

Python Lab 4: Objects

If you get confused, use these resources available to you:

1. Sample code on the next page
2. Lecture slides (<http://aiti.mit.edu/materials/ghana-summer-2013/>)
3. Previous labs
4. Other students
5. Python documentation (<http://docs.python.org/2/library/index.html> and <http://docs.python.org/3/library/index.html>)
6. Google
7. Instructors

Exercise 1: Review

Write a function that takes in an integer and returns True if the integer is even and False if the integer is odd.

Exercise 2: Objects

Let's pretend you are making a computer game about car racing. You want to track where each racecar is and how fast each one is going. Represent each car as an object. Create a new program called `car.py` to hold this code. **You can use the code below as a starting point.**

Car Object starting code:

```
# A class to represent a car
class Car:
    def __init__(self, start_pos, speed):

    def drive(self, time, direction):

    def printPosition(self):
```

Add the following code when you have finished to test your Car code

```
myCar = Car(2, 3)
myCar.printPosition()
myCar.drive(3, 'forward')
myCar.printPosition()
myCar.drive(2, 'backward')
myCar.printPosition()
```

Exercises:

1. Implement the constructor (the `__init__` method). The constructor takes in two arguments: a starting position and speed. Both arguments are integers. Set the instantiated car's starting position and speed to these arguments.

2. Implement the `drive` method, which takes in two arguments: `time`, which is a positive number, and `direction`, which is either the string 'forward' or 'backward'. The `drive` method changes the car's position by `time` multiplied by the car's speed. If `direction` is 'forward', then increase the position by this amount. If the `direction` is 'backward', then decrease the position by this amount. For example, if the car's speed is 4, and it drives forward for 2 units of time, its position should increase by 8.

3.) Implement the `printPosition` method. This method should be print out 'This car is currently at position <pos>' where <pos> is the car's position. For example, if the car is located at position 2, then calling `printPosition()` should print:
`This car is currently located at position 2.`

4.) Type in the code at the end of the template and run it. Be sure to examine what the code is doing. Explain the code to an instructor. If you implemented the `Car` class correctly, then your code should output:
`The car is currently at position 2`
`The car is currently at position 11`
`The car is currently at position 5`

Exercise 3: Dictionaries

Let's say you're running a store and want to know the price of each object. Create a dictionary to represent this information. The keys of the dictionary should be the food names and the values should be prices.

1. What variable type (`int`, `float`, `String`, `Boolean`) should the food names be? What variable type should the prices be?
2. Create a dictionary to represent this information:
`Chicken costs 4.50`
`Rice costs 1.50`
3. Add 3 more foods and prices to the dictionary
4. Update the price of chicken to 5.00
5. Write a line of code to find the price of chicken plus the price of rice.

Sample code from lecture:

Dictionaries

Create a dictionary:

```
ages = { }
```

Add items to dictionary:

```
ages["Leah"] = 22
```

```
ages["Taibo"] = 21
```

Change values in dictionary:

```
ages["Leah"] = 23
```

Print values from dictionary:

```
ages["Leah"]
```

```
ages["Leah"] + ages["Taibo"]
```

Objects

```
class Car():
```

```
    def __init__(self, color):
```

```
        self.wheels = 4
```

```
        self.color = color
```

```
    def lose_a_wheel(self):
```

```
        self.wheels = self.wheels - 1
```

```
    def lose_wheels(self, num_wheels_lost):
```

```
        self.wheels = self.wheels - num_wheels_lost
```

```
    def printColor(self):
```

```
        print "My color is", self.color
```

```
myCar = Car("green")
```

```
myCar.printColor()
```

```
myCar.lose_a_wheel()
```

```
myCar.lose_wheels(3)
```

```
myCar.wheels
```