

Worksheet 09: Python For-Loop with Break

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.
- ////////////////////////////////////

Part A — Review: list traversal

1) What does it print?

```
numbers = [3, 1, 4]
for x in numbers:
    print(x)
```

Output:

////////////////////////////////////

2) Write code: traverse and print

Write Python code to print every item in this list, one per line:

```
animals = ["cat", "dog", "fish"]  
  
# complete the code:
```

Expected output:

```
cat  
dog  
fish
```

Part B — What does `break` do?

3) Stop early

What is the output?

```
for x in [1, 2, 3, 4, 5]:  
    if x == 3:  
        break  
    print(x)
```

Output:

4) `break` with printing first

What is the output?

```
for x in [1, 2, 3, 4, 5]:
    print(x)
    if x == 3:
        break
```

Output:

Part C — Search a number (Boolean `found`)

5) Fill in the blanks.

We want to check if `target` is in `numbers` .

Fill in the blanks:

- Start with `found = _____` (`True` or `False` ?)
- If we see the target, set `found = _____` (`True` or `False` ?)
- At the end, we print `found` .

6) Search without break: what does it print?

```
numbers = [4, 7, 2, 9, 5]
target = 9

found = False

for x in numbers:
    if x == target:
        found = True

print(found)
```

Output:

7) Search with break: what does it print?

```
numbers = [4, 7, 2, 9, 5]
target = 2

found = False

for x in numbers:
    if x == target:
        found = True
        break

print(found)
```

Output:

////////////////////////////////////

8) Write code: search with break

Write Python code to check if `target` is in `numbers` .

- If found, print `True`
- If not found, print `False`
- Use `break`

```
numbers = [6, 1, 8, 3]
target = 10

# complete the code:
```

Part D — Search a student name (print index or -1)

9) What is the output?

```
students = ["Ava", "Ben", "Cody", "Dina"]
target = "Dina"

result = -1
n = len(students)

for i in range(n):
    if students[i] == target:
        result = i
        break

print(result)
```

Output:

10) Write code: find the index (or -1)

Write Python code to search for `target` in `students` .

- If found, print the index
- If not found, print `-1`
- Use `break`

```
students = ["Ava", "Ben", "Cody", "Dina"]
target = "Ben"

# complete the code:
```

11) Not found: what is the output?

```
students = ["Ava", "Ben", "Cody", "Dina"]
target = "Eli"

result = -1
n = len(students)

for i in range(n):
    if students[i] == target:
        result = i
        break

print(result)
```

Output:

Part E — Negative indices

12) What does it print?

```
nums = [10, 20, 30, 40]
print(nums[-1])
print(nums[-2])
```

Output:

////////////////////////////////////

13) Fill in the blanks

Given:

```
letters = ["a", "b", "c", "d", "e"]
```

Fill in:

- `letters[-1]` is _____
- `letters[-3]` is _____

////////////////////////////////////

Part F — Fibonacci numbers (break when too big)

14) What does it print?

```
fib = [1, 1]

for _ in range(1000):
    nxt = fib[-2] + fib[-1]
    if nxt > 20:
        break
    fib.append(nxt)

print(fib)
```

Output:

15) Write code: Fibonacci up to 50

Write Python code to build a list of Fibonacci numbers that are `<= 50` , then print the list.

```
fib = [1, 1]

# complete the code:
```

Part G — Mini challenge (use `break`)

16) First even number

Write Python code to find the **first even number** in the list.

- If you find an even number, print it and stop the loop using `break` .
- If there is **no even number**, print `"No even"` .


```
numbers = [7, 9, 5, 12, 3, 8]
```

```
# complete the code:
```