Worksheet 24: Python Dictionary Basics

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.

Part A — Dictionary basics (create, empty, keys/values)

1) Create a dictionary (what does it print?)

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

print(scoreboard)
print(type(scoreboard))
```

Output:

```

```

2) Empty containers (fill in the types)

Fill in the type of each variable (choose: `list`, `set`, `dict`).

```
a = []
b = {}
c = set()
d = dict()
```

- `a` is a _____
- `b` is a _____
- `c` is a _____
- `d` is a _____

3) Keys vs values (short answer)

Given:

```
pets = {"Mochi": "cat", "Boba": "dog", "Luna": "cat"}
```

1. List all **keys**: _____
2. List all **values**: _____

4) Duplicate keys

What happens if a dictionary has the same key twice?

```
d = {"A": 1, "B": 2, "A": 99}
print(d)
print(d["A"])
```

Hint:

- When Python reads this literal: `{"A": 1, "B": 2, "A": 99}`, it processes the pairs left to right.
- The second "A" is the same key as the first "A", so it overwrites the earlier value.
- In `d`, the key `"A"` maps to value `99`.

Output:

```

```

Part B — Membership check + lookup

5) `in` checks keys only (what does it print?)

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

print("Ben" in scoreboard)
print(5 in scoreboard)
print("Dylan" in scoreboard)
```

Output:

```

```

6) Safe look up (fill in the blanks)

Complete the code so it prints the score if the player exists, otherwise prints `"Player not found!"`.

Choose from: `in`, `not in`, `scoreboard[player]`, `"Player not found!"`.

```
scoreboard = {"Amy": 3, "Ben": 5}
player = "Dylan"

if player __________ scoreboard:
    print(_____________________)
else:
    print(_____________________)
```

7) Look up by key (what does it print?)

```
scores = {"Amy": 2, "Ben": 4, "Chloe": 1}

print(scores["Ben"])
print(scores["Amy"] + scores["Chloe"])
```

Output:

8) Look up

Python code:

```
scores = {"Amy": 2, "Ben": 4}
print(scores["Chelsea"])
```

Does the code has error? Explain.

_____

_____

Part C — Update + add keys

9) Update scores (what is the final dictionary?)

```
scoreboard = {"Amy": 2, "Ben": 2}
scoreboard["Ben"] = scoreboard["Ben"] + 1
scoreboard["Chloe"] = 1
scoreboard["Ben"] = scoreboard["Ben"] + 1

print(scoreboard)
```

Output:

10) Add or update? (what does it print?)

```
d = {"x": 1}
d["y"] = 5
d["x"] = 9
print(d)
```

Output:

```

```

11) Write code

Goal:

- If `"Chloe"` is in the dictionary, add her score by `1`.
- If not, set her score to `1`.

```
scoreboard = {"Amy": 1, "Ben": 2}

if ____________________:
    scoreboard["Chloe"] = ____________________________
else:
    scoreboard["Chloe"] = ____________________________

print(scoreboard)
```

12) Scoreboard updates from a list (write code)

We have a list of players who scored 1 point each time they appear.

Example:

- If `"Amy"` appears 3 times, Amy gets 3 points.
- If `"Chloe"` appears 1 time, Chloe gets 1 point.

Write code to build a dictionary `scoreboard` that counts points.

```
events = ["Amy", "Ben", "Amy", "Chloe", "Amy", "Ben"]

# Write code here:
# scoreboard should become: {"Amy": 3, "Ben": 2, "Chloe": 1}

scoreboard = {}

for name in events:
    if name not in scoreboard:
        scoreboard[name] = ___________________________________
    else:
        scoreboard[name] = ___________________________________

print(scoreboard)
```

Expected output:

```
{'Amy': 3, 'Ben': 2, 'Chloe': 1}
```

Part D — Counting with dictionaries (loops)

13) Count letters (fill in the blanks)

Complete the code so it counts how many times each letter appears.

Choose from: `in`, `not in`, `= 1`, `= d[ch] + 1`.

```
letters = ["A", "B", "A", "C", "B", "A"]
d = {}

for ch in letters:
    if ch __________ d:
        d[ch] __________
    else:
        d[ch] __________

print(d)
```

Expected output:

```
{'A': 3, 'B': 2, 'C': 1}
```

14) What does it print? (counting)

```
words = ["hi", "hi", "bye", "hi"]
count = {}

for w in words:
    if w not in count:
        count[w] = 1
    else:
        count[w] = count[w] + 1

print(count)
```

Output:

```

```

15) Find the highest score (fill in the blanks)

Complete the code to find the player with the highest score.

(Assume there is no tie.)

```
scoreboard = {"Amy": 2, "Ben": 5, "Chloe": 1}

best_player = ""
best_score = -1

for player in scoreboard:  # visit every key: "Amy", "Ben", "Chloe"
    score = scoreboard[player]
    if score > best_score:
        best_score = __________
        best_player = __________

print(best_player, best_score)
```

Expected output:

```
Ben 5
```

16) Merge two scoreboards (write code)

We have two scoreboards. Add scores from `b` into `a`.

- If the player already exists in `a`, add the score.
- If not, create a new key.

Example: If `a["Ben"]` is 2 and `b["Ben"]` is 3, then `a["Ben"]` becomes 5.

```
a = {"Amy": 2, "Ben": 2}
b = {"Ben": 3, "Chloe": 1}

for player in b:  # visit every key of b: "Ben", "Chloe"
    if player in a:
        a[player] = _____________________________
    else:
        a[player] = _____________________________

print(a)
```

Expected output:

```
{'Amy': 2, 'Ben': 5, 'Chloe': 1}
```