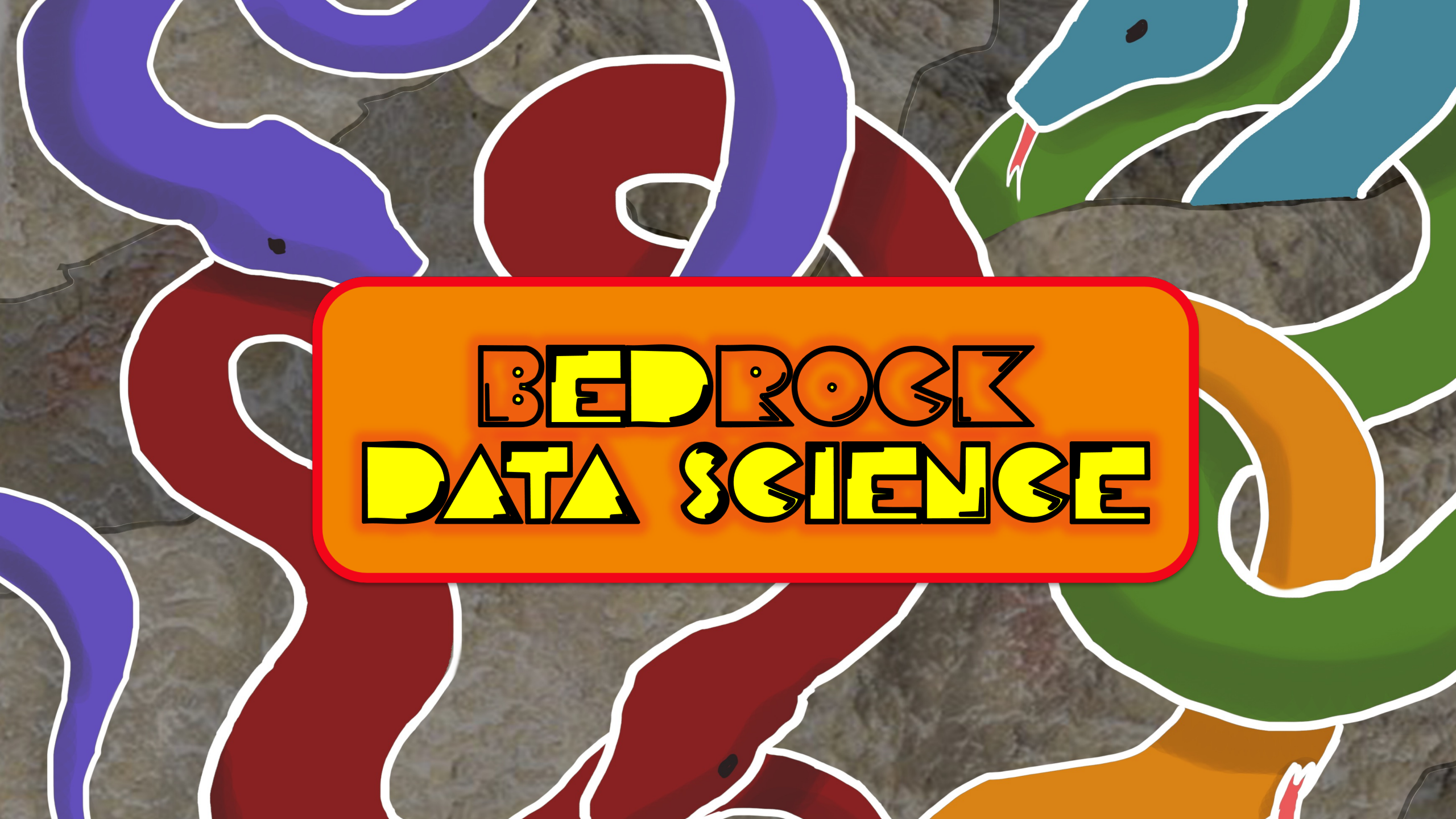

BEDROCK
DATA SCIENCE

Session 2: Data structures

What is numpy ?

What is with ?

What is a dictionary ?

