LESSON 8

main.py

```
class BankAccount:
  def __init__(self,
      owner):
    self.owner = owner
```

Py

Classes & Objects

Learn it all @MyWebUniversity



A FREE MYWEBUNIVERSITY COURSE

PYTHON SCRIPTING & OOP

LESSON 8

OOP: Classes and Objects

class, __init__, self, attributes, methods and __str__

Why Classes?



- As programs grow, many functions pass the same data around. A class bundles that data with …
- The class is the blueprint, and each object is one real thing built from it, like one bank …

01

Defining a Class

account.py

```python
class BankAccount:
    def __init__(self, owner, balance=0):
        self.owner = owner
        self.balance = balance

    def deposit(self, amount):
        self.balance += amount
        return self.balance
```

- Write class and a capitalized name. The special method double underscore init runs when a …
- self is the object itself. Every method receives it first, and self dot balance means this …

02

Creating and Using Objects

use.py

```
a = BankAccount("Mariam", 500)
b = BankAccount("Ahmad")
a.deposit(250)                  # 750
b.deposit(100)                  # 100
print(a.owner, a.balance)       # Mariam 750
print(isinstance(a, BankAccount))   # True
```

- Call the class like a function to create an object. Each object keeps its own attribute …
- Call methods with a dot. Python passes the object as self for you, so you never write it in …

03

__str__ and __repr__

str.py

```
class BankAccount:
    ...
    def __str__(self):
        return f"{self.owner}: {self.balance:,} AFN"
    def __repr__(self):
        return f"BankAccount({self.owner!r}, {self.balance})"

print(a)        # Mariam: 750 AFN
```

- Printing an object shows something ugly like a memory address. Define double underscore str ...
- double underscore repr returns a developer view, used in the REPL and in lists. These ...

04

Class Attributes and Class Methods

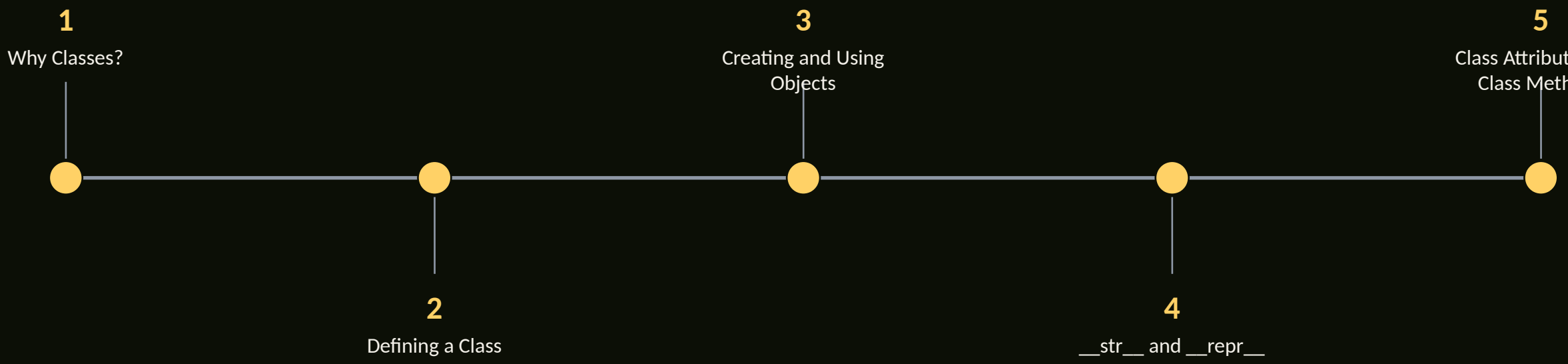
classattr.py

```python
class BankAccount:
    interest_rate = 0.03          # shared by all accounts
    count = 0
    def __init__(self, owner, balance=0):
        self.owner, self.balance = owner, balance
        BankAccount.count += 1
    @classmethod
    def from_csv(cls, row):
        return cls(row["owner"], float(row["balance"]))
```

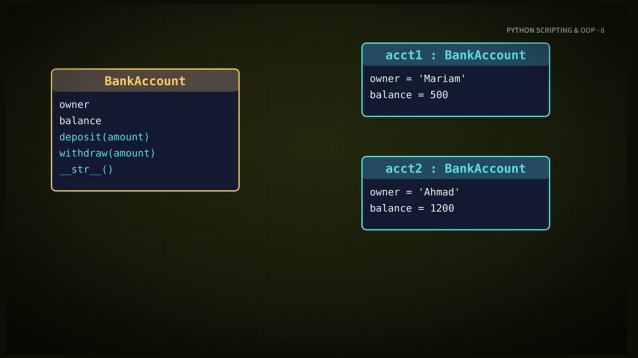
- Attributes set on the class itself are shared by all objects, such as an interest rate or a …
- A classmethod receives the class instead of an object, and is often used as an alternative …

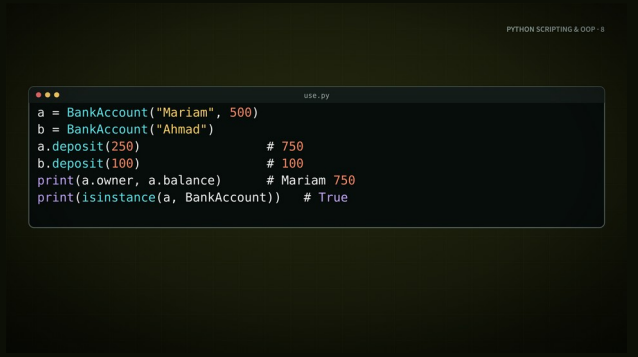
05

Lesson Map


1
Why Classes?
2
Defining a Class
3
Creating and Using Objects
4
__str__ and __repr__
5
Class Attributes and Class Methods

Sources and Further Reading





Thank you for watching

MyWebUniversity.com

Free technical education since 2000 — English and Dari

@MyWebUniversity · @MyWebUniversity-Dari