

Instantiating Custom Classes



Mateo Prigl
Software Developer

Designing a Class

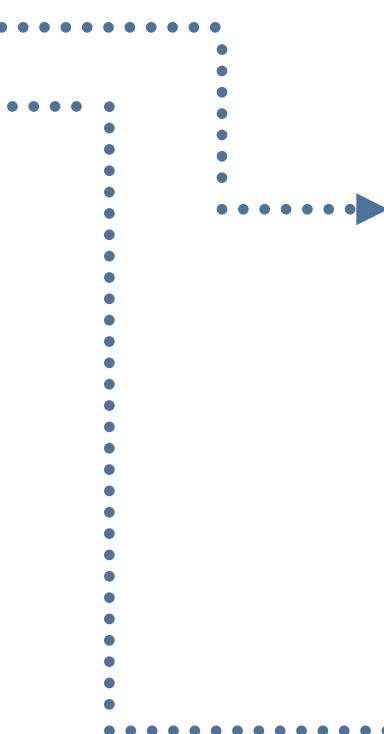
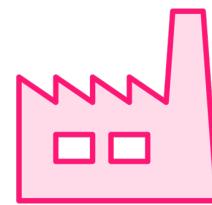
class Employee

Object attributes

name age
position salary

Functions

increase_salary()
info()



employee1

name:	"Ji-Soo"
age:	38
position:	"developer"
salary:	1200

Employee class object

employee2

name:	"Lauren"
age:	44
position:	"tester"
salary:	1000

Employee class object

```
employee1 = Employee("Ji-Soo", 38, "developer", 1200)  
employee2 = Employee("Lauren", 44, "tester", 1000)
```



Class Functions for Constructing an Object

`__new__`

**Allocate memory for a new object
and send it to the `__init__`
function**

`__init__`

**Receive a new object from the
`__new__` function as a "self"
parameter**



Designing a Class

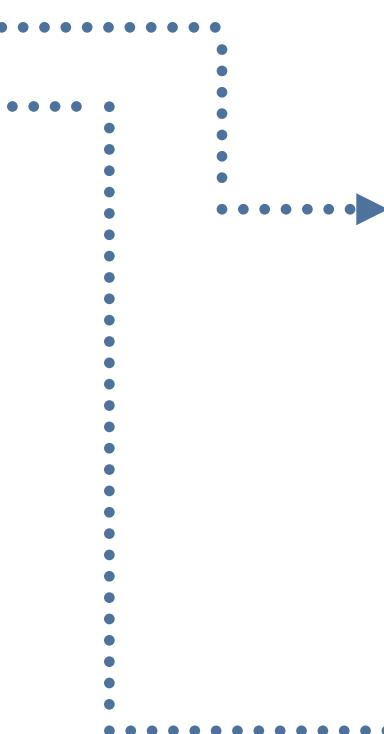
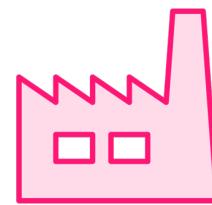
class Employee

Object attributes

name age
position salary

Functions

increase_salary()
info()



employee1

name:	"Ji-Soo"
age:	38
position:	"developer"
salary:	1200

Employee class object

employee2

name:	"Lauren"
age:	44
position:	"tester"
salary:	1000

Employee class object

```
employee1 = Employee("Ji-Soo", 38, "developer", 1200)  
employee2 = Employee("Lauren", 44, "tester", 1000)
```



Designing a Class

class Employee

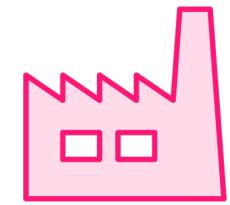
Instance attributes

name age
position salary

Functions

increase_salary()

info()



employee1

name:	"Ji-Soo"
age:	38
position:	"developer"
salary:	1200

Employee class instance

employee2

name:	"Lauren"
age:	44
position:	"tester"
salary:	1000

Employee class instance

```
employee1 = Employee("Ji-Soo", 38, "developer", 1200)
```

```
employee2 = Employee("Lauren", 44, "tester", 1000)
```



Designing a Class

class Employee

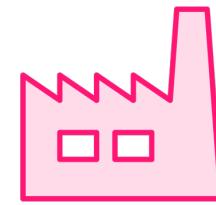
Instance attributes

name age
position salary

Functions

increase_salary()

info()



employee1

name:	"Ji-Soo"
age:	38
position:	"developer"
salary:	1200

Employee class instance

employee2

name:	"Lauren"
age:	44
position:	"tester"
salary:	1000

Employee class instance

```
employee1 = Employee("Ji-Soo", 38, "developer", 1200)
```

```
employee2 = Employee("Lauren", 44, "tester", 1000)
```



Designing a Class

class Employee

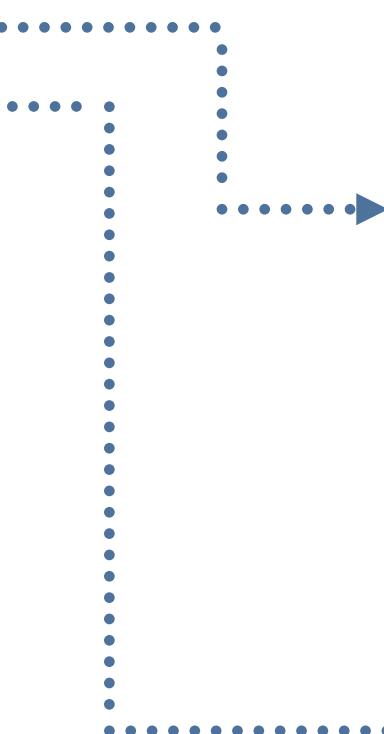
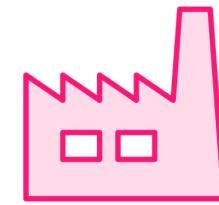
Instance attributes

name	age
position	salary

Instance methods

increase_salary()

info()



employee1

name:	"Ji-Soo"
age:	38
position:	"developer"
salary:	1200

Employee class instance

employee2

name:	"Lauren"
age:	44
position:	"tester"
salary:	1000

Employee class instance

```
employee1 = Employee("Ji-Soo", 38, "developer", 1200)
```

```
employee2 = Employee("Lauren", 44, "tester", 1000)
```



Dunder Methods

**Special "magic" methods that start and end with the double underscore.
Usually invoked by a special syntax.**



Summary

Up Next:

Managing Attribute Access

