Tasks – for Loops and range()

Practice `for` loops with `range()`: one argument, two arguments, step, countdown, and combinations with input, variables, and conditions. Create each file, run it, and check the output.

Run scripts with: `python3 script_name.py`

Part 1 – Basics: for and range()

Task 1.1 – range(stop): 0 to n-1 (`for_range_basic.py`)

- Create `for_range_basic.py`.
- Use `for i in range(5):` and in the body print `i`. Run the script. You should see 0, 1, 2, 3, 4.

Expected output:

```
0
1
2
3
4
```

Task 1.2 – range(start, stop): 1 to n (`for_range_start_stop.py`)

- Create `for_range_start_stop.py`.
- Use `for i in range(1, 6):` and print `i` each time. The loop runs with i = 1, 2, 3, 4, 5 (stops before 6).

Expected output:

```
1
2
3
4
5
```

Task 1.3 – range with step (`for_range_step.py`)

- Create `for_range_step.py`.
- Use `for i in range(0, 10, 2):` and print `i`. You should get 0, 2, 4, 6, 8 (step 2).

Expected output:

```
0
2
4
6
8
```

Task 1.4 – range countdown (negative step) (`for_countdown.py`)

- Create `for_countdown.py`.
- Use `for i in range(5, 0, -1):` and print `i`. Then after the loop print `"Go!"`. You should see 5, 4, 3, 2, 1, then Go!

Expected output:

```
5
4
3
2
1
Go!
```

Task 1.5 – Print 1 to N with input (`for_one_to_n.py`)

- Create `for_one_to_n.py`.
- Read an integer `n` with `input()` and convert to `int`. Use `for i in range(1, n + 1):` and print `i`. Run and enter e.g. 4.

Expected output (if you enter 4):

```
Enter n: 4
1
2
3
4
```

Task 1.6 – Sum 1 to N with for (`for_sum.py`)

- Create `for_sum.py`.
- Read an integer `n`. Use a variable `total = 0`. Loop with `for i in range(1, n + 1):` and add `i` to `total`. After the loop print `total`. For n=5 you should get 15.

Expected output (n = 5):

```
Enter n: 5
15
```

Part 2 – More for and range

Task 2.1 – Repeat a message N times (`for_repeat.py`)

- Create `for_repeat.py`.
- Read an integer `n`. Use `for _ in range(n):` (underscore if you don't use the loop variable) and print `"Hello"` each time. Run with n=3.

Expected output (n = 3):

```
How many? 3
Hello
```

```
Hello
Hello
```

Task 2.2 – Even numbers from 0 to N (`for_evens.py`)

- Create `for_evens.py`.
- Read an integer `n`. Use `for i in range(0, n + 1, 2):` and print `i`. For n=8 you get 0, 2, 4, 6, 8.

Expected output (n = 8):

```
Enter n: 8
0
2
4
6
8
```

Task 2.3 – range(2, 11, 2) (`for_two_to_ten.py`)

- Create `for_two_to_ten.py`.
- Use `for i in range(2, 11, 2):` and print `i`. Then print one line with the sum of those numbers (2+4+6+8+10 = 30). Use a variable to accumulate the sum inside the loop.

Expected output:

```
2
4
6
8
10
30
```

Part 3 – for with input and real tasks

Task 3.1 – Multiplication table (1 to 10) (`for_times_table.py`)

- Create `for_times_table.py`.
- Read a number `num` (int). Use `for i in range(1, 11):` and print each line as `num * i` (e.g. "7 x 1 = 7"). Use variables and `str()` to build the message: `str(num) + " x " + str(i) + " = " + str(num * i)`.

Expected output (num = 7, first 3 lines):

```
Enter a number: 7
7 x 1 = 7
7 x 2 = 14
7 x 3 = 21
...
7 x 10 = 70
```

Task 3.2 – Average of N scores (`for_scores_avg.py`)

- Create `for_scores_avg.py`.
- Ask "How many scores?" and read `n`. Use `total = 0` and `for i in range(n):` to read a score each time (convert to float), add to total. After the loop compute `average = total / n` and print "Average: " + str(round(average, 1)). Use a variable for the current score.

Expected output (example – 3 scores 80, 90, 70):

```
How many scores? 3
Score 1: 80
Score 2: 90
Score 3: 70
Average: 80.0
```

Task 3.3 – Print a line of characters (`for_line.py`)

- Create `for_line.py`.
- Read an integer `n` and a character (e.g. ask "Character?" and read one character or a string like "*"). Use `for i in range(n):` and inside the loop use `print(char, end="")` so*

they appear on one line. After the loop print a newline (e.g. `print()`*). Run with 5 and ""* to get `*****`.

Expected output (n=5, char *):

```
Length: 5
Character: *
*****
```

Task 3.4 – List of names and greeting (`for_names.py`)

- Create `for_names.py`.
- Ask "How many names?" and read `n`. Use `for i in range(n):` to ask for a name each time (e.g. "Name 1:", "Name 2:", ...). Build the prompt with `"Name " + str(i+1) + ": "`. Print each name with a greeting (e.g. "Hello, Alice") using the variable and string concatenation.

Expected output (example – 2 names):

```
How many names? 2
Name 1: Ali
Hello, Ali
Name 2: Bo
Hello, Bo
```

Task 3.5 – Sum of N entered numbers (`for_sum_input.py`)

- Create `for_sum_input.py`.
- Ask "How many numbers?" and read `n`. Use `total = 0` and `for i in range(n):` to read a number each time (int or float), add it to total. After the loop print "Sum: " + str(total). Use variables for the current number and total.

Expected output (example – 3 numbers 10, 20, 30):

```
How many numbers? 3
Number 1: 10
Number 2: 20
Number 3: 30
Sum: 60
```

Done

You've used: `for i in range(stop)`, `range(start, stop)`, `range(start, stop, step)`, negative step for countdown, accumulation in a for loop, and combined for + range with input(), variables, str(), and print(end="").