PYTHON FOUNDATIONS: FROM BASICS TO PRACTICE

SESSION 3:
LOOPING AROUND:
LET’S REPEAT!

TABLE OF CONTENT

01 Introduction

02 while Loops

03 For Loops

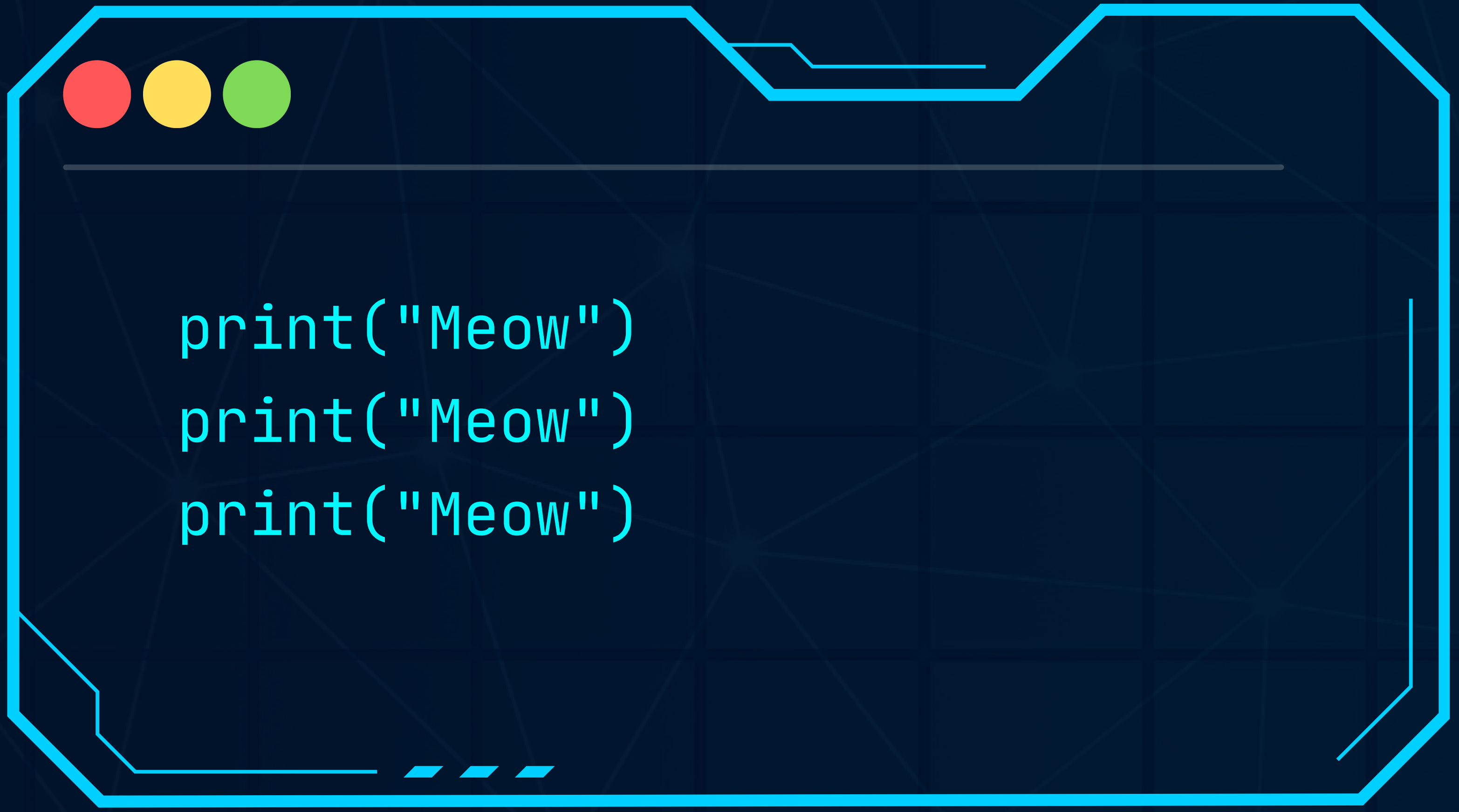
04 Lists

05 Coke Machine

INTRODUCTION

In the world of Python programming, loops are your trusty companions. They enable you to perform repetitive tasks efficiently. Imagine you have a specific action you want to perform multiple times; loops are your solution.

