

Assessment – 2

24-12-2021

K. T. Chaitanya Sai

1. Null pointer exception
2. An exception is an event that disrupts the normal flow of the program. It is an object which is thrown at runtime. When an error occurs within a method, the method creates an object and hands it off to the runtime system. The object, called an exception object, contains information about the error, including its type and the state of the program when the error occurred. Creating an exception object and handing it to the runtime system is called throwing an exception.

Exception Handling : exception handling is to maintain the normal flow of the application. An exception normally disrupts the normal flow of the application; that is why we need to handle exceptions.

Exception can be handled using five keywords : try, catch, throw, throws and final.

3. custom exceptions are used to customize the exception according to user need. Creating our own Exception is known as custom exception or user-defined exception. Custom exception is used for To catch and provide specific treatment to a subset of existing Java exceptions. Business logic exceptions: These are the exceptions related to business logic and workflow. It is useful for the application users or the developers to understand the exact problem.
4. Encapsulation refers to the bundling of fields and methods inside a single class.

Rules of Encapsulation

- All variables must be private.
- Add getters and setters
- Should have atleast one default Constructor.
- Class should be in public.
- Class cannot be declared as final.

5. Polymorphism is a concept of performing a single action in different ways. Polymorphism is derived from 2 Greek words: poly and morphs. The word "poly" means many and "morphs" means forms. So polymorphism means many forms.

There are two types of polymorphism in Java:

Runtime polymorphism.

compiletime polymorphism.

6. Overloading is a feature that allows a class to have more than one method having the same name.

```
public class OverLoading {  
    void Add(int a, int b) {  
        System.out.println(a+b);  
    }  
    void sub(double c, double d) {  
        System.out.println(c-d);  
    }  
}
```

```
public class Tester {  
    public static void main(String[] args) {
```

```

        OverLoading overloading = new OverLoading();
        overloading.Add(5, 10);
        overloading.sub(25.53, 15.25);
    }
}

```

7. Overriding is a child class has its method implementation for the method already present in the parent class. overriding is a function that requires a subclass or child class to provide a variety of method implementations, that are already provided by one of its superclasses or parent classes.

```

public class India {
    void averagetemperature() {
        System.out.println("India average temperature is
25.78 Degree Celsius");
    }
}

public class AndhraPradesh extends India{
    @Override
    void averagetemperature() {
        System.out.println("Andhra Pradesh average temperature is
27.58 Degree Celsius");
    }
}

public class Karnataka extends India{
    @Override
    void averagetemperature() {
        System.out.println("Karnataka average temperature is
24.87 Degree Celsius");
    }
}

public class Runtime {
    public static void main(String[] args) {
        India india = new India();
        AndhraPradesh andhrapradesh = new AndhraPradesh();
        Karnataka karnataka = new Karnataka();
        india.averagetemperature();
        andhrapradesh.averagetemperature();
        karnataka.averagetemperature();
    }
}

```

8. Args
9. error
10. X-workz provide java Enterprise application training
11. Abstraction is a process of hiding the implementation details and showing only functionality to the user.

There are two ways to achieve abstraction

- Interface.
- Abstract Class.

By using interface we can achieve 100% abstraction.

12. ways to initialize value
 - Datatype : Type of data can be stores in variable.
 - Variable name
 - Value : initial value is stored.