Worksheet 12: Python Nested For-Loop

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).

Part A — Nested loop basics

1) Outer vs inner (how many times?)

Fill in the blanks.

- The **outer** loop runs _____ times.
- The **inner** loop runs _____ times for each outer step.
- Total number of lines printed = _____.

```
for outer in [1, 2, 3]:
    for inner in ["A", "B"]:
        print(outer, inner)
```

2) What does it print?

```
xs = [1, 2]
ys = [10, 20]

for x in xs:
    for y in ys:
        print(x + y)
```

Output:

```
```

3) Build all coordinate pairs (fill in the blanks)

Goal: build this list:

```
[[0, 0], [0, 1], [0, 2], [1, 0], [1, 1], [1, 2]]
```

Complete the code:

```
result = []

for x in [0, 1]:
    for y in [0, 1, 2]:
        result.append([____, ____])

print(result)
```

Part B — 2D lists (list of lists)

4) 2D list print (what does it print?)

```
words = [
    ["hi", "bye"],
    ["yes", "no"]
]

for row in words:
    for w in row:
        print(w)
```

Output:

```

```

5) Flatten a 2D list (fill in the blanks)

Goal:

```
flat = [2, 4, 6, 1, 3, 5]
```

Complete the code:

```
grid = [
    [2, 4, 6],
    [1, 3, 5]
]

flat = []

for _______ in grid:
    for _______ in _______:
        flat.append(_______)

print(flat)
```

6) Flatten but skip zeros (fill in the blanks)

Goal:

```
flat = [1, 2, 3]
```

Complete the code:

```
grid = [
    [0, 1, 0],
    [2, 0, 3]
]

flat = []

for row in grid:
    for x in row:
        if x ______ 0:
            flat.append(x)

print(flat)
```

7) Sum all numbers in a 2D list (what does it print?)

```
nums = [
    [1, 2, 3],
    [4, 5, 6]
]

total = 0

for row in nums:
    for x in row:
        total = total + x

print(total)
```

Output:

```

```

8) Count numbers > 5 (what does it print?)

```
grid = [
    [2, 7],
    [6, 5],
    [9, 1]
]

count = 0

for row in grid:
    for x in row:
        if x > 5:
            count = count + 1

print(count)
```

Output:

```

```

Part C — Build new lists with nested loops

9) Multiplication table (fill in the blanks)

Goal:

```
[[1, 2, 3], [2, 4, 6], [3, 6, 9]]
```

Complete the code:

```
numbers = [1, 2, 3]
table = []

for a in numbers:
    row = []
    for b in numbers:
        row.append(a _____ b)
    table.append(_____)

print(table)
```

10) Print a rectangle of stars (what does it print?)

```
rows = 3
cols = 5

for r in range(rows):
    line = ""
    for c in range(cols):
        line = line + "*"
    print(line)
```

Output:

```

```

11) Pairs in a different order (write code)

Goal:

```
["R1", "B1", "R2", "B2", "R3", "B3"]
```

Write code to produce the list above.

```
colors = ["R", "B"]
numbers = [1, 2, 3]

result = []

# write code here









print(result)
```

12) “Sandwich combos” (what does it print?)

```
breads = ["white", "wheat"]
fillings = ["ham", "egg", "cheese"]

combos = []

for b in breads:
    for f in fillings:
        combos.append(b + " + " + f)

print(combos)
```

Output:

```

```

Part D — Debugging and challenges

13) Fix the bug (row sums)

You are given a 2D list such as:

```
grid = [
    [1, 2, 3],
    [4, 5, 6]
]
```

You are required to compute the **row sum**:

- 1 + 2 + 3 = 6
- 4 + 5 + 6 = 15

So the expected output is:

```
[6, 15]
```

Chelsea wrote the following code, and it is **wrong**. It prints the wrong list.

```
grid = [
    [1, 2, 3],
    [4, 5, 6]
]

sums = []
row_sum = 0

for row in grid:
    for x in row:
        row_sum = row_sum + x
    sums.append(row_sum)

print(sums)
```

The output of the **wrong code** is:

```

```

To get the expected output, you can add or move **one line of code** to fix the wrong code. Please tell Chelsea how to fix the code:

```

```

14) 1D list → 2D list (what does it print?)

```
nums = [1, 2, 3, 4, 5, 6]

grid = []
idx = 0

for r in range(2):
    row = []
    for c in range(3):
        row.append(nums[idx])
        idx = idx + 1
    grid.append(row)

print(grid)
```

Output:

```

```

15) Find the largest number in a 2D list (fill in the blanks)

Complete the code so it prints the largest number.

```
grid = [
    [3, 7, 2],
    [9, 1, 5]
]

max_val = grid[0][0]

for row in grid:
    for x in row:
        if x > max_val:
            max_val = _____

print(max_val)
```

16) Make a list of coordinates (what does it print?)

```
result = []

for r in range(2):
    for c in range(3):
        result.append(str(r) + "," + str(c))

print(result)
```

Output:

```

```