Solution 24: Python Dictionary Basics

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:

```
{'Amy': 3, 'Ben': 5}
<class 'dict'>
```

2) Empty containers (fill in the types)

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

- `a` is a **list**
- `b` is a **dict**
- `c` is a **set**
- `d` is a **dict**

3) Keys vs values (short answer)

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

1. Keys: `Mochi`, `Boba`, `Luna`
2. Values: `cat`, `dog`, `cat`

---

4) Duplicate keys (what does it print?)

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

Output:

```
{'A': 99, 'B': 2}
99
```

Explanation: the second `"A": 99` overwrites the first `"A": 1`.

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:

```
True
False
False
```

---

6) Safe look up (fill in the blanks)

One correct solution:

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

if player in scoreboard:
    print(scoreboard[player])
else:
    print("Player not found!")
```

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:

```
4
3
```

8) Spot the error (short answer)

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

Error: **`KeyError`** (because `"Chelsea"` is not a key).

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:

```
{'Amy': 2, 'Ben': 4, 'Chloe': 1}
```

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

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

Output:

```
{'x': 9, 'y': 5}
```

11) Write code

One correct solution:

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

if "Chloe" in scoreboard:
    scoreboard["Chloe"] = scoreboard["Chloe"] + 1
else:
    scoreboard["Chloe"] = 1

print(scoreboard)
```

12) Scoreboard updates from a list (write code)

One correct solution:

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

scoreboard = {}

for name in events:
    if name not in scoreboard:
        scoreboard[name] = 1
    else:
        scoreboard[name] = scoreboard[name] + 1

print(scoreboard)
```

Expected output:

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

Part D — Counting with dictionaries (loops)

13) Count letters (fill in the blanks)

Filled code:

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

for ch in letters:
    if ch not in d:
        d[ch] = 1
    else:
        d[ch] = d[ch] + 1

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:

```
{'hi': 3, 'bye': 1}
```

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

Filled code:

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

best_player = ""
best_score = -1

for player in scoreboard:
    score = scoreboard[player]
    if score > best_score:
        best_score = score
        best_player = player

print(best_player, best_score)
```

Expected output:

```
Ben 5
```

16) Merge two scoreboards (write code)

15) Merge two scoreboards (write code)

One correct solution:

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

for player in b:
    if player in a:
        a[player] = a[player] + b[player]
    else:
        a[player] = b[player]

print(a)
```

Expected output:

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