**Design Pattern**

1. **What is design pattern?**

* **Design patterns are solutions to general problems that software developers faced during software development.**
* **During our software development we will face some kind of general problem, we have solution for the same which is called design pattern.**
* **They are generally targeted at solving the problems of object generation and integration.**

1. **what are the design pattern you worked?**

* **Singleton – This pattern is used when we need to create that only one instance of a particular class need to be created in JVM. Make sure the singleton class provide global access point to get the instance. Used to create logging, caching, data base connection creation**
* **Factory – Factory pattern creates object without exposing the creation logic to the client and refer newly created object using a common interface. Mainly it is used to when we have multiple sub classes of a Super class & based on input we want to return one class instance.It provides the abstraction between the implementation & client classes**
* **Builder Patternv-** The builder pattern is a design pattern that allows for the step-by-step creation of complex objects using the correct sequence of actions. If there is too many attributes and we want to create object by passing only few attribute we can use builder design pattern.also here param sequence not matter.
* **Prototype Pattern -** prototypes design pattern allows you to create objects by **cloning** an existing object instead of creating a new object from scratch. This pattern is used when the process of **object creation is costly**.

1. **what is singleton design pattern, Write the program for Singleton Design Pattern**
2. **How to break singleton design pattern**
3. **what is Factory Design pattern, Write the program for Factory Design Pattern**
4. **what is builder pattern write the program for factory pattern**

**RestApi**

**1.how you will create rest api in springboot**

**2. what is idemponent**

**SQL-theory**

1. what is primary key
2. what is foreign key
3. primary key vs forignkey
4. what is indexing/why we need indexing
5. what is candidate key
6. **what is table & field**

**SQL-Query**

1. **write a query to find the second maximum salary of employee**

* **select name, max(salary) from Employee where salary not in (select max(salary) from employee) (need to test in local)**
* **select min(salary) from  (select distinct salary from emp order by salary desc)  where rownum < 3;**

1. **write a query to get the login status of user based on timestamp (last login status)**
2. **select dept , sum of salary from each department**
3. **find the min , max salary of given employee**
4. **groupby, max,min, limit sql quries**

**Jdbc/Hibernate/JPA**

**1.What is the JDBC?**

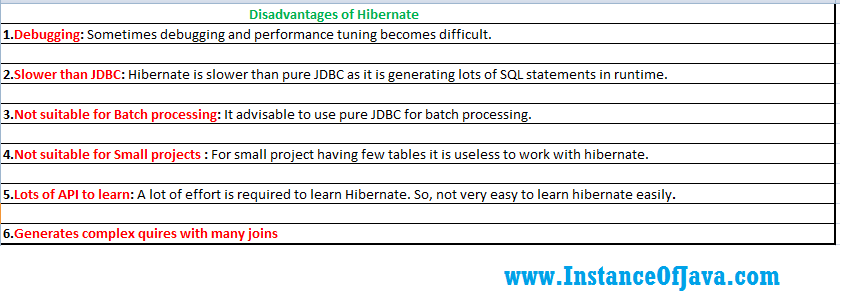
* **JDBC stands for Java Database Connectivity.**
* **It is a free open-source application programming interface for Java which enables the application to access the database.**
* **It enables the developers to create queries, update data to the relational database using a Structured Query Language (SQL).**

**2. what is hibernate? why we are using hibernate, advantage, disadvantage**

* **Hibernate is an open-source and lightweight ORM tool that is used to store, manipulate, and retrieve data from the database.**
* **ORM is an acronym for Object/Relational mapping. It is a programming strategy to map object with the data stored in the database. It simplifies data creation, data manipulation, and data access.**

**Advantage:**

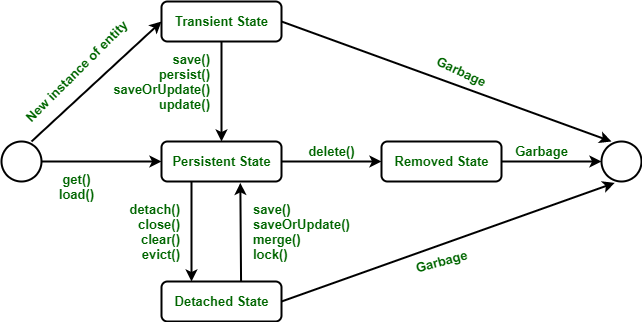
* **It is ORM tool**
* **Hibernate has an exception translator, which converts checked exceptions of JDBC in to unchecked exceptions of hibernate. So all exceptions in hibernate are unchecked exceptions and Because of this no need to handle exceptions explicitly.**
* **Hibernate has its own query language called Hibernate Query Language. With this HQL hibernate became database independent.**
* **Hibernate supports relationships like One-To-One, One-To-Many, Many-To-One ,Many-To-Many.**
* **Hibernate has Caching mechanism. using this number of database hits will be reduced. so performance of an application will be increases.**
* **Hibernate also supports annotations along with XML.**
* **Hibernate supports Lazy loading.**
* **Hibernate is easy to learn it is developers friendly.**
* **Hibernate maintains database connection pool.**
* **Hibernate has Concurrency support.**

[**[](https://1.bp.blogspot.com/-zGgLuXdqU8U/WAtr0-Y_TeI/AAAAAAAABVg/4hKLy7M8AQcRXslZFSEytlV38YGdnfpYQCLcB/s1600/advantages%2Band%2Bdisadvantages%2Bof%2Bhibernate.png)**](https://1.bp.blogspot.com/-zGgLuXdqU8U/WAtr0-Y_TeI/AAAAAAAABVg/4hKLy7M8AQcRXslZFSEytlV38YGdnfpYQCLcB/s1600/advantages%2Band%2Bdisadvantages%2Bof%2Bhibernate.png)

