Page No. _____
Date

Python Day - 18 (Intermediate)

Topic :- Turtle Graphics, Tuples, Importing Module

*) Why Day 18 Matters (Big picture)

Day 18 is not about drawing pretty pictures. It teaches you

- How external libraries work
- How to think in coordinates
- How to use immutable data (tuples)
- How to build logic + creativity together

This is the first day where code starts to feel alive

(*) Turtle Graphics - (Core idea)

What is Turtle?

- Think of a robot with a pen.
- It starts at the center of the screen
- It faces right (East) by default
- It can:
  - Move forward / backward
  - Turn left / right
  - Draw while moving

Key rule :-

Turtle draws while moving not when standing still

(*) Setting up turtle (Very Important)

Basic Setup

```
from turtle import Turtle, Screen

tim = Turtle()
Screen = Screen()

screen.exitonclick()
```

(*) what each line does

line	Meaning
from turtle import	Imports Specific classes
Turtle()	Creates a turtle object
Screen()	Creates a canvas
exitonclick()	Keeps window Open

without exitonclick(), the window will close instantly

(*) Turtle Movement Commands (Memorize)

Movement

```
tim.forward(100)
tim.backward(50)
```

Turning

```
tim.left(90)
tim.right(45)
```

Turning does NOT move the turtle

Date

(*) Pen Control (very Important)

Pen Up/Down

```
tim.penup()
tim.pendown()
```

- penup() → move without drawing
- pendown() → resume drawing

Pen Size

```
tim.pensize(5)
```

Speed

```
tim.speed(0)
```

Speed	Meaning
1	very slow
10	fast
0	Instant

(*) Drawing Shapes (logic + Math)
Square Example

```
for _ in range(4):
    tim.forward(100)
    tim.right(90)
```

Angle formula
Angle = 360/number of sides

Generic polygon function

```
def drawShape(sides):
    angle = 360/sides
    for _ in range(sides):
        tim.forward(100)
        tim.right(angle)
```

(*) Dashed line (Interview Question)

```
for _ in range(15):
    tim.forward(10)
    tim.penup()
    tim.forward(10)
    tim.pendown()
```

Shows:
- loop control
- pen logic
- visual thinking

(*) Colors in Turtle

Named colors

```
tim.color("red")
```

RGB Colors (ADVANCED)

```
import random
import turtle

turtle.colormode(255)
tim.color((255,100,50))
```

RGB only works if colormode(255) is set

(*) Random walk (Important Concept)

Directions as Angles

directions = [0, 90, 180, 270]

Random walk Code

```
import random
tim.fonsize(10)
tim.Speed(0)

for _ in range(200):
    tim.forward(30)
    tim.Setheading(random.choice(directions))
```

Why Setheading()?
It directly sets direction instead of rotating incrementally.

(*) Tuples (Crucial Theory)

What is a Tuple?

```
my_tuple = (255, 0, 100)
```

- Ordered
- Immutable
- Used for fixed data

Tuple vs List

Feature	list	Tuple
Mutable	✓	✗
Syntax	[ ]	( )
Use Case	Changeable	Fixed

RGB colors are tuples because color shouldn't change accidentally.

(*) Importing Modules (Very important)
3 ways to import

(1) Import everything

```
import turtle
tim = turtle.Turtle()
```

(2) Import Specific classes (Best practice)

```
from turtle import Turtle, Screen
```

(3) Import and rename

```
import turtle as t
```

(*) Spirograph (Final Boss)
Idea
- Draw many circles
- Slightly rotate each time

Code

```
import random
import turtle

turtle.colormode(255)
tim.speed(0)

def random_color():
    r = random.randint(0,255)
    g = random.randint(0,255)
    b = random.randint(0,255)
    return (r,g,b)

for _ in range(36):
    tim.color(random_color())
    tim.circle(100)
    tim.setheading(tim.heading()+10)
```

This Combines:

- loops
- functions
- Tuples
- Angles
- Creativity

*) Mental Model

Turtle is not about art

Turtle is about thinking spatially with code

if you understand:

- Angles
- loops
- functions
- Tuples

{ you're done close laptop }