Worksheet 27: Python Dictionary Tools

Name: _____ Date: _____

Instructions

- Answer in the blanks.
- For "write code" questions, write valid Python code.
- For "what does it print" questions, write the exact output.
- Assume dictionaries keep **insertion order** (Python 3.7+), so iteration order matches how keys were added.

Part A — Review: `items()` (key, value pairs)

1) What does it print?

```python
scoreboard = {"Amy": 7, "Ben": 4, "Chloe": 9}

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

Output:

2) Fill in the blanks: loop over pairs

Fill in the blanks so the loop prints each key and value:

```
pets = {"cat": 2, "dog": 1, "fish": 5}

for _____, _____ in pets.items():
    print(_____, _____)
```

3) Write code: print with a dash

Write code to print each pair like `key - value` (one per line).

```
prices = {"apple": 3, "banana": 2, "cookie": 5}

# complete the code:
```

Expected output:

```
apple - 3
banana - 2
cookie - 5
```

Part B — `len(d)` (how many entries)

4) What does it print? (`len` after updates)

```
d = {"red": 1, "blue": 2, "green": 3}

d["blue"] = 99
d["yellow"] = 4

print(len(d))
```

Output:

5) Fill in the blank: print how many players

```
scoreboard = {"Amy": 7, "Ben": 4, "Chloe": 9, "Dan": 4}

print( _____ )
```

Expected output:

```
4
```

6) True or False (circle one)

For a dictionary `d` :

- `len(d)` counts how many **keys** are in the dictionary. True / False

- Updating an existing key changes `len(d)` . True / False

- Adding a new key changes `len(d)` . True / False

Write your answers:

1) _____

2) _____

3) _____

Part C — `keys()` and `list(d.keys())`

7) Fill in the blank: keys as a list

```
scoreboard = {"Amy": 7, "Ben": 4, "Chloe": 9, "Dan": 4}

players = _____

print(players)
print(len(players))
```

Expected output (order matches insertion order):

```
['Amy', 'Ben', 'Chloe', 'Dan']
4
```

8) List vs dictionary: what changes?

Read the code and answer the questions.

```
d = {"A": 1, "B": 2}
keys_list = list(d.keys())

keys_list.append("C")
d["C"] = 3

print(keys_list)
print(list(d.keys()))
```

1) What does the first `print` show?

2) What does the second `print` show?

9) Write code: check if a key exists (using `keys()`)

Fill in the blank so it prints `True` if `"Ben"` is a key in the dictionary.

```
scoreboard = {"Amy": 7, "Ben": 4, "Chloe": 9}

print( "Ben" in _____ )
```

Expected output:

```
True
```

Part D — `values()` and adding them

10) What does it print? (traverse values)

```
d = {"x": 10, "y": 20, "z": 30}

for v in d.values():
    print(v)
```

Output:

---

11) Write code: total score

Compute the total of all scores. Print the total.

```
scoreboard = {"Amy": 7, "Ben": 4, "Chloe": 9}

total = _____
for score in _____:
    total = _____

print(_____)
```

Expected output:

```
20
```

---

12) Write code: average score

Compute the average score and print it. (Use `len(scoreboard)` for how many players.)

```
scoreboard = {"Amy": 7, "Ben": 4, "Chloe": 9, "Dan": 4}

total = _____

for score in _____:
    total = _____

n = _____
average = total _____ n   # / or //?

print(average)
```

Expected output:

```
6.0
```

Part E — Mixed practice (use the tools together)

13) Count "big" scores

Count how many players have a score **greater than or equal to 8**.

```
scoreboard = {"Amy": 7, "Ben": 4, "Chloe": 9, "Dan": 8, "Eli": 10}

count = 0
for score in _____:
    if _____:
        _____

print(count)
```

Expected output:

```
3
```

14) Highest score (find the winner)

Fill in the blanks to find the player with the highest score. Assume there is only one winner.

```
scoreboard = {"Amy": 7, "Ben": 4, "Chloe": 9, "Dan": 8}

best_name = ""
best_score = -1

for name, score in scoreboard.items():
    if score > best_score:
        best_score = score
        best_name = name

print(best_name, best_score)
```

Output:

15) Find all players with score 4

Write code to collect all names whose score is 4 into a list named result . Then print result .

```
scoreboard = {"Amy": 7, "Ben": 4, "Chloe": 9, "Dan": 4, "Eli": 10}

result = []

for name, score in _____:
    if _____:
        result.append(name)

print(result)
```

Expected output (order matches insertion order):

```
['Ben', 'Dan']
```

16) Mini project: snack inventory summary

You have a dictionary that stores how many snacks you have.

1. Print the number of different snack types.
2. Print a list of snack names (keys as a list).
3. Print the total number of snacks (add up the values).

```python
snacks = {"chips": 2, "cookies": 5, "apples": 3}

# 1. Print the number of different snack types.
print("types:", _____)

# 2. Print a list of snack names (keys as a list).
names = _____
print("names:", names)

# 3. Print the total number of snacks (add up the values).
total = _____
for count in _____:
    total = _____
print("total:", total)
```

Expected output (order matches insertion order):

```
types: 3
names: ['chips', 'cookies', 'apples']
total: 10
```