PYTHON CLASSES AND OBJECTS

By HadoopExam Learning Resources

Note: These instructions should be used with the HadoopExam Apache OOzie: Professional Trainings. Where it is executed and you can do hands on with trainer.

Content:

- Class : User Defined Types
- Class Attributes
- Passing object to a function
- Embedded object
- Mutable objects
- Deep Copy
- Shallow copy

User Defined Types: We will be defining our own types.

Let's define a type named TrainingCourse, which has three variables as below

- name
- price
- publisher

To create a new user defined variable, you have to use **class** keyword. As below

```
class TrainingCourse(object):
        "This User Defined variable represents the HadoopExam Training Courses"
```

- TrainingCourse is a name of type, which represent object type. Object is a built in datatype.
- We can define functions and variables in a class
- Get the detail of the TrainingCourse class as below

```
print TrainingCourse
<class '__main__.TrainingCourse'>
```

- It means TrainingCourse is a top level class, hence its full name is __main__.TrainingCourse
- Class object is similar to factory to create different objects.
- Create a new course without any detail as below.

```
hadoop = TrainingCourse() #Creating new object, known as instantiation.
print hadoop #hadoop is an instance of TrainingCourse class
```

Attributes in a Class:

- Here, name, price and publisher are attributes of class TrainingCourse object. And hadoop is a variable, pointing to a TrainingCourse object.

```
hadoop = TrainingCourse()
hadoop.name= "Hadoop Professional Training"
hadoop.price=3900
hadoop.publisher="HadoopExam.com"
print ("Course %s has Price %d INR and its publuisher is %s "
%(hadoop.name,hadoop.price,hadoop.publisher ) )
```

Passing Class Object as an argument to a function:

```
def print_course_detail(x):
        print ("Course %s has Price %d INR and its publuisher is %s " %(x.name,x.price,x.publisher ) )

spark=TrainingCourse()
spark.name= "Spark Professional Training"
spark.price=4500
```

```
spark.publisher="HadoopExam.com"

print_course_detail(spark)
```

Object containing another Object (Embedded Object) :

```
class Trainer(object):
        "Trainer Details"


spark_trainer=Trainer()
spark_trainer.name="Amit"
spark_trainer.experience=12
spark_trainer.course_detail=spark


print (spark_trainer.name , spark_trainer.experience , spark_trainer.course_detail.price,
spark_trainer.course_detail.name, spark_trainer.course_detail.publisher)
```

You can return Class Instance as a function value:

```
class TrainingCourse(object):
        "This User Defined variable represents the HadoopExam Training Courses"


hadoop = TrainingCourse()
hadoop.name= "Hadoop Professional Training"
hadoop.price=3900
hadoop.publisher="HadoopExam.com"

def total_price(course,tax):
        course.price=course.price+course.price*tax/100
        return course


hadoop=total_price(hadoop,10)

#Price after applying tax to course price
print ("Course %s has Price %d INR and its publuisher is %s "
%(hadoop.name,hadoop.price,hadoop.publisher ) )
```

Object Mutability:

- Above 'hadoop' TrainingCourse object is mutable, we changed its price.
- By passing the same object to a function, we have changed the price.

Object Copying:

- Using copy function, we can duplicate an object
- There is a module named 'copy' we have to import it.

```
import copy
hadoop_new = copy.copy(hadoop)

#Printing new course detail
print ("Course %s has Price %d INR and its publuisher is %s "
%(hadoop_new.name,hadoop_new.price,hadoop_new.publisher ) )
```

Object Equality: Check whether they are same or different object

```
hadoop_new is hadoop
hadoop_new == hadoop #Default behavior is same as is operator
```

Copying Embedded Object: Copy function does not copy embedded object

```
spark=TrainingCourse()
spark.name= "Spark Professional Training"
spark.price=4500
spark.publisher="HadoopExam.com"
print_course_detail(spark)


class Trainer(object):
        "Trainer Details"


spark_trainer=Trainer()
spark_trainer.name="Amit"
spark_trainer.experience=12
spark_trainer.course_detail=spark


print (spark_trainer.name , spark_trainer.experience , spark_trainer.course_detail.price,
spark_trainer.course_detail.name, spark_trainer.course_detail.publisher)


#Copy the trainer object
new_trainer=copy.copy(spark_trainer)

spark_trainer is  new_trainer #This is a different object
spark_trainer.course_detail is new_trainer.course_detail #This will return true, its same object
```

Shallow Copy: While copying the object, only main object will be copied and any embedded object will not be copied. Internal references will still point to the same object. And this is not good.

