their ideas to make ProfessorPlatypus more interesting

**INSTRUCTION**: Intro Godot: (game engines in general) (5 min)

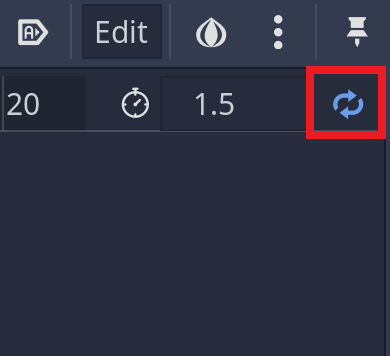
* Launch GoDot.
* What is a game engine?
  + A game engine is a framework that allows us to create games without worrying about how our code interacts with the computer
  + Game engine: <https://youtu.be/DKrdLKetBZE>
* What is Godot?
  + Is an open-source game engine
* Games made in Godot
  + Go over a few and/or show
    - Game1: <https://youtu.be/Kf4lVuTYdeM>
    - Game2: <https://youtu.be/E9SnpPXg8hw>

**INSTRUCTION**: Setup Visual Studio Code (20 min)

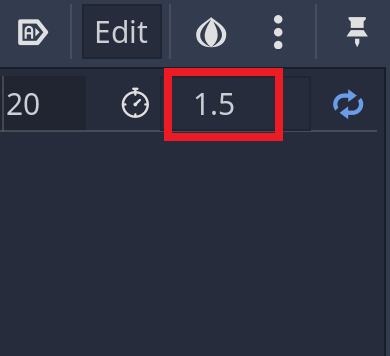
* Open Visual Studio Code
  + Open extensions
  + Install the following
    - Godot-tools
    - Live Share Extension Pack

**ACTIVITY**: Create “ProfessorPlatypus” in godot [see one – do one – show one]

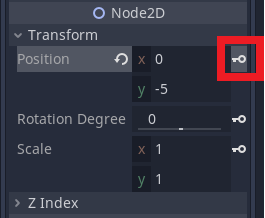
* **INSTRUCTION**: Download project starter from GitHub (10 min) (not git pull)
  + Video instructions: <https://youtu.be/bnFf_b74CQ8>
  + Use this link to download the starter version of ProfessorPlatypus
    - <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.1>
  + Unzip the file
* **INSTRUCTION**: Open the project in visual studio code, press F1 and select the below option (5 min)
  + - Godot Tools: Open workspace with Godot editor
    - If requested, point it towards your Godot installation
* **INSTRUCTION**: Create player object (10 min)
  + Video instructions: <https://youtu.be/YVvJWxasCeU>
  + Open world scene by double clicking **World.tscn** in the **FileSystem**
  + Create a new scene by using the **+** next to **World**
  + Press **+** to add a **RigidBody2D** node. Rename the node to **player**
  + Press **+** to add a **Sprite**
  + Select **player** and press **+** to add a **CollisionShape2D**
  + Select the **Sprite**. Drag the **platypus.png** file into **Texture** in the **Inspector**
  + In **Inspector** -> **Animation** -> **Hframes**, set the value to 3
  + Select **CollisionShape2D**. In **Inspector** -> **Shape**. Select **New RectangleShape 2D**
  + Increase the size of the collider as required
  + Save scene as player.tscn
  + GitHub: <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.2>
* **INSTRUCTION**: Create player animation (10 min)
  + Video instructions: <https://youtu.be/6ZvLHejkwFQ>
  + Select **player** and press **+** to add an **AnimationPlayer**
  + Create idle animation
    - Use **Animation** -> **New** to create a new animation
    - Name it as **idle** and press **OK**
    - Enable looping using the button



* + - Set animation duration to 1.5 seconds



* + - Select **Sprite** from **Scene** tab and navigate to **Animation** in **Inspector**
    - Select time **0** on animation
    - Set Hframes to 0
    - Create a keyframe by pressing the key button

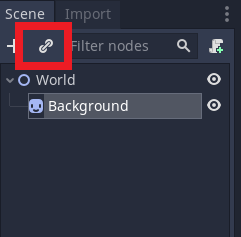


* + - Select time **0.2** on animation
    - Set Hframes to 1
    - Create another keyframe
    - Select time **0.4** on animation
    - Set Hframes to 2
    - Create another keyframe
  + Save the game
  + GitHub: <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.3>
* **INSTRUCTION** Create a second animation (10 min)
  + Video instructions: <https://youtu.be/BmVsRuwtex0>
  + Creating swim animation
    - Use **Animation** -> **New** to create a new animation
    - Name it as **swim** and press **OK**
    - Set duration of animation to **0.6**

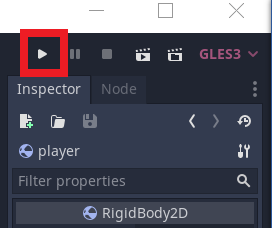
Graphical user interface, application

Description automatically generated

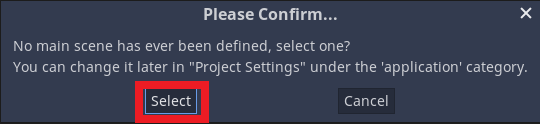
* + - Select **Sprite** from **Scene** tab and navigate to **Animation** in **Inspector**
    - Select time **0** on animation
    - Set Hframes to 0
    - Create a keyframe by pressing the key button
    - Select time **0.2** on animation
    - Set Hframes to 1
    - Create a keyframe by pressing the key button
    - Select time **0.4** on animation
    - Set Hframes to 2
    - Create a keyframe by pressing the key button
    - Enable looping using the button
  + Save the game
  + GitHub: <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.4>
* **INSTRUCTION**: Add player instance to World scene (5 min)
  + Video instructions: <https://youtu.be/gnaTLSOjAck>
  + Open the World scene
  + Create an instance of the Player



