

Python Programming

Classes Homework 4

Mostafa S. Ibrahim

Teaching, Training and Coaching since more than a decade!

Artificial Intelligence & Computer Vision Researcher

PhD from Simon Fraser University - Canada

Bachelor / Msc from Cairo University - Egypt

Ex-(Software Engineer / ICPC World Finalist)



Problem #3: AutoTrader Class - Code Review

- The following code is working properly in **production**
- Find all design issues and fix them!

```
class CarSpecs:
    def __init__(self):
        self.trim = None
        self.engine_type = None
        self.horsepower = None
        self.steering_ratio = None
        # Expected more to be added in future
```

```
class AutoTrader:
    def __init__(self):...

    def load_database(self):...

    def get_matches(self, car_specs):...
```

```
42
43 ▶ if __name__ == '__main__':
44     trader = AutoTrader()
45     trader.load_database()
46
47     query = CarSpecs()
48     query.engine_type = 'EG12121'
49
50     ans = trader.get_matches(query)
51     print(len(ans)) # 2
52
53     query.horsepower = 10
54     ans = trader.get_matches(query)
55     print(len(ans)) # 1
```

Problem #3: AutoTrader Class - Code Review

```
11 class AutoTrader:
12     def __init__(self):
13         self.db_cars_specs = []
14
15     def load_database(self):
16         car1 = CarSpecs()
17         car1.engine_type = 'EG12121'
18         car1.horsepower = 10
19         self.db_cars_specs.append(car1)
20
21         car2 = CarSpecs()
22         car2.engine_type = 'EG12121'
23         car2.horsepower = 12
24         self.db_cars_specs.append(car2)
25         # Load More
26
27     def get_matches(self, car_specs):...
```

Problem #3: AutoTrader Class - Code Review

```
11 class AutoTrader:
12     def __init__(self):...
14
15     def load_database(self):...
16
17
18
19
20
21
22
23
24
25
26
27     def get_matches(self, car_specs):
28         found = []
29         for db_car in self.db_cars_specs:
30             if car_specs.trim is not None and car_specs.trim != db_car.trim:
31                 continue
32             if car_specs.engine_type is not None and car_specs.engine_type != db_car.engine_type:
33                 continue
34             if car_specs.horsepower is not None and car_specs.horsepower != db_car.horsepower:
35                 continue
36             if car_specs.steering_ratio is not None and car_specs.steering_ratio != db_car.steering_ratio:
37                 continue
38             found.append(db_car)
39
40         return found
```

Homework #: Configuration Manager

- Your team lead asked to develop a **configuration manager** component
 - This one loads system default values and critical information
 - E.g. Some servers IPs, Databases paths (& usernames & passwords)
 - E.g. also loads some **heavy** files and cache for frequent requests
- Many teams used it. The complaint: with every construction, it takes a 30 min
 - But all are same information! So why the duplicating same processing
 - Fix the issue, but you must maintain the old expected calls for the class
 - Specifically, only the first object should load the data, but others used same loaded data
 - Don't use global variables!

Homework #: Configuration Manager: Before

```
class ConfigurationManger:
    def __init__(self, configuration_path):
        self.configuration_path = configuration_path
        # Other attributes
        self._load()

    def _load(self):
        # takes 30 minutes to load data
        print('Loading the Configuration')
        self.servers_ips = ["10.20.30.40",
                            "10.20.30.41", "10.20.30.42"]
        self.aws_service_url = "amazon-aws.com"
        # load heavy data
        import time
        time.sleep(1)
```

```
def f1():
    mgr = ConfigurationManger('disk/config.json')
    print(mgr.configuration_path)

def f2():
    mgr = ConfigurationManger('disk/config.json')
    print(mgr.configuration_path)

def f3():
    mgr = ConfigurationManger('disk/config.json')
    print(mgr.configuration_path)

if __name__ == '__main__':
    f1()
    f1()
    f1()
    f2()
    f3()
```

```
/home/moustara/system-inst
Loading the Configuration
disk/config.json
Loading the Configuration
disk/config.json
Loading the Configuration
disk/config.json
Loading the Configuration
disk/config.json
Loading the Configuration
disk/config.json
```

Homework #: Configuration Manager: After

```
def f1():  
    mgr = ConfigurationManger('disk/config.json')  
    print(mgr.configuration_path)  
  
def f2():  
    mgr = ConfigurationManger('disk/config.json')  
    print(mgr.configuration_path)  
  
def f3():  
    mgr = ConfigurationManger('disk/config.json')  
    print(mgr.configuration_path)  
  
if __name__ == '__main__':  
    f1()  
    f1()  
    f1()  
    f2()  
    f3()
```

Loading the Configuration
disk/config.json
disk/config.json
disk/config.json
disk/config.json
disk/config.json

“Acquire knowledge and impart it to the people.”

“Seek knowledge from the Cradle to the Grave.”