THE PROBLEM: MEOWING CATS

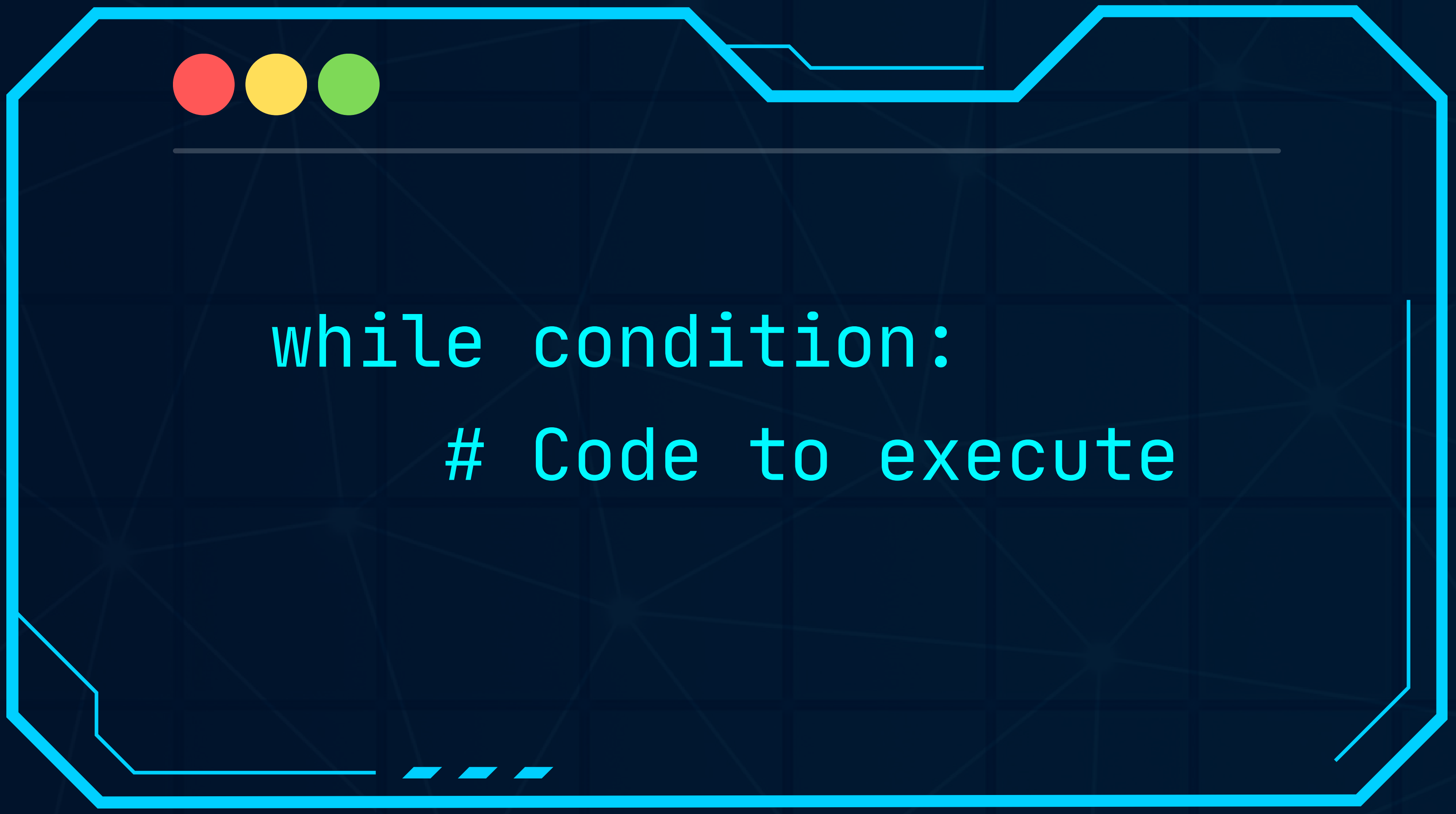


```
print("Meow")
print("Meow")
print("Meow")
```

Imagine you're building a virtual pet game, and you want to create a virtual cat that meows. To make it seem lifelike, you decide that your virtual cat should meow three times in a row when it's happy.

This code does the job, but what if you want your virtual cat to meow 500 times? Would it be logical to type print("Meow") over and over again 500 times? Certainly not!