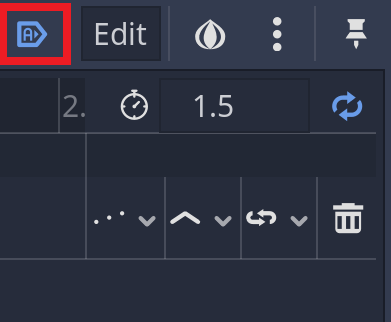
* + Place player in the middle of the scene
    - You can use the mouse to click and drag the player or you could use the transform property
  + Use the play button on top right to play the game



* + Click **Select** from the below shown popup. This will help us define the main scene



* + Select **World.tscn** and click **open**
  + Open **player.tscn**
  + Select **AnimationPlayer**, select **idle** animation
  + Enable autoplay on load by pressing the button



* + Select **player**
  + Set **Inspector** -> **RigidBody2D** -> **Gravity Scale** to **0**
  + Set Gravity Scale to 0
  + Save the game
  + GitHub: <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.5>

10 MINUTE BREAK

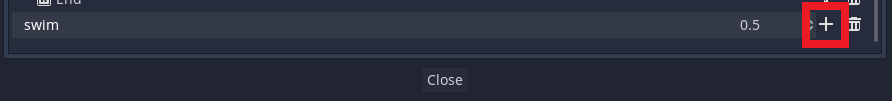
(James will display a 10-minute countdown timer and present his screen).

* **INSTRUCTION**: Make player move with keyboard input (10 min) (VS code liveshare)
  + Video instructions: <https://youtu.be/OfhooWUMxU4>
  + Open **Player** scene
  + Select **Player**
  + Define an action
    - Navigate to **Project** -> **Project Settings** -> **Input Map**
    - Add an action **swim**

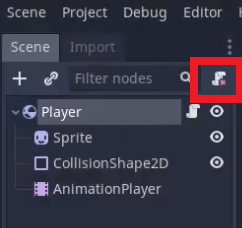
Graphical user interface, application

Description automatically generated

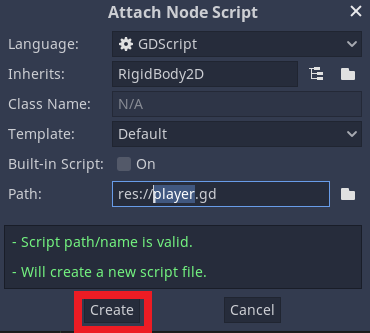
* + - Use the **+** next to swim to link action with space bar press
      * Key -> Press **Space** -> OK



* + Click close
  + Add a new empty script to the player by using the script button



* + Click Create



* + Save the game
  + GitHub: <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.6>
* **INSTRUCTION**: Add code to move player up and down (15 min)
  + Video instructions part 1: <https://youtu.be/ATUObD9IEsU>
  + Video instructions part 2: <https://youtu.be/oafvLTNNA2Q>
  + Open player.gd in Microsoft Visual Studio Code
  + Replace the contents with below code
    - This will move the player up and down as well as play the swim animation

extends RigidBody2D

var started = false

func \_physics\_process(delta):

    if Input.is\_action\_just\_pressed("swim"):

        if !started:

            start\_swim()

        swim\_speed()

func start\_swim():

    started = true

    gravity\_scale = 5.0

    $AnimationPlayer.play("swim")

func swim\_speed():

    linear\_velocity.y = -200

* + Save the game
  + GitHub: <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.07>
* **INSTRUCTION**: Add limits to where the player can go (10 min)
  + Video instructions: <https://youtu.be/tkjs7on87zk>
  + Modify Player.gd script to sop the player from falling beyond a limit

extends RigidBody2D

var started = false

func \_physics\_process(delta):

    if Input.is\_action\_just\_pressed("swim"):

        if !started:

            start\_swim()

        swim\_speed()

    if position.y >= 500:

        gravity\_scale = 0

        linear\_velocity.y = 0

        $AnimationPlayer.stop()

func start\_swim():

    started = true

    gravity\_scale = 5.0

    $AnimationPlayer.play("swim")

func swim\_speed():

    linear\_velocity.y = -200

* + Save the game
  + GitHub: <https://github.com/domini4/ProfessorPlatypus/releases/tag/1.8>

INSTRUCTION: Show the students how to commit the game to Github. (5 min)

**HOMEWORK**:

* Be sure you have completed all steps shown today
* Make one change to “ProfessorPlatypus” to make it more interesting
* Commit homework to GitHub
  + Create a token
    - <https://docs.github.com/en/github/authenticating-to-github/creating-a-personal-access-token>
    - <https://youtu.be/SzrETQdGzBM>