Xworkz sep Batch test

INHERITANCE :- Inheritance is concept of acquiring all the properties from one object to another object.

Basically here there will be a parent and child relationship

Where child will aquire all the properties of parent object.

There are 5 types of Inheritance :-

1. Single inheritance
2. Multilevel inheritance
3. Hierarchical inheritance
4. Multiple inheritance
5. Hybrid inheritance
6. Single inheritance:- where single parent and single child(child will aquire all the properties of parent). By using keyword called inheritance.

Parent object (class bird { )

extends

Child object (class parrot { )

1. Multilevel inheritance:- there will be a chain of inheritance. As shown in bellow diagram.

Parent object(class Animal{ )

extends

Child object(class dog{ )

extends

Sub child object (class cat{ )

1. Hierarchical inheritance:- Here one parent and many child (many child can aquire properties of one parent).

Parent object(class rbi{ )



Child object(class hdfc{ )

Child object(class canara{ )

extendsssss

4.Multiple inheritance:- Here two parent one child and it not supported by Java classes it is supported by Interface in java because ambiguity issue is created.

Parent object

Parent object

Child object

5.Hybrid Inheritance :- It is combination of Hirarchy and Multiple inheritance.

Class A

Class B

Class C

Class D

2. Polymorphism:- Performing Single task in Multiple ways is known as Polymorphism.

There are two types of polymorphism

1. Method overloading
2. Method overriding

Method overloading :- is a mechanism where method name is same in a class and parameters should be different for example change in number of parameter or change in data type of parameter.

Method overriding :- To override the method there should be is a relationship where method name should be changed and parameter also be the same

Where child object method will be override the parent object method

3. Abstraction :- is a mechanism where hiding the implementation and showing the behaviour or functionality of an object.

A class which is declared with the abstract keyword is known as an abstract class in [Java](https://www.javatpoint.com/java-tutorial). It can have abstract and non-abstract methods (method with the body). It can have constructor and static method also

Ways to achieve abstraction

1. Through Abstraction( we can achive abstraction through abstract clss is 0-100% )

2. Interface (through interface we can achieve 100% abstraction)

4. Encapsulation :- mechanism of binding data and behaviour of an object into single unit is known as Encaosulation.

We can create a fully encapsulated class in Java by making all the fields of the class private. Now we can use setter and getter methods to set and get the data in it.

The **Java Bean** class is the example of a fully encapsulated class.

2. JDBC stands for Java Database Connectivity. JDBC is to connect and execute the query with the database.

JDBC can perform the following tasks:

1. Connect to the database
2. Execute queries and update statements to the database
3. Close Connection

1 2 3

connection

Execute and close the connection

Writing a query

JDBC Driver:- Driver is a Class that enables java application to interact with the database.

The **forName()** method of Class is used to call the driver class. This method is used load the driver class.

The **getConnection()** method of DriverManager class is used to establish connection with the database.

The createStatement() method of Connection interface is used to create statement. The object of statement is responsible to execute queries with the database.

The executeQuery() method of Statement interface is used to execute queries to the database. This method returns the object of ResultSet that can be used to get all the records of a table.

The close() method of Connection interface is used to close the connection.

Commonly used methods of Statement interface:

|  |
| --- |
| 1)ResultSet executeQuery(String sql): is used to execute SELECT query. It returns the object of ResultSet. For read operation it is used |
| 2)int executeUpdate(String sql): is used to execute specified query, it may be create, drop, insert, update, delete it will return int value like 0, 1, 2…. So on |
| 3)boolean execute(String sql): is used to execute queries that may return Boolean values. |

preparedStatment :- is used for make application as dynamic here we will use place holder

statement:- we give the value directly instead of place holder only that particular value will be executed

3. Collection :-

Iterable (I)

Collection (I)

List (I)

Set (I)

Queue (I)

Priority queue(C)

Array list (c)

Hash set(C)

Linked Hash Ser(I)

Linked list(C)

Deque (I)

Stack(c)

Vector(C)

Sorted set (I)

Array deque(c)

Tree set(c)

I- interface

C – class

The Iterable interface is the root interface for all the collection classes.

