🐍 04: Lists and Dictionaries

Instructions:

Write code to solve each of these problems.
Problems have been split into two chunks.
Write part 1 in a file called `fruits.py`, and part 2 in a file called `grades.py`.

Keep PEP8 in mind and make your code easy to read with meaningful variable names.

> Remember: code that you haven't run or tested is not trustworthy!

Reference

Lists:

```
a_list = ["a", "mixed", "collection", "of", "things", 5, 4, 3, 2, 1]
```

Accessed by index:

```
print(a_list[0])
>>> a
```

Dictionaries

```
a_dict = { "description": "A collection with structured keys",
"score": 123 }
```

Accessed by key/name:

```
print(a_dict["score"])
>>> 123
```

Accessing and Adding Data

A List of Fruits

1. Create a List of Fruits

Create a list called `fruits` with at least 3 fruit names.
Print the list.

2. Access an Element by Index

Print the second item in the `fruits` list.

3. Add an Item to a List

Add a new fruit to the `fruits` list and print the updated list.

4. Replace an Item in a List

Change the first fruit in the `fruits` list to a different fruit. Print the updated list.

5. Check if an Item Exists in a List

Check if "apple" is in the `fruits` list and print the result.

6. Print All Fruits

Use a loop to print each fruit in the `fruits` list.

7. Count Items in a List

Use a loop to count how many items are in the `fruits` list (without using `len()`).
You could write a function called "list_length", if you like.

8. Find the Longest Fruit Name

Use a loop to find and print the fruit with the longest name in the `fruits` list.

9. Create a List of Fruit Name Lengths

Use a loop to create a new list that contains the length of each fruit name.

A Dictionary of Grades

10. Create a Dictionary of Student Grades

Create a dictionary called `grades` with 3 students and their grades. Print the dictionary.
The dictionary keys will be student names, and the values will be a number from 0-100.

11. Access a Value by Key

Print the grade of one specific student from the `grades` dictionary.

12. Add a New Key-Value Pair

Add a new student and their grade to the `grades` dictionary. Print the updated dictionary.

13. Update a Value in a Dictionary

Change the grade of one student in the `grades` dictionary. Print the updated dictionary.

14. Check if a Key Exists in a Dictionary

Check if "Alice" is a key in the `grades` dictionary and print the result.

15. Print All Student Names

Use a loop to print all the keys (student names) in the `grades` dictionary.

16. Print All Grades

Use a loop to print all the values (grades) in the `grades` dictionary.

17. Print All Key-Value Pairs

Use a loop to print each student and their grade from the `grades` dictionary.

18. Calculate the Average Grade

Use a loop to calculate and print the average grade from the `grades` dictionary.

> **Note**
>
> Averages are calculated by adding everything together and then dividing by the number of things that there are.
> Hint: Always remember that you can create an 'accumulator' variable.

19. Find Students with Grade Above 80

Use a loop to print the names of students who scored above 80.

Extension

20. Create a Dictionary of Word Counts

Ask the user to enter a sentence. Create a dictionary where each word is a key and its value is the number of times it appears.
Test sentences:

- "The quick brown fox jumped over the lazy dog"
- "Rose rose to put rose roes on her rows of roses"
- "James while John had had had had had had had had had had had a better effect on the teacher"

> We'll do something like this again when we do file-handling, but we're going to pass an entire book to Python.