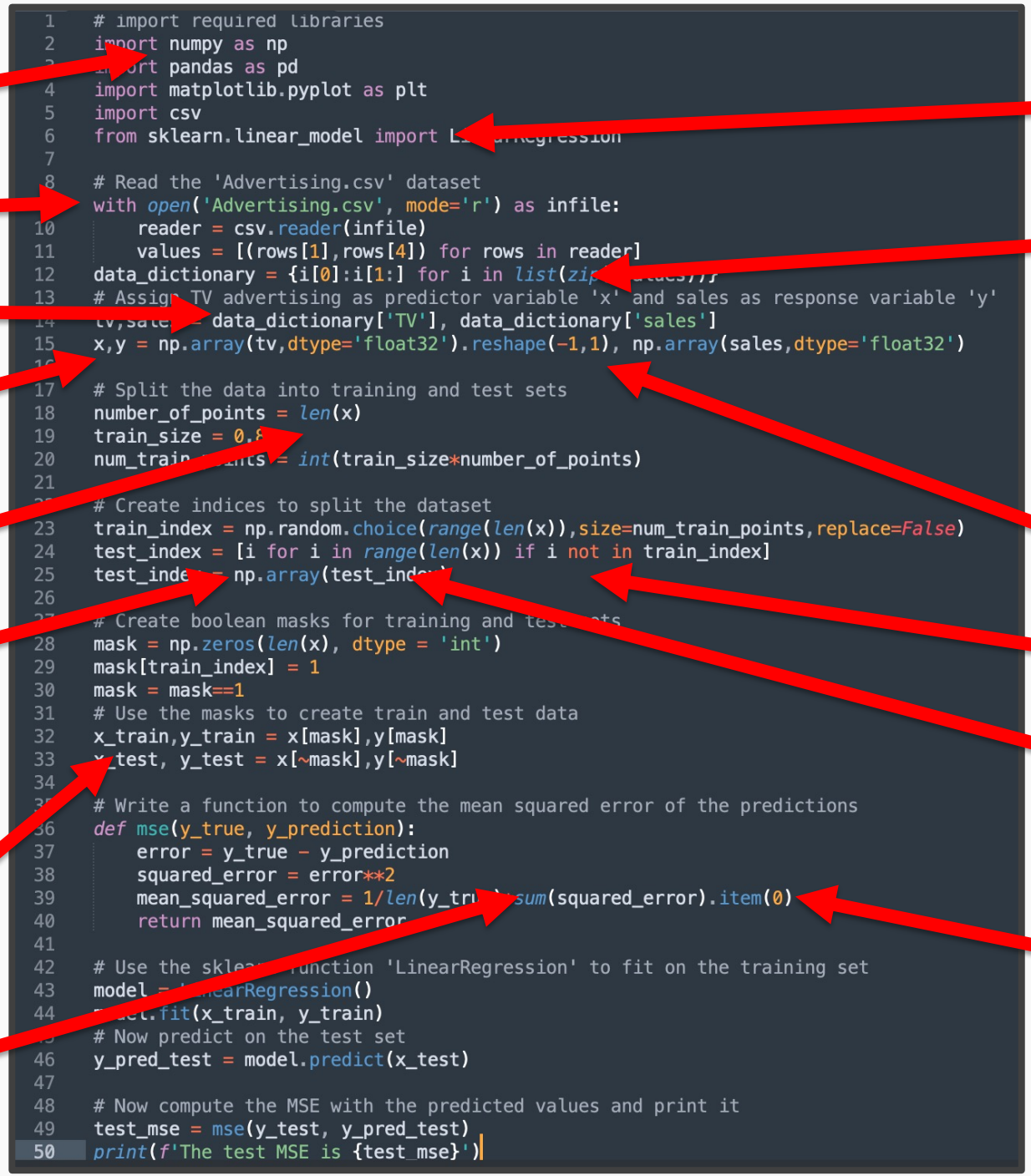
What is a tuple ?

What is len() ?

What is list comprehension ?

What is train & test set ?

What is sum ?



What is import ?

What is zip ?

What is the * operator ?

What is a conditional list comprehension ?

What is range() ?