02 WHILE LOOPS

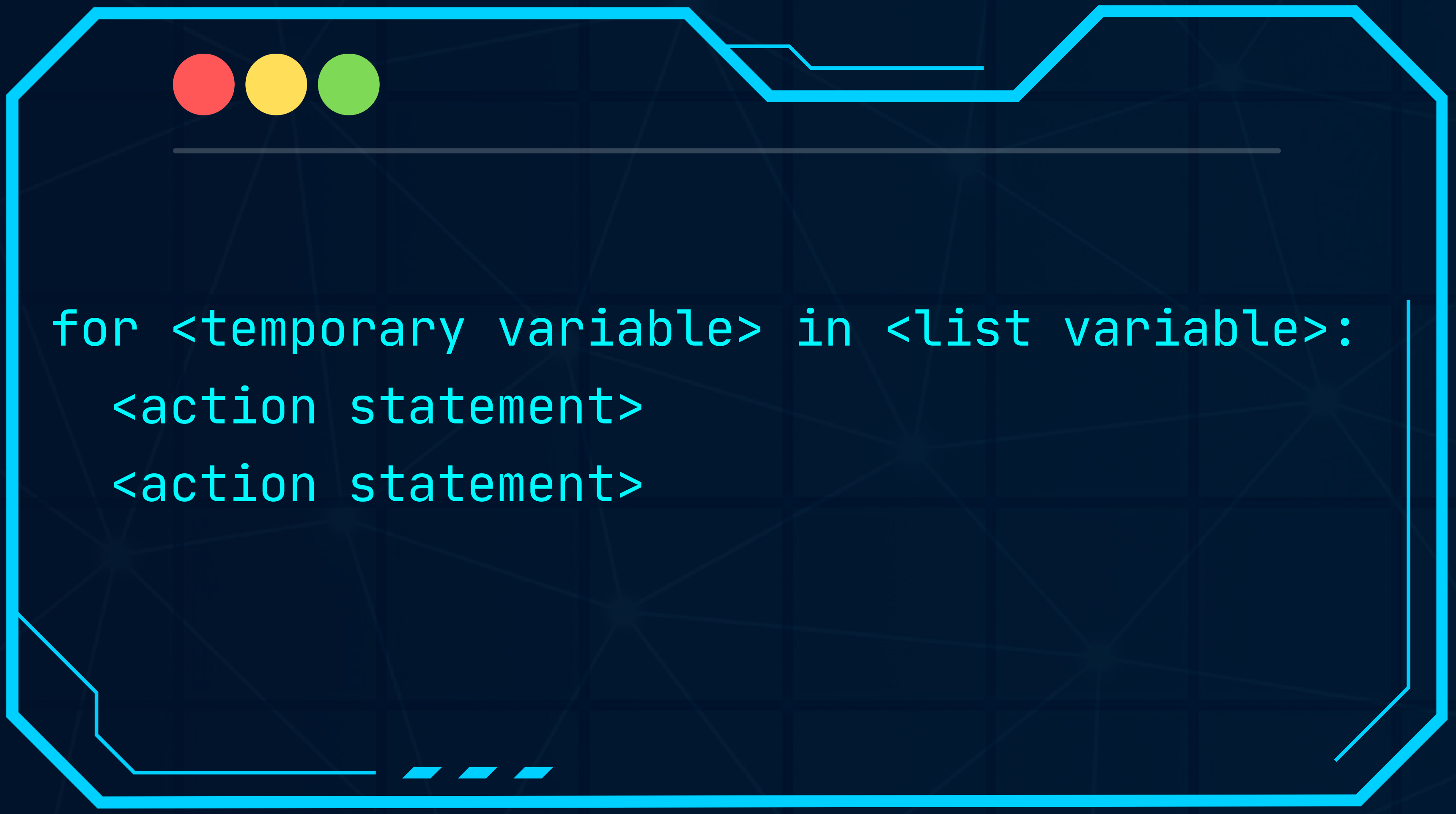


```
while condition:
    # Code to execute
```

In Python, while loops are a way to repeatedly execute a block of code as long as a specified condition remains true.

It's essential to understand that in a while loop, the condition is checked before the code block is executed. If the condition is false from the start, the code block will never run. This means that a while loop may not run at all if the initial condition is false.

