**PYTHON ASSIGNMENT**

Scenario: You’re developing a digital scoreboard for a sports event, where the scores of two teams are tracked throughout a game. The scoreboard should update each time a team scores and display the current score. Additionally, it should determine and display the winning team when the game ends.

• Tasks:

1. What variables are necessary for storing each team’s score?

2. How could conditional statements update and display scores and determine the winner?

3. Would conditionals alone be sufficient for this program? Discuss the ease of updating without functions.

4. Describe how tuple assignment could help organize team names and scores for display.

• Deliverables:

1. Scoreboard Code: A Python program for tracking and displaying scores in real-time.

2. Learning Outcomes: Students will learn dynamic data updating, conditional checks for determining outcomes, and tuple-based data handling.

Here's a Python solution for the digital scoreboard:

Scoreboard Code

# Initialize team names and scores

team1\_name = "Team A"

team2\_name = "Team B"

team1\_score = 0

team2\_score = 0

# Game duration (minutes)

game\_duration = 60

# Function to update scores

def update\_score(team, points):

global team1\_score, team2\_score

if team == team1\_name:

team1\_score += points

elif team == team2\_name:

team2\_score += points

display\_scores()

# Function to display scores

def display\_scores():

print(f"{team1\_name}: {team1\_score} - {team2\_name}: {team2\_score}")

# Function to determine winner

def determine\_winner():

if team1\_score > team2\_score:

return f"{team1\_name} wins!"

elif team2\_score > team1\_score:

return f"{team2\_name} wins!"

else:

return "It's a tie!"

# Main game loop

for minute in range(game\_duration):

print(f"\nMinute {minute+1}:")

action = input("Enter team name and points (e.g., 'Team A 3') or 'end' to finish: ")

if action.lower() == "end":

break

team, points = action.split()

points = int(points)

update\_score(team, points)

# Display final scores and winner

print("\nFinal Scores:")

display\_scores()

print(determine\_winner())

Learning Outcomes

1. Dynamic data updating: The program updates scores in real-time based on user input.

2. Conditional checks: The program uses conditional statements to determine the winner.

3. Tuple-based data handling: Although not used extensively, tuple assignment can help organize team names and scores for display.

Discussion

- Variables: team1\_name, team2\_name, team1\_score, and team2\_score store team names and scores.

- Conditional statements: Used to update scores, determine the winner, and handle user input.

- Sufficiency of conditionals: Conditionals alone are sufficient for this program, but using functions (e.g., update\_score, display\_scores, determine\_winner) improves code organization and reusability.

- Tuple assignment: Can be used to assign team names and scores simultaneously, e.g., team1\_name, team1\_score = "Team A", 0.

Ease of updating without functions

Without functions, the code would become lengthy and repetitive, making it harder to maintain and update. Functions encapsulate logic, making it easier to modify or extend the program.

Tuple assignment example

Instead of assigning team names and scores separately, you can use tuple assignment:

team1\_info = ("Team A", 0)

team2\_info = ("Team B", 0)

team1\_name, team1\_score = team1\_info

team2\_name, team2\_score = team2\_info

This approach can simplify data handling and display.