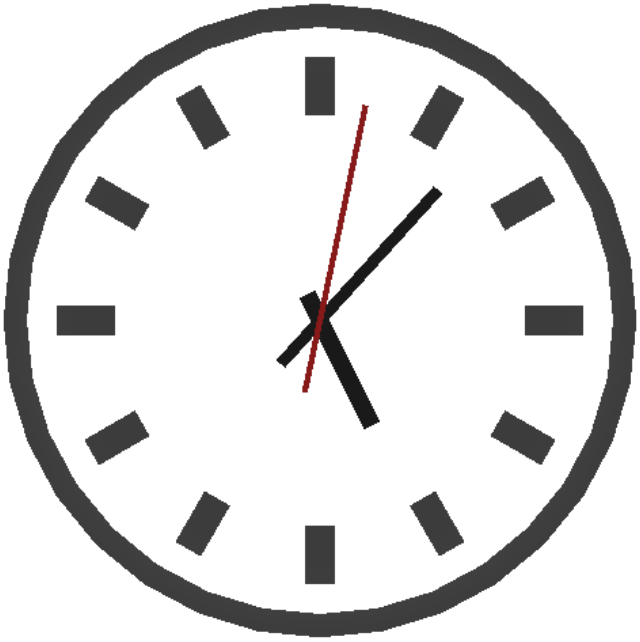
Clock Project



Project Engines:

- Unity
- Unreal
- Godot
- Custom

Credits:
https://catlikecoding.com/unity/tutorials/basics/game-objects-and-scripts/
https://catlikecoding.com/godot/introduction/2-programming-a-clock/

Game Objects and Basic Motion

- Build a clock face using primitive objects in your given engine
- Rotate the clock arms to show the current time
- Write code to animate the arms in real time

Create a Project

- Create an empty project
- Locate the scene view
- Locate the scene tree
- Locate the code editor

Create a Scene

- Create an empty scene
- Locate the primitive objects
- Locate the scene camera
  - Position the camera 10 units in the up direction
  - Change the camera type to orthographic and the size to 15 (optional)
  - Orient the camera facing down
- Locate the scene light
  - Position the light 10 units in the up directional
  - Change the light type to directional
  - Orient the light facing down
- Create a folder for Meshes
- Create a folder for Materials
- Create a folder for Scripts
- Import the clock ring shape to the Meshes folder

Credits:
https://catlikecoding.com/unity/tutorials/basics/game-objects-and-scripts/
https://catlikecoding.com/godot/introduction/2-programming-a-clock/

Create the Clock

- Add a cylinder primitive for the clock face
  - Position: <0, 0, 0>
  - Rotation: <0, 0, 0>
  - Scale: <10, 0.2, 10>
- Add a box primitive for the hour mark
  - Position: <0, 0.25, 4>
  - Rotation: <0, 0, 0>
  - Scale: <0.5, 0.1, 1>
- Rename the box Hour12
- Locate the shader material for the hour mark and change the color to dark gray
- Likewise, make Hour3, Hour6, and Hour9 marks
- Create a node named Center and make it the parent to the hour marks
- Copy and rotate the hour mark around the entire clock face
  - Copy the Center object and rotate the copy 30 degrees
  - Copy the Center object again and rotate the copy 60 degrees
- Add a second box primitive for the hour hand
  - Position: <0, 0.25, 0.75>
  - Rotation: <0, 0, 0>
  - Scale: <0.3, 0.1, 2.5>
- Create a node named HourPivot and make it the parent to the hour hand
- Locate the shader material for the hour hand and change the color to black
- Add another, longer and skinnier, box primitive for the minute hand
  - Position: <0, 0.35, 1>
  - Rotation: <0, 0, 0>
  - Scale: <0.2, 0.1, 4>
- Create a node named MinutePivot and make it the parent to the minute hand
- Locate the shader material for the minute hand and change the color to black
- Add yet another, even longer and skinnier, box primitive for the second hand
  - Position: <0, 0.45, 1.25>
  - Rotation: <0, 0, 0>
  - Scale: <0.1, 0.1, 5>
- Create a node named SecondPivot and make it the parent to the second hand
- Locate the shader material for the second hand and change the color to dark red
- Add the clock ring shape

Credits:
https://catlikecoding.com/unity/tutorials/basics/game-objects-and-scripts/
https://catlikecoding.com/godot/introduction/2-programming-a-clock/

Rotate the Clock Arms

- Create a variable for the current hour
- Link the variable to the rotation axis of the hour arm
- Set the variable to the current hour
- Create a variable for the current minute
- Link the variable to the rotation axis of the minute arm
- Set the variable to the current minute
- Create a variable for the current second
- Link the variable to the rotation axis of the second arm
- Set the variable to the current second

NOTE: Include factional units for minutes and hours for continuous hand movement

Animate the Arms

- Locate the frame's update method
- Set the hour variable to the current time in the update method
- Set the minute variable to the current time in the update method
- Set the second variable to the current time in the update method

More

- Try the project in another game engine
- Are the results the same? Why/Why not?

Credits:
https://catlikecoding.com/unity/tutorials/basics/game-objects-and-scripts/
https://catlikecoding.com/godot/introduction/2-programming-a-clock/