03 FOR LOOPS

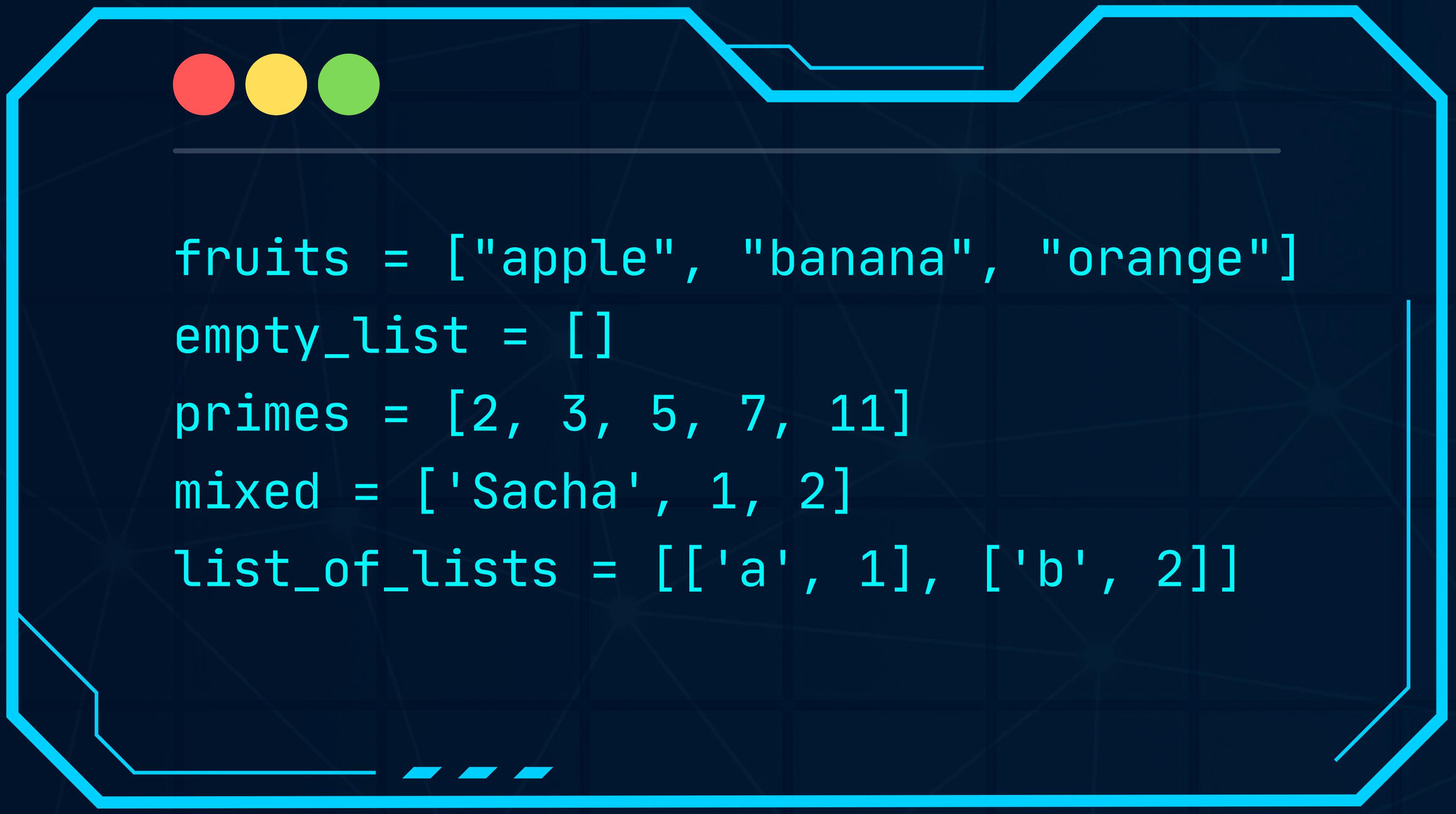


```
for <temporary variable> in <list variable>:
  <action statement>
  <action statement>
```

In Python, for loops are a way to repeatedly perform a set of actions on each item in a sequence, such as a list or string.

It's important to note that for loops iterate through each item in the sequence one by one and execute the specified actions for each item. This allows you to efficiently work with lists, strings, and other iterable objects in Python.

04 LISTS



```python
fruits = ["apple", "banana", "orange"]
empty_list = []
primes = [2, 3, 5, 7, 11]
mixed = ['Sacha', 1, 2]
list_of_lists = [['a', 1], ['b', 2]]
```

Python's lists are precisely that—ordered collections of items that make working with data a breeze.

