Prepared by *Coderistic*

Chapter 3: Control Flow and Loops

Lesson 3: Loops

For Loops

Used to iterate over a sequence (list, string, range, etc.).

```
# Basic for loop with a list
fruits = ["apple", "banana", "cherry"]
for fruit in fruits:
    print(fruit)

# Using range()
for i in range(5):  # Prints 0 to 4
    print(i)
```

While Loops

Repeats as long as a condition is true.

```
# Basic while loop
count = 0
while count < 5:
    print(count)
    count += 1  # Increment to avoid infinite loop
```

Key Controls

- `break`: Exit the loop early.
- `continue`: Skip to the next iteration.

```
for i in range(10):
    if i == 3:
        continue  # Skips 3
    if i == 7:
        break    # Stops at 7
    print(i)
```

Exercises for practice

1. Sum of Sales

Task: Calculate the total of a list of sales amounts.
Input: `[25.50, 14.99, 33.25, 9.75]`
Expected Output: `Total: $83.49`
Hint: Use a `for` loop to iterate over the list and add each amount to a running total.

---

2. Even Number Printer

Task: Print all even numbers from 1 to 20.
Expected Output: `2 4 6 8 10 12 14 16 18 20`
Hint: Use a `for` loop with `range()` and check if each number is even using `%`.

---

3. Password Retry Limit

Task: Ask the user for a password until they enter "secret" or reach 3 attempts.
Expected Output:

- If correct: `Access granted!`
- If 3 attempts fail: `Locked out!`
  Hint: Use a `while` loop with a counter and `input()`.

---

4. Inventory Cleanup

Task: From a list of stock quantities, print only items with more than 0 in stock, stopping if a negative value is found.
Input: `[5, 12, 0, 8, -1, 10]`
Expected Output: `In stock: 5 In stock: 12 In stock: 8 Negative stock detected!`
Hint: Use a `for` loop with `continue` and `break`.

---

5. Email Validator

Task: Check a list of emails and print "Valid" if they contain "@", otherwise skip them.
Input: `["user@domain.com", "no-at-sign", "test@site.org", "plain"]`
Expected Output: `Valid: user@domain.com Valid: test@site.org`
Hint: Use a `for` loop and `continue` to skip invalid emails.

---

5. Countdown Timer

Task: Simulate a countdown from 10 to 1, printing each number, then "Liftoff!" at the end.
Expected Output: `10 9 8 7 6 5 4 3 2 1 Liftoff!`
Hint: Use a `while` loop with a decreasing counter.
