Import the simpleGE, pygame, and random modules

Define a class called Coin that inherits from the Sprite class in simpleGE

Define a constructor method for the Coin class

Call the constructor method of the superclass (Sprite) within the Coin class constructor

Color the rectangle representing the coin with yellow color and size (25, 25)

Set the size of the coin sprite to (50, 50)

Set the initial x-coordinate of the coin to a random value between 0 and the screen width

Set the initial y-coordinate of the coin to a random value between 0 and the screen height

Set the minimum speed of the coin to 3 and the maximum speed to 8

Call the reset method to initialize the position and speed of the coin

Define a method named reset for the Coin class

Set the y-coordinate of the coin to 10

Set the x-coordinate of the coin to a random value between 0 and the screen width

Set the vertical speed (dy) of the coin to a random value between the minimum and maximum speed

Define a method named checkBounds for the Coin class

Check if the bottom edge of the coin sprite exceeds the screen height

If the bottom edge exceeds the screen height, call the reset method to reposition the coin

Define a class called Mickey that inherits from the Sprite class in simpleGE

Define a constructor method for the Mickey class

Call the constructor method of the superclass (Sprite) within the Mickey class constructor

Set the image of the Mickey sprite to "Mickey\_Mouse.jpg"

Set the size of the Mickey sprite to (50, 50)

Define a method named process for the Mickey class

Check if the 'a' key is pressed, if so, move the Mickey sprite 10 units to the left

Check if the 'd' key is pressed, if so, move the Mickey sprite 10 units to the right

Check if the 'w' key is pressed, if so, move the Mickey sprite 10 units upward

Check if the 's' key is pressed, if so, move the Mickey sprite 10 units downward

Define a class called LblScore that inherits from the Label class in simpleGE

Define a constructor method for the LblScore class

Call the constructor method of the superclass (Label) within the LblScore class constructor

Set the text of the label to "Score: 0"

Set the center coordinates of the label to (100, 30)

Define a class called LblTime that inherits from the Label class in simpleGE

Define a constructor method for the LblTime class

Call the constructor method of the superclass (Label) within the LblTime class constructor

Set the text of the label to "Time Left: 10"

Set the center coordinates of the label to (500, 30)

Define a class called Game that inherits from the Scene class in simpleGE

Define a constructor method for the Game class

Call the constructor method of the superclass (Scene) within the Game class constructor

Initialize pygame

Create a new scene object

Set the caption of the scene to "Game"

Set the score to 0

Create a new timer object

Set the total time of the timer to 10 seconds

Set the background image of the scene to "sky.jpeg"

Load the sound file "coin.wav" and assign it to the variable sndCoin

Create a new Mickey sprite object within the scene

Create a new Coin sprite object within the scene

Create a list to hold multiple Coin sprite objects

Iterate 10 times and append a new Coin sprite object to the coins list

Create a new label to display the score within the scene

Create a new label to display the remaining time within the scene

Define a method named process for the Game class

Iterate over each coin sprite in the coins list

Check if the Mickey sprite collides with any of the coins

If collision occurs, play the coin sound, reset the coin's position, increment the score, and update the score label

Update the text of the time label with the remaining time from the timer

Check if the remaining time is less than 0

If time is up, print the final score and stop the scene

Define a class called Instructions that inherits from the Scene class in simpleGE

Define a constructor method for the Instructions class

Create a new MultiLabel object to display multiple lines of text

Set the text lines for the instructions

Set the center coordinates and size of the instructions

Create a new label to display the last score

Set the text of the score label to include the previous score

Set the center coordinates of the score label

Create a new button to play the game

Set the text of the play button

Set the center coordinates of the play button

Create a new button to quit the game

Set the text of the quit button

Set the center coordinates of the quit button

Define a method named process for the Instructions class

Check if the quit button is clicked

If clicked, set the response to "Quit" and stop the scene

Check if the play button is clicked

If clicked, set the response to "Play" and stop the scene

Check if the up arrow key is pressed

If pressed, set the response to "Play" and stop the scene

Check if the down arrow key is pressed

If pressed, set the response to "Quit" and stop the scene

Define a main function

Initialize a variable named keepGoing to True

Initialize a variable named score to 0

Create a while loop that continues while keepGoing is True

Create a new Instructions scene with the previous score

Start the Instructions scene

If the response from the Instructions scene is "Play", create and start a new Game scene

Update the score variable with the score from the Game scene

If the response from the Instructions scene is "Quit", set keepGoing to False to exit the loop

Define the entry point of the script

Call the main function when the script is executed