Python fundamentals

Welcome

Dictionary

Damilola Omifare
Data Scientist & Developer

keenarc

Scope & Objective

This session is designed to imparting a basic level understanding of how **dictionaries** work, how to use **dictionaries** in python programming.

Contents

- Lamba Recap.
- Quick look at dictionaries.

Quick Recap on lambda

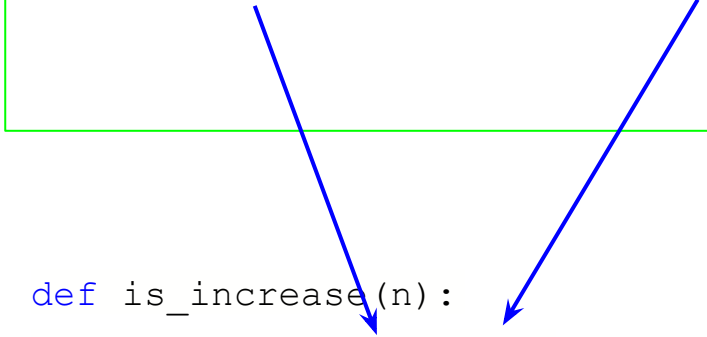
A lambda function → small anonymous **function**.
It can take any number of arguments, but syntactically restricted to one expression

Why use it → if you need have an anonymous **function** inside another **function** for a short period of time.

keenarc

Quick Recap on lambda

```
lambda <arguments>: <expression>
```



```
def is_increase(n):
 return lambda x: x + n

to_increase = is_increase( 12)
print(to_increase(5))
```

keenarc

Dictionaries

is a data type in python which is unordered, changeable and indexed with **no duplicate members.**

dictionaries are indexed by keys (***which can be any immutable type. strings*** *and* ***numbers*** *can always be keys.* **tuple** can be used as key)

think of a dictionary as a set of key: value pairs, with the requirement that the keys are unique (within one dictionary).

The main operations on a **dictionary** are storing a value with some key and extracting the value given the key.

keenarc

Dictionaries

An empty dictionary is initiate with **{}**

```
house =  {
 "living_room": 5,
 "name": "Newton Close",
 "year_built": 2020
}
print(house)
```

More on Dictionaries

access the **items** of a dictionary by referring to its **key** name, inside **square brackets**.

```
house =  {
 "living_room": 5,
 "name": "Newton Close",
 "year_built": 2020
}
print(house)
print(house["living_room"])
```

You can also use a **get()** method to achieve this.

More on Dictionaries

They are various methods and operations you can call on a **dictionary:**

- **pop** *If key is in the dictionary, remove it and return its value,*
- **len** *Return the number of items in the dictionary*
- **clear()** *Remove all items from the dictionary.*
- **copy()** *Return a shallow copy of the dictionary*
- **get()** *Return the value for key if key is in the dictionary*
- **items** *Return a new view of the dictionary's items*

Read more here :
https://docs.python.org/3/library/stdtypes.html#typesmapping

keenarc

dict()

The new dictionary can be created using the **dict()** constructor

```
student = dict([('name', "Chrome"), ('age', 18), ('address',
"Python avenue")])
print(student)

car = dict(brand="Tesla", model="X Series", year=2019)
print(car)
```

Thank you for your time