

```
1 #Vaibhav Choudhary gau7jk
2
3 import uvage
4 import random
5
6 camera = uvage.Camera(800, 600)
7 game_true = False
8 game_false = True
9 score = 0
10 y_coordinate = 0
11
12 # where the camera begins
13 camera.x = 400
14 camera.y = 350
15
16 # components of the game
17 player = uvage.from_color(600, 500, "cyan", 15, 15)
18 floor = []
19
20 def tick():
21     global score
22     global game_true
23     global game_false
24     global floor
25     global player
26     global y_coordinate
27     global stop
28
29     stop = uvage.from_color(400, camera.y - 300, "
cyan", 800, 40)
30
31     # The player can start the game by clicking on
the mouse
32     if camera.mouseclick:
33         game_true = True
34         game_false = False
35
36     def visual():
37         """
38         This function creates the background for
the game.
```

```

39         :return: This function does not return
        anything.
40         """
41         global stop
42         global floor
43
44         camera.clear("black")
45         camera.draw(stop)
46
47         if game_true == True:
48             for each in floor:
49                 camera.draw(each)
50
51         player.draw(camera)
52
53     def instructions():
54         """
55         This function explains what happens once
        the player begins the game by clicking the down
        arrow key.
56         The function shows how each of the floors
        are created using a random number generator.
57         :return: none
58         """
59         global game_true
60         global y_coordinate
61         global floor
62         global score
63         if game_true == True:
64             if game_false == False:
65                 player.move_speed()
66                 camera.move(0, 1)
67
68         if game_true == True:
69             if game_false == False and score % 60
70             == 0:
71                 left_floor = random.randint(0, 299)
72                 left_width = 2 * left_floor
73                 right_floor = (800 + left_width +
74                 75) * (1/2)
75                 right_width = 800 - (left_width +

```

```

73 75)
74
75         floor.append(uvage.from_color(
    left_floor, camera.y + 300, "red", left_width, 15
    ))
76         floor.append(uvage.from_color(
    right_floor, camera.y + 300, "red", right_width,
    15))
77         y_coordinate = y_coordinate + 100
78         if game_false == True:
79             y_coordinate = y_coordinate + 0
80
81     def x_move():
82         """
83         This function explains what each key does
    to move the player along the x-plane.
84         :return: This function does not return
    anything.
85         """
86         if uvage.is_pressing("right arrow"):
87             if player.x != 800:
88                 player.x += 10
89         if uvage.is_pressing("left arrow"):
90             if player.x != 0:
91                 player.x -= 10
92
93     def y_move():
94         """
95         This function explains how the camera
    moves with relation to the player's position.
96         :return: This function does not return
    anything.
97         """
98         if player.y > camera.y + 250:
99             player.y = camera.y + 250
100        elif player.y == camera.y + 250:
101            player.y = camera.y + 250
102        elif player.y < camera.y + 250:
103            player.y = player.y
104
105

```

```

106
107     def touch():
108         """
109         This function explains what happens when
110         the player interacts with the floors.
111         :return: This function does not return
112         anything.
113         """
114         global floor
115         for each in floor:
116             if player.touches(each):
117                 player.yspeed = 0
118             else:
119                 player.yspeed = player.yspeed + 0.25
120         for wall in floor:
121             if player.touches(wall):
122                 player.yspeed = 0
123
124     visual()
125
126     def displays():
127         """
128         This function explains what is displayed
129         on the screen during the game.
130         :return: This function does not return
131         anything.
132         """
133         global game_true
134         global game_false
135         global score
136         global stop
137
138         if game_true == True:
139             if game_false == False:
140                 camera.draw(uvage.from_text(700,
141                 500 + score, str(score), 40, "magenta", bold =
142                 True))
143
144         if game_true == False:

```

```
141         if game_false == True:
142             camera.draw(uvage.from_text(400,
143             250, "Press the mouse to start!", 60, "white",
144             bold = True))
145
146         if player.top_touches(stop):
147             camera.move(0,0)
148             camera.draw(uvage.from_text(camera.x,
149             camera.y, "Game Over!", 50, "red"))
150             game_true = False
151             game_false = True
152
153     displays()
154     instructions()
155     x_move()
156     y_move()
157     touch()
158
159     camera.display()
160     score += 1
161
162
163
164 ticks_per_second = 30
165 uvage.timer_loop(ticks_per_second, tick)
```