What is item(0) ?

```
8   # Read the 'Advertising.csv' dataset
9   with open('Advertising.csv', mode='r') as infile:
10      reader = csv.reader(infile)
11      values = [(rows[1],rows[4]) for rows in reader]
12  data_dictionary = {i[0]:i[1:] for i in list(zip(*values))}
13  # Assign TV advertising as predictor variable 'x' and sales as response variable 'y'
14  tv,sales = data_dictionary['TV'], data_dictionary['sales']
15  x,y = np.array(tv,dtype='float32').reshape(-1,1), np.array(sales,dtype='float32')
16
17  # Split the data into training and test sets
18  number_of_points = len(x)
19  train_size = 0.8
20  num_train_points = int(train_size*number_of_points)
21
22  # Create indices to split the dataset
23  train_index = np.random.choice(range(len(x)),size=num_train_points,replace=False)
24  test_index = [i for i in range(len(x)) if i not in train_index]
25  test_index = np.array(test_index)
26
```

What is a tuple ?

What is a dictionary ?

What is list comprehension ?

What is the zip operator ?

What is the * operator ?

Outline

- Data Structures
  - Sets
  - Tuples
  - Dictionaries
- Pythonic Syntax
  - List comprehension
  - Set comprehension
  - Dictionary comprehension
- Special built-in functions
  - Enumerate
  - Zip

Data Structures

Why do we need data structures?

In a land far away, where books were abundant...

The Grand Keeper of Books

Oh no, how am I going to go through this chaos?

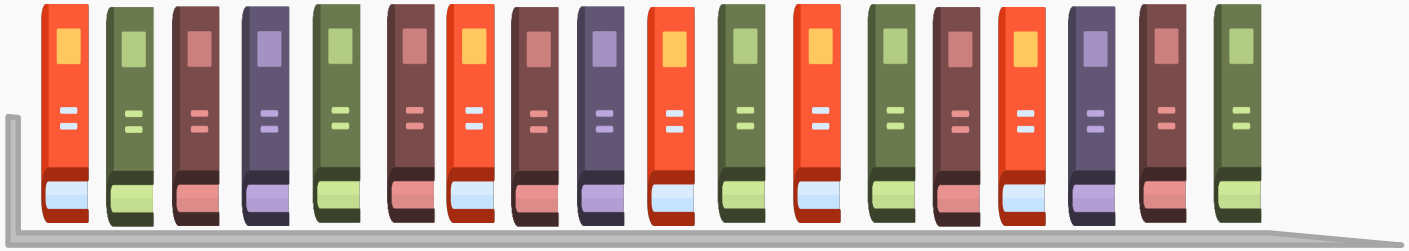
Uh...your Highness, of course. Just give me a moment.

The Great keeper, I need the book "Become a Data Scientist in 30 days"

Why do we need data structures?

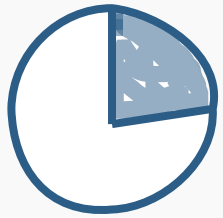
The Grand Keeper of Books

Like a List



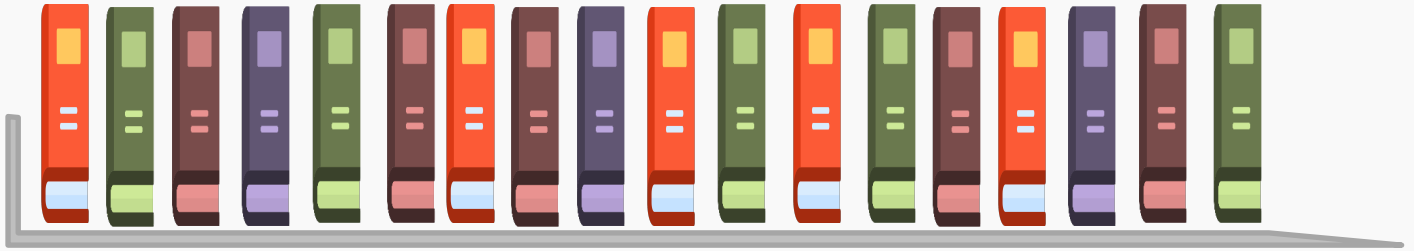
Why do we need data structures?