The Collection interface is the interface which is implemented by all the classes in the collection framework.

Collection is collection of different type object in a single unit .

List interface is the child interface of Collection interface. in which we can store the ordered collection of objects. It can have duplicate values.

List interface is implemented by the classes ArrayList, LinkedList, Vector, and stack

Queue interface maintains the first-in-first-out order.

There are various classes like PriorityQueue, Deque, and ArrayDeque which implements the Queue interface.

Set Interface in Java It extends the Collection interface.

It represents the unordered set of elements

doesn't allow to store the duplicate elements.

We can store only one null value in Set.

Set is implemented by HashSet, LinkedHashSet, and TreeSet.

**LinkedList**

LinkedList implements the Collection interface.

It uses a doubly linked list internally to store the elements.

It can store the duplicate elements.

It maintains the insertion order

it is not synchronized.

In LinkedList, the manipulation is fast because no shifting is required.

Interal working of linkedList

Basically there will be a 3 nodes the 1 node will have the address of prvious element middle node will have present address value and last node will have address of next value

3rd node

2nd node

1st node

Prvious next

Tree set

TreeSet also contains unique elements. The elements in TreeSet stored in ascending order.

doesn't allow to store the duplicate elements.

We can store one null value in Set.

But from java 8 we can stoe multiple value in tree set

HashMap:-

Java **HashMap** class implements the Map interface which allows us to store key and value pair, where keys should be unique. It is easy to perform operations using the key index like updation, deletion, HashMap class is found in the java.util package.

HashMap in Java is like the legacy Hashtable class, but it is not synchronized. It allows us to store the null elements , but there should be only one null key.

Java HashMap contains values based on the key.

it contains only unique keys.

it may have one null key and multiple null values.

It is non synchronized.

maintains no order.

The initial default capacity of Java HashMap class is 16 with a load factor of 0.75.

SERVLET

Servlet is a technology which is used to create a web application.

Servlet is a web component that is deployed on the server to create a dynamic web page.

There are many interfaces and classes in the Servlet such as HttpServletRequest, ServletResponse, etc.

The HTTPRequest:- client sends the request to the server in the form of request message.

The response is used to provide the information to client with the resource it requested. It is also used to inform the client that the action requested has been carried out.

RequestDispatcher :- The RequestDispatcher interface provides the facility of dispatching the request to another resource it may be html, servlet or jsp. This interface can also be used to include the content of another resource also.

There are two methods

Include:- Includes the content of a resource in the response.

forward :- it Forwards a request from a servlet to another resource on the server.

HIBERNATE :-(ORM TOOL)

Hibernate is a Java framework that simplifies the development of Java application to interact with the database.

The Hibernate includes many objects such as session factory, transaction , session, etc.

#### SessionFactory

The SessionFactory is a factory of session and method of Connection. The org.hibernate.SessionFactory interface provides factory method to get the object of Session.

#### Session

The session object provides an interface between the application and data stored in the database. The org.hibernate.Session interface provides methods to insert, update and delete the object. It also provides factory methods for Transaction, Query and Criteria.

#### Transaction

The transaction object specifies the atomic unit of work it is usedto begin the task .

Java Persistence API (JPA) is a Java specification that provides certain standard to ORM tools. The **javax.persistence** package contains the JPA classes and interfaces.

JPA is just a specification that facilitates object-relational mapping to manage relational data in Java applications. It provides a platform to work directly with objects instead of using SQL statements.

As JPA is just a specification, it doesn't perform any operation by itself. It requires an implementation. So, ORM tools like Hibernate, TopLink and eclipselink implilements JPA specifications for data persistence.

Entity manager is used to read, delete and write an entity.

The **EntityManagerFactory** interface is used to provide an entity manager.

The role of this method is to create and return an EntityManagerFactory for the named persistence unit. Thus, this method contains the name of persistence unit passed in the Persistence.xml file.

JPA is only a specification it is not an implementation but

Hibernate is implementation of JPA