

B1- BAIT – CIT110 Assignment 1

The following Python script provides a foundational model of a wall clock. To create a more accurate and visually appealing simulation, several enhancements are necessary. You are required to make the following improvements.

- Accurately represent the current time with the clock hands.
- Revamp the clock face to make it more attractive.
- Add at least one (or more) additional features

Submit your Python code as a .py file so that can be run. Example: 22UG3-dddd.py
Unreadable versions are not encountered.

Saved as b1_baitass1.py

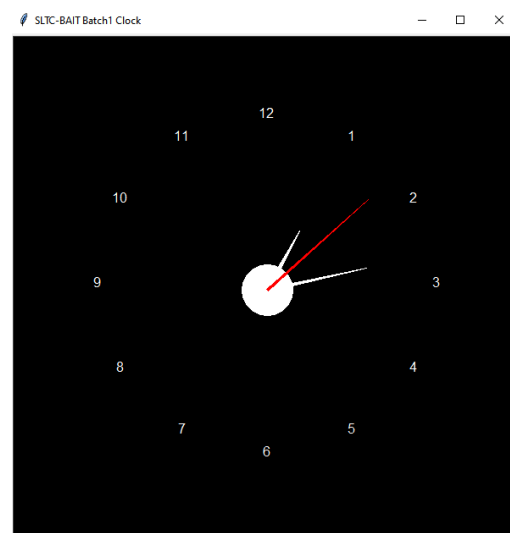
```
import turtle
import datetime
import math

# Set up the turtle screen
screen = turtle.Screen()
screen.bgcolor("black")
screen.setup(width=600, height=600)
screen.title("SLTC-BAIT Batch1 Clock")
screen.tracer(0)

# Very basic clock face
face = turtle.Turtle()
face.shape("circle")
face.color("white")
face.fillcolor("white")
face.shapesize(stretch_wid=3, stretch_len=3)
face.penup()

# Hour hand
hour_hand = turtle.Turtle()
hour_hand.shape("arrow")
hour_hand.color("white")
hour_hand.shapesize(stretch_wid=0.3, stretch_len=8)
hour_hand.penup()

# Minute hand
minute_hand = turtle.Turtle()
minute_hand.shape("arrow")
minute_hand.color("white")
minute_hand.shapesize(stretch_wid=0.2, stretch_len=12)
minute_hand.penup()
```



```

# Second hand
second_hand = turtle.Turtle()
second_hand.shape("arrow")
second_hand.color("red")
second_hand.shapesize(stretch_wid=0.1, stretch_len=16)
second_hand.penup()

# Hour labels
hour_labels = turtle.Turtle()
hour_labels.color("white")
hour_labels.penup()
hour_labels.hideturtle()

# Draw the hour labels
def draw_hour_labels():
    for i in range(1, 13):
        angle = math.radians(30 * i)
        x = 200 * math.sin(angle)
        y = 200 * math.cos(angle)
        hour_labels.goto(x, y)
        hour_labels.write(str(i), align="center", font=("Arial", 12, "normal"))

# Update the clock time
def update_clock():
    now = datetime.datetime.now()
    # Angles for each hand
    hour_angle = (now.hour % 12) * 30 + now.minute / 2
    # minute_angle = now.minute * 6
    minute_angle = now.minute * 6 + now.second * 0.1 # Consider the seconds here
    second_angle = now.second * 6

    # Rotate the hands
    hour_hand.setheading(-hour_angle)
    minute_hand.setheading(-minute_angle)
    second_hand.setheading(-second_angle)

    # Update the screen
    screen.update()

    # Call ontimer() again after 1 second
    screen.ontimer(update_clock, 1000)

# Draw the hour labels
draw_hour_labels()

# Update the clock
update_clock()

# Start main loop
screen.mainloop()

```