Searching for the book



Time it took

Like a List

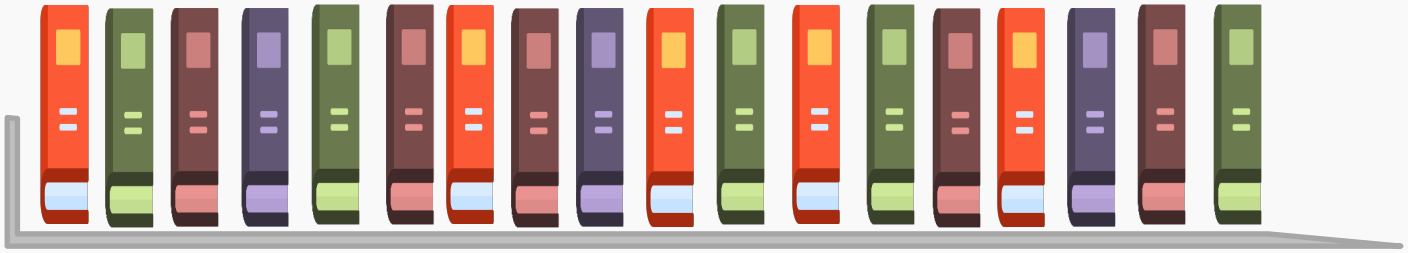


Why do we need data structures?

The Grand Keeper of Books

Here you go, your Highness!

Like a List

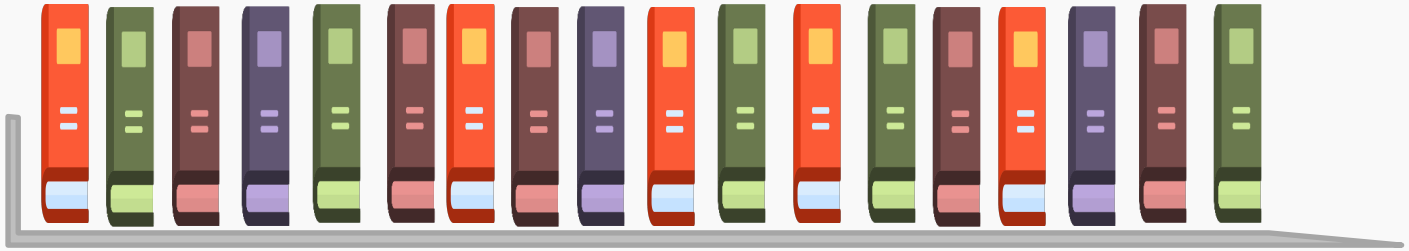


I need another book "Zen of Python" – could you please get it for me?

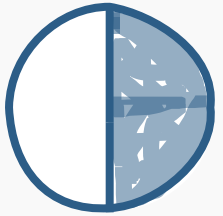
Why do we need data structures?

The Grand Keeper of Books

Like a List



Why do we need data structures?

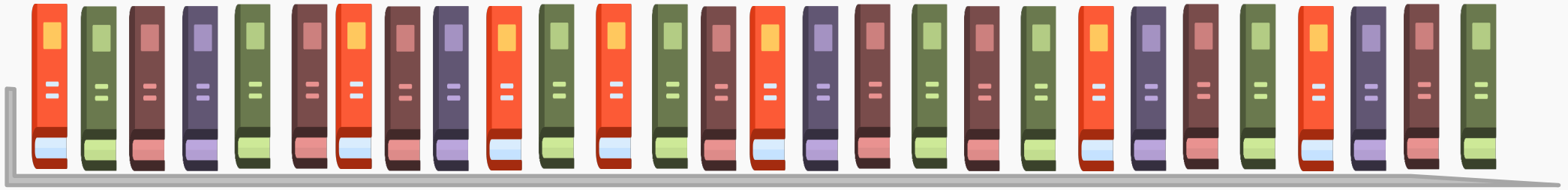


Time it took

Like a List

Why do we need data structures?

Like a List



The Grand Keeper of Books

Here you go, your Highness!

Thank you once again!

...although, respected Keeper, that was slow. Father is sending 10,000 more books your way!

Why do we need data structures?

10,0000 books!!!? How am I going to arrange it so that I can fetch these books fast & also give information about it?

