



UC Berkeley EECS
Lecturer
Michael Ball

Computational Structures in Data Science



Object-Oriented Programming



Example: Account

```
class BaseAccount:
    def __init__(self, name, initial_deposit):
        self.name = name
        self.balance = initial_deposit
    def account_name(self):
        return self.name
    def account_balance(self):
        return self.balance
    def withdraw(self, amount):
        self.balance -= amount
        return self.balance
```

new namespace

attributes

The object

dot

methods



Creating an object, invoking a method

The Class Constructor

```
my_acct = BaseAccount("John Doe", 93)  
my_acct.withdraw(42)
```

dot



Special Initialization Method

```
class BaseAccount:

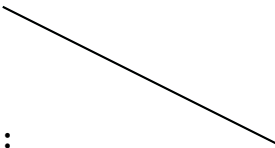
    def __init__(self, name, initial_deposit):
        self.name = name
        self.balance = initial_deposit

    def account_name(self):
        return self.name

    def account_balance(self):
        return self.balance

    def withdraw(self, amount):
        self.balance -= amount
        return self.balance
```

`return None`





More on Attributes

- Attributes of an object accessible with ‘dot’ notation
`obj.attr`
- You can distinguish between “public” and “private” data.
 - Used to clarify to programmers how you class should be used.
 - In Python an `_` prefix means “this thing is private”
 - `_foo` and `__foo` do different things inside a class.
 - [More for the curious.](#)
- Class variables vs Instance variables:
 - Class variable set for all instances at once
 - Instance variables per instance value

Example



```
class BaseAccount:

    def __init__(self, name, initial_deposit):
        self.name = name
        self.balance = initial_deposit

    def name(self):
        return self.name

    def balance(self):
        return self.balance

    def withdraw(self, amount):
        self.balance -= amount
        return self.balance
```



Example: “private” attributes

```
class BaseAccount:

    def __init__(self, name, initial_deposit):
        self._name = name
        self._balance = initial_deposit

    def name(self):
        return self._name

    def balance(self):
        return self._balance

    def withdraw(self, amount):
        self._balance -= amount
        return self._balance
```



Example: class attribute

```
class BaseAccount:
    account_number_seed = 1000

    def __init__(self, name, initial_deposit):
        self._name = name
        self._balance = initial_deposit
        self._acct_no = BaseAccount.account_number_seed
        BaseAccount.account_number_seed += 1
    def name(self):
        return self._name

    def balance(self):
        return self._balance

    def withdraw(self, amount):
        self._balance -= amount
        return self._balance
```




More class attributes

```
class BaseAccount:
    account_number_seed = 1000
    accounts = []
    def __init__(self, name, initial_deposit):
        self._name = name
        self._balance = initial_deposit
        self._acct_no = BaseAccount.account_number_seed
        BaseAccount.account_number_seed += 1
        BaseAccount.accounts.append(self)

    def name(self):
        ...

    def show_accounts():
        for account in BaseAccount.accounts:
            print(account.name(),
                  account.account_no(), account.balance())
```



UC Berkeley EECS
Lecturer
Michael Ball

Computational Structures in Data Science



Midterm Review



Announcements & Policies

- Midterm:
 - 2 hours, 120 Minutes
 - Unlimited Handwritten Cheat sheets
 - 1 CS88 Provided Reference Sheet
 - Remote Exams:
 - » Zoom Proctoring https://docs.google.com/document/d/18Zkd-PP5qEa3hQVRhKvU9_6CmL9WVPsQMScu65D1tuI/edit?usp=sharing
 - Verify Scheduling / Accommodations
 - » <https://edstem.org/us/courses/8883/discussion/750341>



Learning Objectives

- Midterm Review
 - Let's talk general questions
 - Then we will find specific problems to go over.



My Advice

- Don't rush!
- Skim the exam first
 - It's ok to do questions out of order!
 - Get the stuff you're good without out of the way
 - BUT don't spend too much time planning the exam.
- Read through the question once
 - What's it asking you to do at a high level?
 - What do the doctests suggest?
 - What techniques should you be using?