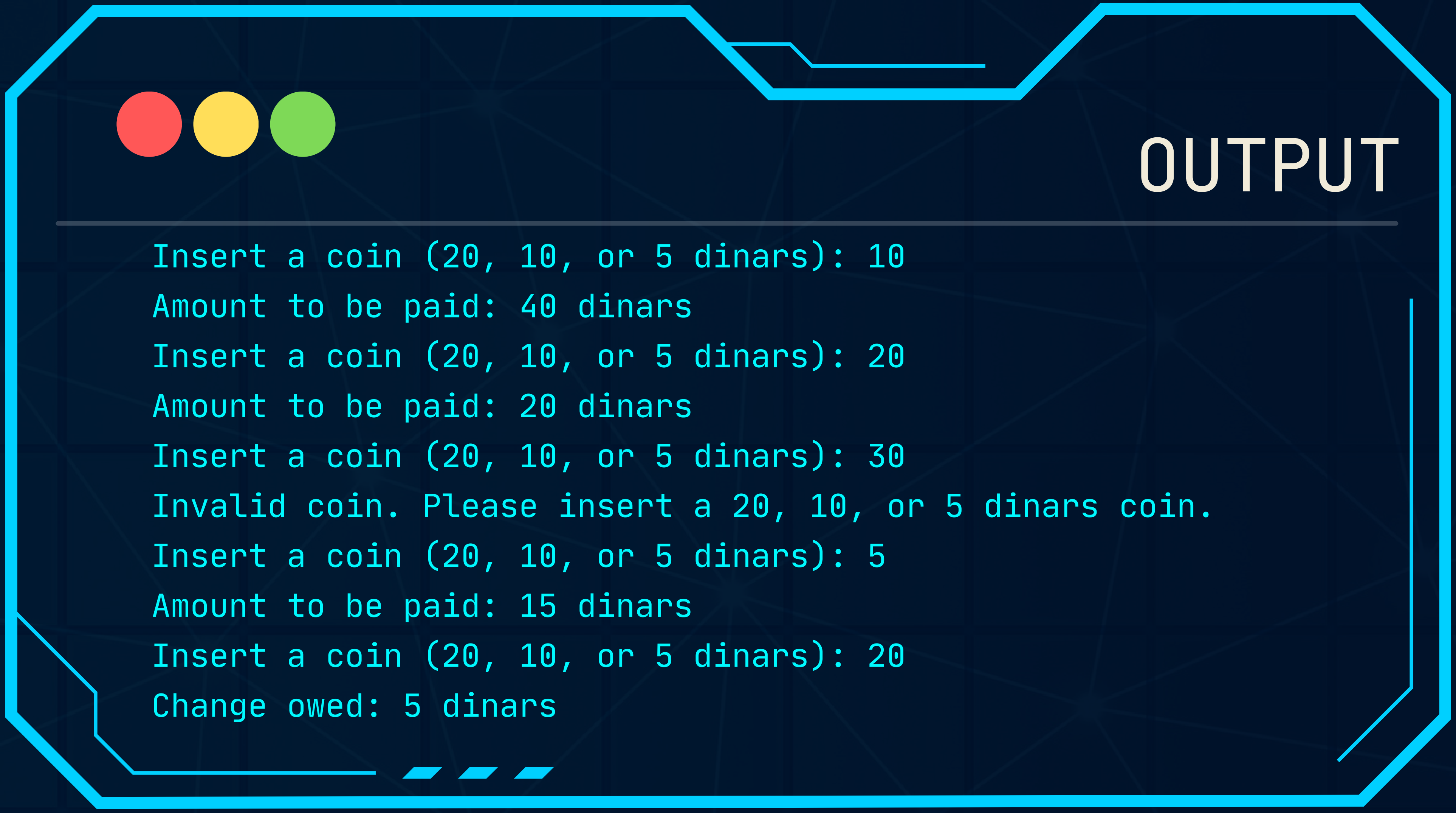
You can put any data values inside lists, and they are enclosed in square brackets [ ], with commas separating the values.

It's a good habit to include spaces after the commas for clarity. Lists can contain repeated values; uniqueness is not a requirement.

05 COKE MACHINE

Let's envision a vending machine in Algeria that offers Coca-Cola (Coke) bottles for 50 Algerian dinars. This machine exclusively accepts coins in the following denominations: 20 dinars, 10 dinars, and 5 dinars.

Your objective is to create a program that requests the user to insert coins one at a time. After each coin insertion, the program should inform the user of the remaining amount due. Once the user has inserted a minimum of 50 dinars, the program should display the amount of change owed to the user.



OUTPUT

```
Insert a coin (20, 10, or 5 dinars): 10
Amount to be paid: 40 dinars
Insert a coin (20, 10, or 5 dinars): 20
Amount to be paid: 20 dinars
Insert a coin (20, 10, or 5 dinars): 30
Invalid coin. Please insert a 20, 10, or 5 dinars coin.
Insert a coin (20, 10, or 5 dinars): 5
Amount to be paid: 15 dinars
Insert a coin (20, 10, or 5 dinars): 20
Change owed: 5 dinars
```

HAPPY CODING AHEAD!

Every line of code you write is a step towards mastering Python. Keep practicing, keep exploring, and keep pushing your boundaries.

Thank you for joining us on this journey!

Sarah Haddad
haddads686@gmail.com