Core Java – Assignments

Session Name : **Assignment Details**

Introduction to Classes and Members Part-I

Assignment-I

- Create a new Class Customer
- Create 2 Class Members - custId and custName
- Add new method print()
- Create instance of the class and invoke print method

Assignment-II

- Create total 4 member variables for Customer class
- Assign public, private, protected and default access modifiers respectively to these variables
- Try to access these variable from other classes [same package and different package]

Assignment-III

- Define a static variable custCount. This contains total count of customers
- Create 2 instances of Customer object
- Increment the custCount static variable and print its value after every increment
- Access custCount using class name as well as object name

Introduction to Classes and Members Part-II

: Assignment-I

- Define a final variable MAXPHONES. This contains maximum allowable phones for a customer record
- Initialize MAXPHONES
- Try to initialize in the constructor
- Access it using class name and object name

Assignment-II

- Define a getter method for empId variable of Employee class
- Define setter method for empID variable taking int parameter
- Define a print method which prints the empID

Introduction to Methods

: Assignment-I

- Define a static variable empCount for Employee class
- Define a static method to increment the empCount variable
- Invoke increment() method using class name
- Invoke increment() method using object name

Assignment-II

- Define a print method with variable length argument list of type String
- Print the contents of the argument list using for loop

Assignment-III

- Create Employee class
- Create instance of Employee class using default constructor. Does it execute?
- Now define default constructor, print a message inside it and execute the program
- Try to add one more constructor taking empID and empName as parameters. Create another object of Employee using this second constructor. Does the program execute?
- Remove the default constructor definition. Does the program execute?

Constructors

: Assignment-I

- Define a private constructor for Employee class
- Define a protected constructor for Employee class
- Invoke both private and protected constructors from the main method. Does the program execute?
- Invoke both private and protected constructors from the subclass say ContractEmp. Does the program execute?
- Remove the public constructor for Employee class. How will another class create instance of Employee class?

Inheritance Part-I :Assignment-I

- Create a subclass Contractor for base class Employee
- Define a constructor for Contractor. Can you initialize all variables defined in the base class?
- Create instance of subclass Contractor. Are base class constructors accessible?
- Invoke Contractor's constructor. Which all constructors are getting invoked?
- Define a variable in Contractor matching its definition in the base class Employee. Is that allowed?

Inheritance Part-II :Assignment-II

- Implement equals method in Employee class
- Validate that the parameter to equals method is instance of Employee
- Perform comparison on empName
- Test that equals method returns true for two different objects having same empName

Polymorphism Part-I :Assignment-I

- Create base class Customer and subclasses SilverCustomer and GoldCustomer
- Define discount() method in Customer class whcih returns 20% discount
- Overload discount method in the subclasses and return different discount value
- Define base class variable as "Customer cust"
- Assign different objects of Customer, SilverCustomer and GoldCustomer to variable cust one after other and invoke discount method each time. What is the discount % returned each time?

Assignment-II

- In the earlier assignment, how will you access the discount() method of Customer class from SilverCustomer and GoldCustomer classes?

Assignment-III

- Continue earlier assignment in Polymorphism Part-I
- Define another discount method in Customer class taking String location as parameter. Can it be invoked?
- Can this method be invoked from SilverCustomer or GoldCustomer classes?
- Try to overload this discount(String) method in SilverCustomer class. Try to invoke it. Does the program run?

Polymorphism Part-II : Assignment-I

- Continue earlier assignment
- Define a new method getCustomer. This should return instance of Customer or SilverCustomer or GoldCustomer class randomly.
- Main method will call getCustomer and then invoke the discount() method on the object received. Execute and observe the results.
- Now create a new PlatinumCustomer class extending Customer class. Implement discount() method in this class
- How do you invoke discount method of PlatinumCustomer class without changing main method?

Abstract Classes

:Assignment-I

- Create abstract class Car
- Define an abstract method ignition()
- Define a non-abstract/normal method changeGear(). Is this allowed?
- Create concrete classe Sedan. Overload ignition methods
- Create instance of Sedan and invoke ignition() and changeGear() methods. Does the program execute?
- Define a variable noOfWheels in Car class. Can it be accessed in Sedan class?

Assignment-II

- Continue earlier assignment
- Create an abstract class SUV extending Car class. Do you need to implement all abstract methods of Car class here?
- Define method 4WheelDrive() in SUV class
- Create a concrete class Safari extending SUV class. Which methods must be implemented here?

Interfaces

: Assignment-I

- Create a new interface to develop a simple Calculator
- Define two methods sum and divide in the interface. Can you implement the methods inside the interface?
- Create a new class which implements above interface. Do you need to implement both the methods here?
- Define a variable inside calculator interface? Can it be invoked from calculator implementation?
- Now define one more interface ScientificCalculator and add couple of methods
- Can you implementation class implement this new interface as well? Try it.

Assignment-II

- Continue earlier assignment
- Define a variable whose type is an interface for the calculator
- Assign an object of calculator implementation
- Invoke methods defined by the interface using the interface reference. Does the program execute?

Assignment-III

- Define a method doCalc() in a class. It accepts an int parameter iVal
- Perform following calculation "return 100/iVal;"
- Invoke it from main method by passing zero value. What happens when you execute?
- Introduce try-catch block around the method call. Which exception you need to catch for a meaningful message?
- Introduce a finally block after the catch block. When does it get called?
- In the catch block make sure that you print the complete stack trace of error.
- Write following code below the method call doCalc() "String str = argv[2];". Which exception is thrown on execution?
- Handle the exception using another catch block.

Exception Handling Part-I

: Assignment-I

- Continue earlier assignment
- If you remove the two catch block, do you get any error during compilation time?
- Define a new function save()
- Try to open a physical file as follows: "FileOutputStream fout = new FileOutputStream("I:\\emp.dat");". What happens when you compile this program?
- Handle the exception inside the save method using try-catch block
- Invoke this method save() from the main method. What happens when you execute this program?

Assignment-II

- Continue earlier assignment in Exception Handling Part-I session
- Remove the try-catch block from the save method.
- Instead declare the exception in the method signature for save() method. Does the program compile without errors?
- Handle the exception in the main method by adding another catch block
- What happens when you execute this program?

Exception Handling Part-II

: Assignment-I

- Continue with earlier assignment
- In the catch block of runtime exception viz ArrayIndexOutofBoundsException, throw a normal Exception
- Provide a custom message like "Error accessing parameters" etc.
- What happens when you execute this program?

Assignment-II

- Raise new Error("...") randomly from the main method.
- Execute the program couple of times till you get the trace of Error.
- What information does the Error trace provide?

Assignment-IV

- Create a custom exception by extending base class Exception.
- Define a member variable solution and provide getter/setter methods.
- Override printStackTrace method. How will you print your custom message followed by exception stack trace?
- Throw this custom exception instead of normal Exception and observe the stacktrace