Worksheet 25: Python Dictionary Traversal

Name: _____ Date: _____

Instructions

- Answer in the blanks.
- For “write code” questions, write **valid Python code** (**no functions needed**).
- For “what does it print” questions, write the **exact** output (line by line).
- If a question says “order may vary”, any correct order is acceptable.
- You may assume all inputs are valid (unless the question says otherwise).

Part A — Traversing keys

1) Quick facts (fill in the blanks)

Fill in the blanks using: **keys**, **values**, **pairs**.

1. `for k in d:` traverses the dictionary’s _____.
2. `for k, v in d.items():` traverses key-value _____.
3. To get a value, we can use `d[k]` after we have a _____.

2) Print keys (what does it print?)

```
scoreboard = {"Amy": 3, "Ben": 5, "Chloe": 1}

for player in scoreboard:
    print(player)
```

Output:

```

```

3) Keys + look up values (fix the bug)

The code has **one bug**. Fix it so it prints the player and score.

```
scoreboard = {"Amy": 3, "Ben": 5, "Chloe": 1}

for score in scoreboard:
    score = scoreboard[player]
    print(player, score)
```

Which line has bug? Please correct the line.

```

```

4) Values only (write code)

Write code to print **only the scores**, one per line.

```
scoreboard = {"Amy": 3, "Ben": 5, "Chloe": 1}

# Write code here:
```

Expected output (line by line):

```
3
5
1
```

Part B — Traversing key-value pairs with `.items()`

5) `.items()` (fill in the blank)

Fill in the blank to traverse key-value pairs.

```
scoreboard = {"Amy": 3, "Ben": 5, "Chloe": 1}

for player, score in ___________________________:
    print(player, score)
```

Expected output:

```
Amy 3
Ben 5
Chloe 1
```

6) Running total (what does it print?)

```
scoreboard = {"Amy": 3, "Ben": 5, "Chloe": 1}

total = 0
for player, score in scoreboard.items():
    total = total + score
    print(total)

print("done")
```

Output:

```

```

7) Count meets a rule (write code)

Write code to count how many players have score **at least 4**.

```
scoreboard = {"Amy": 3, "Ben": 5, "Chloe": 1, "Drew": 4}

count = 0
for player, score in scoreboard.items():
    # Write code here:




print(count)
```

Expected output:

```
2
```

8) Collect names (write code)

Make a list named `winners` that stores all players with score **>= 3**.

```
scoreboard = {"Amy": 3, "Ben": 5, "Chloe": 1, "Drew": 4}

winners = []

for player, score in scoreboard.items():
    # Write code here:







print(winners)
```

Expected output:

```
['Amy', 'Ben', 'Drew']
```

Part C — Max / Min by traversal

9) Find the winner (fill in the blanks)

Fill in the blanks so the code prints the winner and best score.

```
scoreboard = {"Amy": 3, "Ben": 5, "Chloe": 1, "Drew": 4}

winner = None
best = -1

for player, score in scoreboard.items():
    if score > _____________________:
        best = _____________________
        winner = _____________________

print(winner, best)
```

Expected output:

```
Ben 5
```

10) Find the lowest score (write code)

Write code to find the player with the **lowest** score.

```
scoreboard = {"Amy": 3, "Ben": 5, "Chloe": 1, "Drew": 4}

loser = None
worst = 10**9  # a very large number

for player, score in scoreboard.items():
    # Write code here:









print(loser, worst)
```

Expected output:

```
Chloe 1
```

11) Tie for best (write code)

Sometimes there is a tie. Write code to print a list of all players who have the **max** score.

```
scoreboard = {"Amy": 5, "Ben": 5, "Chloe": 1, "Drew": 4}

# Step 1: find the max score in a variable best
best = -1
for _______, _______  in scoreboard.items():
    if score > best:
        ________________________

# Step 2: collect all players with score == best into a list winners
winners = []
for _______, _______ in scoreboard.items():
    if score == best:
        winners. ________________________

print(best)
print(winners)
```

Expected output:

```
5
['Amy', 'Ben']
```

Part D — Practice tasks (still traversal)

12) Average score (write code)

Compute the average score and print it as a float.

```
scoreboard = {"Amy": 3, "Ben": 5, "Chloe": 1, "Drew": 4}

total = 0
count = 0

for player, score in scoreboard._________________:
    total = ____________________
    count = ____________________

avg = ____________________________
print(avg)
```

Expected output:

```
3.25
```

13) Print like a scoreboard (write code)

Print each line in this format: `player: score`

```
scoreboard = {"Amy": 3, "Ben": 5, "Chloe": 1}

for player, score in scoreboard.items():
    print(_____________________________________)
```

Expected output:

```
Amy: 3
Ben: 5
Chloe: 1
```

14) Even scores (write code)

Print the players whose score is **even**.

```
scoreboard = {"Amy": 3, "Ben": 5, "Chloe": 2, "Drew": 4}

# Write code here:
```

Expected output:

```
Chloe
Drew
```

15) Build a new dictionary (write code)

Create a new dictionary `double_score` where each player's score is doubled.

```
scoreboard = {"Amy": 3, "Ben": 5, "Chloe": 1}

double_score = {}

# Write code here:









print(double_score)
```

Expected output:

```
{'Amy': 6, 'Ben': 10, 'Chloe': 2}
```

16) Total points for names starting with a letter (write code)

Add up the scores for players whose names start with `"A"`.

```
scoreboard = {"Amy": 3, "Alex": 2, "Ben": 5, "Ava": 4}

total_a = 0

for player, score in ____________________________________:
    if player[0] == "A":
        total_a = ____________________________________

print(total_a)
```

Expected output:

```
9
```