1. Nill Pointer Exception because we are trying to do opertion with null value.

2. Exception is an event that reults in stopping the execution of program and
stops the below line code execution.
We can Handle exception in 5 different ways by using keywords listed as below:
a. throw
b. throws
c. try block
d. atch block
e. finalyy block

3.Custom Exception is an exception created by programmer to throw an business required exception.
We are writing the custom exception to handle the exception that may occur while executing the business logic.

4.Encapsulation is a process of binding all the class members in to an single entity is call encapsulation.
The single entity here is Java Bean Class.

Rules for Encapslation:
a. Class cannot be final
b. All the properties must be declared as private.
c. All the properties should have getter and setter methods.
d. Class should be public.

5.Polymorphism is the process where in objects performs the different behaviour.
Types:
a. Runtime Polymorphism
b. Compiletime Polymorphism.

6.Overloading is the process where in class will multiple methods of same name but differ in the number of
parameters,sequence of data type and return type can be different.
Example:

```
Class Calculator{

      public static int addition(int a,int b){
 return a+b;
}
public static int addition(int a,int b,int c){
 return a+b+c;
}
public static  int addition(int a,int b,int c,int d){
 return a+b+c+d;
}
public static void main(String[] args){
System.out.println(addition(10,20));
System.out.println(addition(10,20,30));
System.out.println(addition(10,20.30,40));
```

```
	}
	}
```

7.Method Overriding is the process where child class extends parent class and changes implementaion of parent class method
 where in method signature should be same and accesifier should be of same type or of higher visibility.
 Example:

```
package com.thoughtfocus.assesmenttwo.methodoverriding;

public class Battery {
public void batteryCharge() {
	System.out.println("The Battery life is reduced");
}
}
package com.thoughtfocus.assesmenttwo.methodoverriding;

public class Vehicle extends Battery{

	@Override
	public void batteryCharge() {
			// TODO Auto-generated method stub
			System.out.println("The Battery has Recharged");
	}

}
package com.thoughtfocus.assesmenttwo.methodoverriding;

public class Tester {
public static void main(String[] args) {
	Vehicle bike=new Vehicle();
	bike.batteryCharge();
}
}
```

8.OutPut:- args

9.Compile Time Error
 Because while overloading parameters should be different

10. Compile Time Error
   Because class cannot implements the class

11.Abstraction is a process of hiding the business logic/implementation and showing only functionality to the user
   is called abstraction.

 Abstraction can be achived by using abstract class and interface
 100% Abstraction can be achieved by using the interface.

12.we can initialize the value in
  a. litteral way
  b. By using Object reference
  c. By using constructors

d. By using methods

Example:

```
package com.thoughtfocus.assesmenttwo.initialization;

public class ValueInitialization {
int age;
String name;
long phoneNumber;
String dateOfBirth="05 April,1998";//Litteral Way

//By Using Constructors
public ValueInitialization(String name, long phoneNumber) {
      this.name = name;
      this.phoneNumber = phoneNumber;
}

//Initializing local variable in Method
public void initializeValue(String address) {
      address="Dharwad";
      System.out.println(address);
}

public static void main(String[] args) {
      ValueInitialization initialize=new ValueInitialization("Pramod",
9876541321);
      initialize.age=23;//Initializing using object reference
      System.out.println(initialize.age);
}

}
```