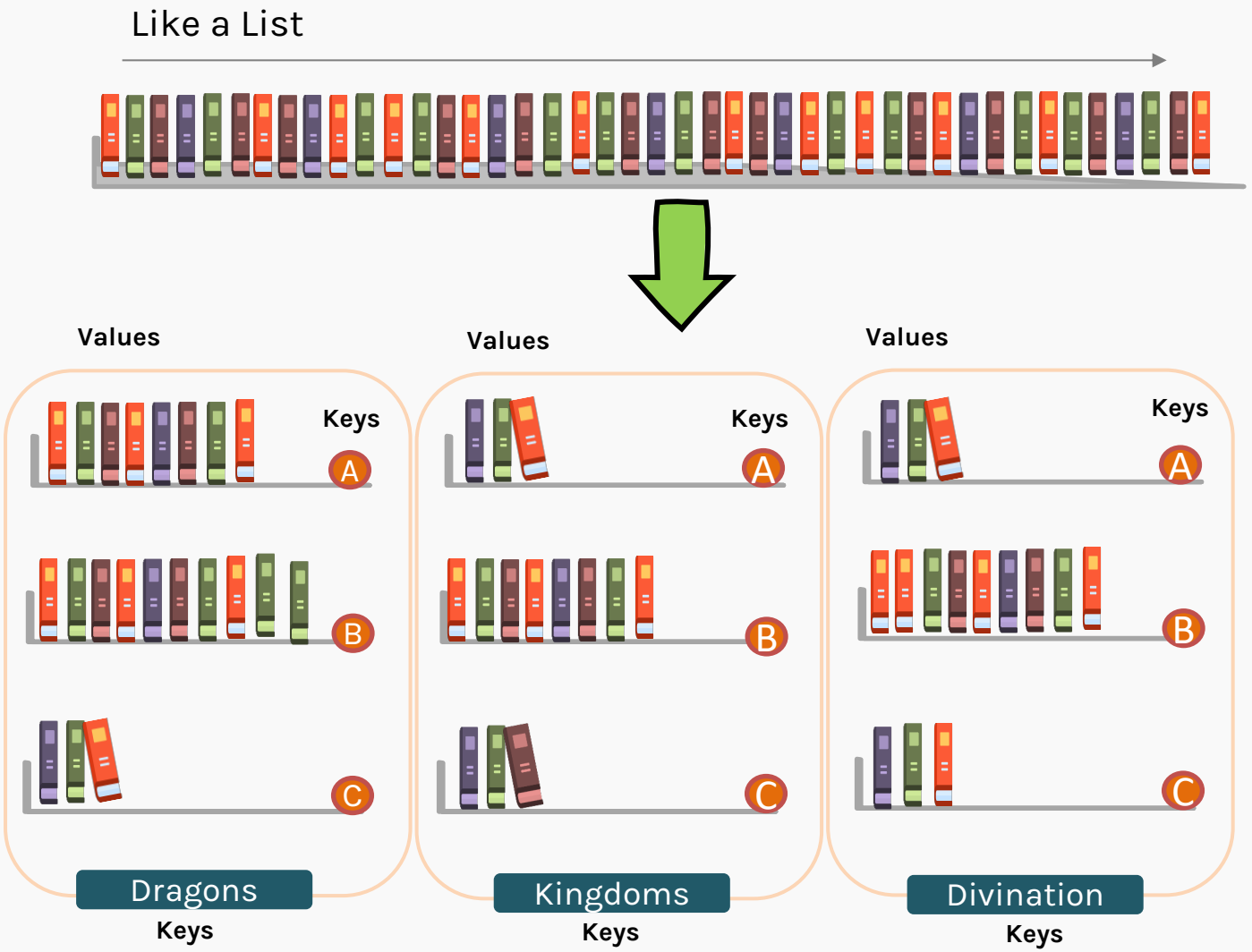
The Grand Keeper of Books

Like a List

Why do we need data structures?

The Grand Keeper of Books

Let me build a library instead



This process is called dictionaries in Python.

Dictionaries in Python – you will use them a lot!

