Explain these OOPS concepts :

Polymorphism  
Inheritance  
Abstraction  
Encapsulation  
Aggreagation  
Composition  
Association

What is default class modifier?  
What are the different method access modifiers?  
What is the use of a final modifier on a class?  
What is the use of a final modifier on a method?

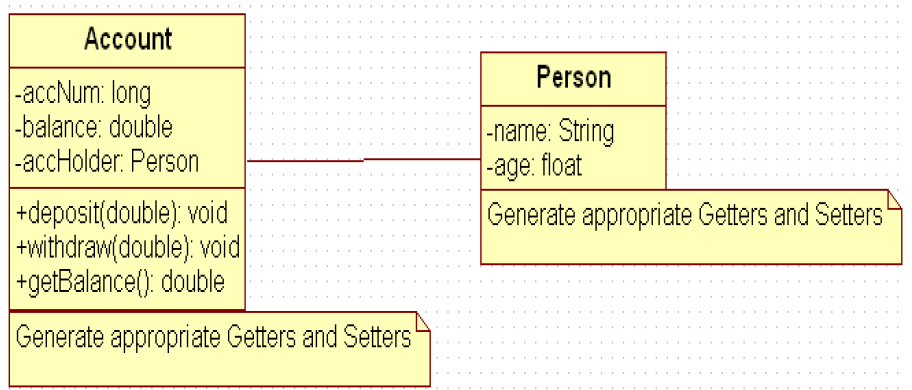
Which package is always imported by default?  
Can I import same package/class twice? Will the JVM load the package twice at runtime?  
Does importing a package imports the sub packages as well? E.g. Does importing com.bob.\* also import com.bob.code.\*?  
What is a Java package and how is it used?  
Explain the usage of Java packages.  
Are the imports checked for validity at compile time?

Difference between Public, Private, Default and Protected ?  
 Can we reduce the visibility of the overridden method ?  
 Which access specifier can be used with Class ?  
 Can we reduce the visibility of the inherited or overridden method ?  
 What will happen if we make the constructor private ?  
 Can we instantiate the object of derived class if parent constructor is protected ?  
 Can we declare an abstract method private ?

What is strictfp keyword and when do you use it?

What is instanceOf operator?

Q)create Account Class as shown below in class diagram. Ensure minimum balance of INR 500 in a bank account is available.



a) Create Account for smith with initial balance as INR 2000 and for Kathy with initial balance as 3000.(accNum should be auto generated).

b) Deposit 2000 INR to smith account.

c) Withdraw 2000 INR from Kathy account.

d) Display updated balances in both the account.

e) Generate toString() method.

Extend the functionality through Inheritance and polymorphism (Maintenance)

Inherit two classes Savings Account and Current Account from account class. Implement the following in the respective classes.

1) Savings Account

a. Add a variable called minimum Balance and assign final modifier.

b. Override method called withdraw (This method should check for minimum balance and allow withdraw to happen)

2) Current Account

a. Add a variable called overdraft Limit

b. Override method called withdraw (checks whether overdraft limit is reached and returns a boolean value accordingly)

**Using packages**

create an application for that requirement by creating packages and classes as given below:

a) com.cg.eis.bean

In this package, create “Employee” class with different attributes such as id, name, salary, designation, insuranceScheme.

b) com.cg.eis.service

This package will contain code for services offered in Employee Insurance System. The service class will have one EmployeeService Interface and its corresponding implementation class.

c) com.cg.eis.pl

This package will contain code for getting input from user, produce expected output to the user and invoke services offered by the system.

The services offered by this application currently are:

i) Get employee details from user.

ii) Find the insurance scheme for an employee based on salary and designation.

iii) Display all the details of an employee.

: Use overrides annotation for the overridden methods available in a derived class of an interface of all the assignments.

Modify account class as abstract class and declare withdraw method.