Operations and variables

1. By one dimension, we mean the object is moving in a straight line. Five variables are put together in several equations to describe this motion:

$$v_f = v_0 + at$$
$$\bar{v} = \frac{v_f + v_0}{2} \quad \bar{v} \text{ is the average speed}$$
$$d = \bar{v}t$$
$$d = v_0 t + \frac{at^2}{2}$$
$$v_1^2 = v_0^2 + 2ad,$$

   where v_1 is the final velocity, v_0 is the initial velocity, t is the time elapsed, a is the object's acceleration and d is the distanced traveled.

   Suppose a ball is released from the top of a building. Write a program to find out the height of the building, given that the ball arrives at the ground after 2.5 seconds. Note that the acceleration due to gravity is constant at 9.8.

2. The force pushing or pulling an object is related to the object's mass, acceleration, and coefficient of friction in the following formula:

$$F = \mu mg + ma,$$

   where F is the force applied to push or pull an object, μ is the coefficient of friction, m is the object's mass, g is the acceleration due to gravity, which is a constant $9.8\,m/s^2$, a is the object's acceleration.

   Given that pulling an object with a mass of 24.5 kg and an acceleration of 4.5 m/s^2 requires a force of 150 Newtons, calculate the coefficient of friction for the surface on which the object is moving.

3. Write a program that displays the slope of the line connecting the two points $(x_1, y_1) = (4.5, -5.5)$ and $(x_2, y_2) = (6.6, -6.5)$. The formula of the slope is

$$\frac{y_2 - y_1}{x_2 - x_1} \tag{1}$$

4. Write a program that displays 123456789 in reverse order.

Tuples

1. Create a tuple named

   `admin_credentials`

   that holds the values "admin", "root", and "P@ssw0rd". Print the tuple and its length.

2. Given the tuple

```
ports = (22, 443, 80, 21, 8080),
```

   print the first and last port. Check if port 22 is in the tuple.

3. Try to modify the second value in the tuple

```
user_roles = ("admin", "user", "guest")
```

   to "superuser". What error do you get?

4. Given the tuple

```
server_info = ("192.168.1.10", "web_server", "active"),
```

   unpack it into three variables: ip, role, and status, then print each variable.

Dictionaries

5. Create a dictionary `firewall_rules` where:
   - "inbound" key maps to the list ["SSH", "HTTPS"].
   - "outbound" key maps to the list ["DNS", "SMTP"].

   Print the dictionary.

6. Using the `firewall_rules` dictionary from the previous exercise, print all the inbound and outbound rules. Also, use the `.get()` method to safely access a non-existing key "blocked".

7. Add a new key "blocked" to the `firewall_rules` dictionary with the value `["FTP", "Telnet"]`. Then, print the updated dictionary.

8. Create a dictionary `attack_signatures` where each attack type ("SQL Injection", "Cross-Site Scripting (XSS)", "Denial of Service (DoS)") maps to a corresponding signature (SQL query, JavaScript code, traffic overload). Add a new attack signature for "Brute Force".

Sets

9. Create a set called `detected_ips` that contains three IP addresses: "192.168.1.1", "10.0.0.2", "192.168.1.1". Print the set and observe the uniqueness property.

10. Create two sets:

```
secure_ips = {"192.168.1.1", "10.0.0.2"}
insecure_ips = {"10.0.0.2", "172.16.0.1"}
```

    Perform a union and intersection of the two sets. Print the results.

11. Using the `secure_ips` and `insecure_ips` sets from the previous exercise, find the difference between `secure_ips` and `insecure_ips` (i.e., IPs in `secure_ips` but not in `insecure_ips`).

12. You have two sets of hashes:

```
known_hashes = {"abc123", "def456", "ghi789"}.
suspicious_hashes = {"ghi789", "xyz123", "abc123"}.
```

    Find the common hashes and the hashes that are in `suspicious_hashes` but not in `known_hashes`.