In Python, dictionaries are **ordered** collection of **mutable objects with immutable keys.**
Elements within a dictionary have a key & value.

$$Dict = \{Key_1:Value_{1,\ldots,} Key_n:Value_n\}$$

```
>>> dictA = {'India':'New Delhi', 'USA':'Washington
DC','Germany':'Berlin','Sri Lanka':'Colombo’}

>>> dictB = {'Apples':1, 'Pineapple':4, 'Grapes’:3}

>>> print(dictA)
{'India': 'New Delhi', 'USA': 'Washington DC', 'Germany': 'Berlin',
'Sri Lanka': 'Colombo’}

>>> print(dictB)
{'Apples': 1, 'Pineapple': 4, 'Grapes': 3}
```

DictA

Key - Type	Value - Type
India<str>	New Delhi<str>
USA<str>	Washington DC<str>
Germany<str>	Berlin<str>

DictB

Key - Type	Value - Type
Apples<str>	1 <int>
Pineapple<str>	4 <int>
Grapes <str>	3 <int>

Dictionaries in Python

Important Dictionary Methods to Remember

Python Code	Function
`dictA = {'India':'Delhi','USA':'Washington'},` `dictA = {name='Sam','age=23}`	Creates a dictionary with the key value pair provided

Dictionaries in Python

Important Dictionary Methods to Remember

Python Code	Function
`dictA = {'India':'Delhi','USA':'Washington'},` `dictA = {name='Sam','age=23}`	Creates a dictionary with the key-value pair provided
`dictA.items()`	Gets a list of the key-value pairs in the dictionary

Dictionaries in Python

Important Dictionary Methods to Remember

Python Code	Function
`dictA = {'India':'Delhi','USA':'Washington'},` `dictA = {name='Sam','age=23}`	Creates a dictionary with the key-value pair provided
`dictA.items()`	Gets a list of the key-value pairs in the dictionary
`dictA.values()`	Gets a list of the values in the dictionary

Dictionaries in Python

Important Dictionary Methods to Remember

Python Code	Function
`dictA = {'India':'Delhi','USA':'Washington'},` `dictA = {name='Sam','age=23}`	Creates a dictionary with the key-value pair provided
`dictA.items()`	Gets a list of the key-value pairs in the dictionary
`dictA.values()`	Gets a list of the values in the dictionary
`dictA.keys()`	Gets a list of the keys in the dictionary

Dictionaries in Python

Important Dictionary Methods to Remember

Python Code	Function
`dictA = {'India':'Delhi','USA':'Washington'},` `dictA = {name='Sam','age=23}`	Creates a dictionary with the key value pair provided
`dictA.items()`	Gets a list of the key-value pairs in the dictionary
`dictA.values()`	Gets a list of the values in the dictionary
`dictA.keys()`	Gets a list of the keys in the dictionary
`dictA['place of birth'] = 'Singapore'`	Sets the value of the key associated with it
`del dictA['age']`	Deletes the key-value pair from the dictionary

Tuples in Python

In Python, tuples are **ordered** collections of **immutable objects. They are like lists but the difference is that they are immutable and enclosed with ( ) instead of [ ].**

$$\text{Tuple} = (e_1, e_2 \ldots e_N)$$

```
#Intialize a tuple
In [2]: tupleA = ('India','USA','Germany')

#Intialize another tuple
In [3]: tupleB = ('violet','indigo','blue','yellow','orange','red')

In [5]: tupleA + tupleB
Out[5]:
('India','USA','Germany','violet','indigo','blue','yellow','orange','red')
```

Program execution is faster when manipulating a tuple than it is for the equivalent list.

Tuples in Python

Important to remember – Since parentheses are also used to define operator precedence in expressions, Python evaluates a number within ( ) as an object of its tuple. But if you want to tell Python to define a singleton tuple, you need to put a comma.

```
In [7]: a = (2,3)
In [8]: b = (2)
In [9]: c = ('string')
In [10]: d = (2,)
In [11]: e,f = 2,3
```