Deep Copy: Duplicating the entire object, including embedded object.

---

new_trainer=copy.deepcopy(spark_trainer)

spark_trainer is new_trainer **#This is a different object**
spark_trainer.course_detail is new_trainer.course_detail **#This is also a different object**

Checking whether an object has an attribute or not: You can use in-built unction hassttr

---

hasattr(spark_trainer, 'last_name') #It should return false
hasattr(spark_trainer, 'name') #It should return true

All Products List of www.HadoopExam.com

TRAINING'S (AVAILABLE)

- [Hadoop BigData Professional Training](#)
- [HBase (NoSQL) Professional Training](#)
- [Apache Spark Professional Training](#)
- [Apache OOZie (Hadoop workflow) Professional Training](#)
- [Beginner AWS Training Course- **(HETRNAWS101**)](#)
- [Core Java 1z0-808 Exam training](#)
- [JAX-WS (Java WebService HandsOn Training)](#)
- [Scala Programming Training](#)
- [Python Programming Training](#)
- [Hortonworks Administration Professional Trainings](#)

MAPR HADOOP AND NOSQL CERTIFICATION (AVAILABLE)

- **[MapR Hadoop Developer Certification](#)**
- **[MapR HBase NoSQL Certification](#)**
- **[MapR Spark Developer Certification (In Progress)](#)**

HORTONWORKS HADOOP AND NOSQL CERTIFICATION (AVAILABLE)

- [HDPCD : NO Java (Hortonworks Developer Certification)](#)
- [HDPCD : Spark (Spark Developer Certifications)](#)
- [HDPCA : Hortonworks Administration Certification](#)
- [Hortonworks Administration Professional Trainings](#)

CLOUDERA HADOOP AND SPARK CERTIFICATION (AVAILABLE)

- CCA131 : Hadoop Administrator
- CCA-175 Cloudera® (Hadoop and Spark Developer)
- CCP:DE575 : Cloudera® Data Engineer Certification
- CCA159 : Cloudera Data Analyst Certifications

DATABRICKSA OREILLY SPARK CERTIFICATION (AVAILABLE)

- Apache Spark Developer

AWS: AMAZON WEBSERVICE CERTIFICATION (AVAILABLE)

- AWS Solution Architect : Associate
- AWS Solution Architect: Professional
- AWS Developer : Associate
- AWS Sysops Admin : Associate

MICROSOFT AZURE CERTIFICATION (AVAILABLE)

- Azure 70-532
- Azure 70-533

DATA SCIENCE CERTIFICATION (AVAILABLE)

- EMC E20-007

EMC CERTIFICATIONS (AVAILABLE)

- EMC E20-007

SAS ANALYTICS CERTIFICATION (AVAILABLE)

- SAS Base A00-211
- SAS Advanced A00-212
- SAS Analytics : A00-240
- SAS Administrator : A00-250

ORACLE JAVA CERTIFICATION (AVAILABLE)

- Java 1z0-808
- Java 1z0-809
- Java 1z0-897 (Java WebService Certification)

ORACLE DATABASE CLOUD CERTIFICATION (AVAILABLE)

- 1z0-060 (Oracle 12c)
- 1z0-061 (Oracle 12c)

Subscribe Here for Regular Updates: Like New Training Module launched

Become Author and Trainer: We are looking for Author (Writing Technical Books) and Trainer (Creating Training Material): **No Compromise on Quality.**

Benefit: You will get very good revenue sharing. Please drop us an email to hadoopexam@gmail.com (For the skills, you feel you are master)

We are sure, you are good at least one technology. Don't limit your potential, contact us immediately with your skill. Our expert team will contact you with more detail. You training and Books will reach to all our existing network and with our expert marketing team we will help you to reach as much as technical professional, with our Smart Advertising network. Contact us with sending an email hadoopexam@gmail.com

Opportunity to share your knowledge with all learners who are in need. We are helping 1000's of learners since last 4 years and established ourselves with Quality low cost material.