Type of the Variable

Type(a) - tuple

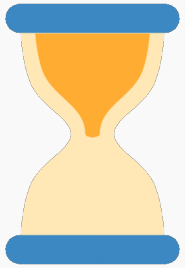
Type(b) - integer

Type(c) - string

Type(d) - tuple

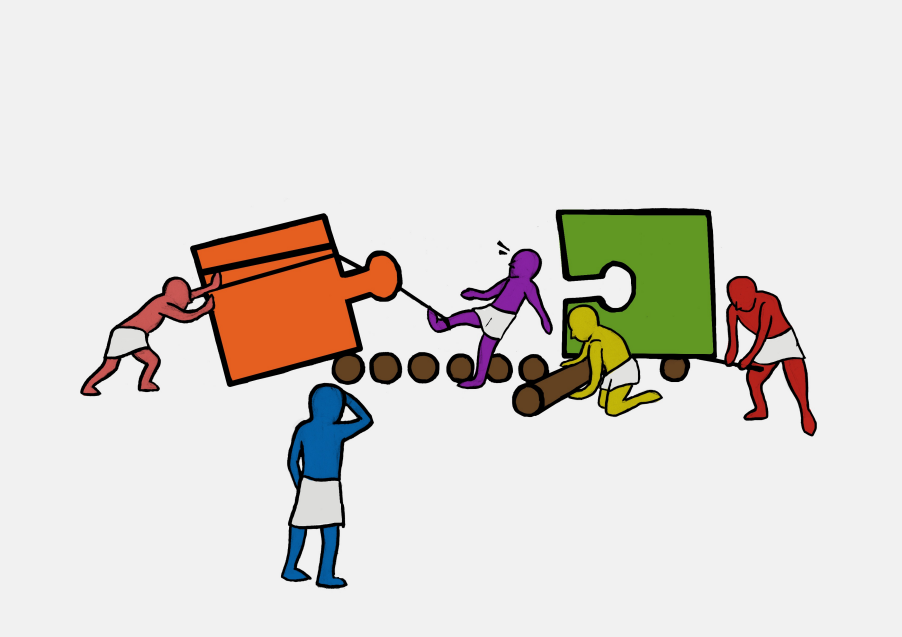
Type(e) - integer

Type(f) – integer

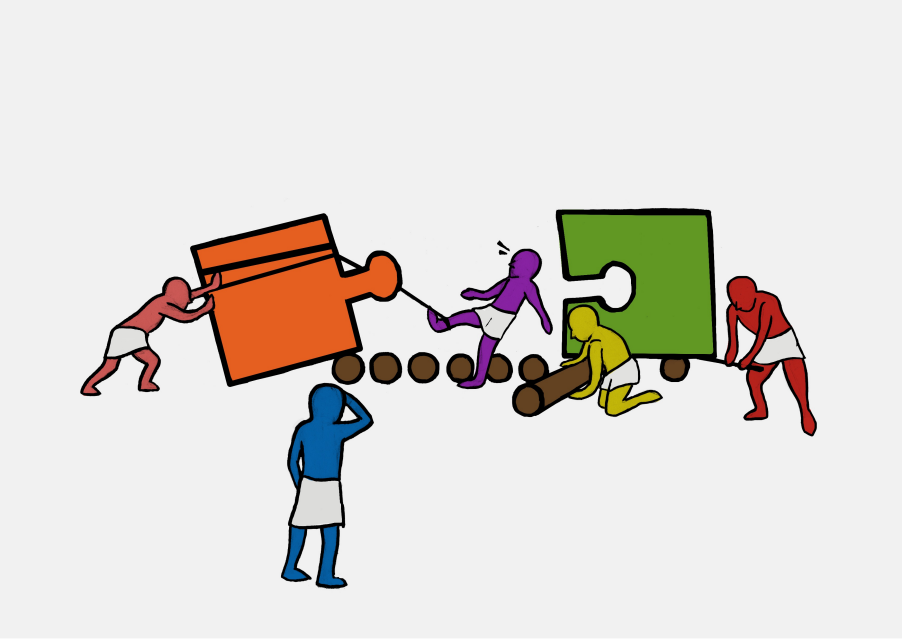


Digestion Time

Exercise



Exercise #3



Description

🏋️‍♀️ Exercise: The Book Keeper's Apprentice - Dicts & List

Congratulations on becoming the book keeper's apprentice! The highness has once again come to the library asking for a book. Unfortunately, the great keeper is not in today and the new books aren't arranged yet. You need to do all the work.

The aim of this exercise is to understand the implications of using a dictionary over a list and getting comfortable creating dictionaries.

Instructions

Import the necessary libraries and the data required for this exercise that is stored in `books_dict`, `authors_list`, `books_list`.