1. **Hibernate life cycle?**

**There are mainly four states of the Hibernate Lifecycle :**

1. **Transient State**
2. **Persistent State**
3. **Detached State**
4. **Removed State**

****

**Eg:** [**https://www.geeksforgeeks.org/hibernate-lifecycle/#:~:text=There%20are%20mainly%20four%20states,Detached%20State**](https://www.geeksforgeeks.org/hibernate-lifecycle/#:~:text=There%20are%20mainly%20four%20states,Detached%20State)

**4. Difference between first level and second level cache**

| **Sr. No.** | **Key** | **First level cache** | **Second level cache** |
| --- | --- | --- | --- |
| **1** | **Basic** | **First level cache is a session level cache and it is always associated with session level object** | **Second level cache is session factory level cache and it is available across all sessions** |
| **2** | **Enabled** | **It is enabled by default.** | **It is not enabled by default.** |
| **3** | **Availability** | **It is available for a session** | **It is available across all session.** |
| **4** | **Configuration** | **No Extra configuration required** | **We have to decide  which concurrency strategy to use and also need to configure cache expiration and physical cache attributes.** |

**5.differennce between update and merge?**

| **Update** | **Merge** |
| --- | --- |
| **Hibernate proprietary method** | **Part of the JPA specification** |
| **Does not return anything, returns a void** | **Returns the updated object** |
| **In update() method, if the object already exists in Cache, then update() method will throw an Exception.** | **In merge() method, if the object already exists in Cache, then merge() method will simply copy the object into Cache.** |

**6.difference between get and load in hibernate?**

| **Sr. No.** | **Key** | **Get()** | **Load()** |
| --- | --- | --- | --- |
| **1** | **Basic** | **It is used to fetch data from the database for the given identifier** | **It  is also used to fetch data from the database for the given identifier** |
| **2** | **Null Object** | **It object not found for the given identifier then it will return null object** | **It will throw object not found exception** |
| **3** | **Lazy or Eager loading** | **It hit the database and returns original object** | **It always returns proxy object** |
| **4** | **Performance** | **It is slower than load() because it return fully initialized object which impact the performance of the application** | **It is slightly faster.** |
| **5.** | **Use Case** | **If you are not sure that object exist then use get() method** | **If you are sure that object exist then use load() method** |

**7. hibernate criteria**

**8.optimistic locking in hibernate?**

**9.what are the inheritance mapping strategies?**

**10. hibernate mapping with annotations**

**11.How do you create Relations between the entity**

**12.Write a program to define relationship between the Employee and department, 1 dept have multiple employees**

[**https://www.baeldung.com/hibernate-one-to-many**](https://www.baeldung.com/hibernate-one-to-many)

**https://www.javatpoint.com/hibernate-one-to-many-mapping-using-annotation-example**

**11.statement vs prep statement?**

**what is jpa and use of jpa?**

**What is the implementation of JPA**

**11. Difference between Hibernate and Spring jpa?**

**12. did jpa is perform actual task?**

**No implementation like hibernate will perform actual task.**

**Programs:**

**Java MCQ / programs**

1. [**https://www.geeksforgeeks.org/java-tricky-output-questions/?ref=lbp**](https://www.geeksforgeeks.org/java-tricky-output-questions/?ref=lbp)

**1. convert hashmap into ArrayList**

**Package:** com.interview.programs.collection

**Program:** HashmapToList.java

**2. Find the maximum repeated character in string**

**Package:** com.interview.programs.string

**Program:** FindMaxRepeatedChar.java

**3.find second non repeating char from string**

**Package:** com.interview.programs.string

**Program:** SecondNonRepeatingCharacter.java

**4.find prime numbers between 1 to 100**

**Package:** com.interview.programs.string

**Program:** PrimeNumbers.java

**5. find out integer start with 1 using streams**

**Package:** com.interview.programs.java8

**Program:** FindIntegerStartWithOne.java

**6. find second duplicate from int array**

**Package:** com.interview.programs.java8

**Program:** SecondDuplicateInteger.java

**7.using java 8 streams filter the employees more than 50000 salary**

**Package :** com.interview.programs.java8

**Program :** FilterEmployee

**8.how to group the employees by department using collection**

**Package :** com.interview.programs.collection

**Program :** EmployeeGroupByDept.java

**9.how to group the employees by department using java8**

**Package :** com.interview.programs.java8

**Program :** EmployeeGroupByDept.java

**10.write the program to find the occurrence of each character in string**

**Package :** com.interview.programs.collection

**Program :** FindOccurChar.java

**11.write the program to find the max occurrence of character in string using collection**

**Package :** com.interview.programs.collection

**Program :** FindMaxRepeatedChar.java

**12.write the program to find the max occurrence of character in string using java8**

**Package :** com.interview.programs.java8

**Program :** FindMaxRepeatedChar.java

**13.write a program to find number of integer occurrence in array, if frequency repeated return max number of array**

**Package :** com.interview.programs.array

**Program :** MaxOccurranceFreq.java

**14.find the output of below program**

**package com.main.test;**

**//Main class**

**class Test {**

**public static void gfg(String s)**

**{**

**System.*out*.println("String");**

**}**

**public static void gfg(Object o)**

**{**

**System.*out*.println("Object");**

**}**

**public static void main(String[] args) {**

**System.*out*.println(100+200+"Java"); //300java**

**System.*out*.println("java"+100+200); //java100200**

**System.*out*.println('j'+'a'+'v'+'a'); //418**

***gfg*(null); //String**

**}**

**}**

* **In case of**[**method overloading**](https://www.geeksforgeeks.org/overloading-in-java/)**, the most specific method is chosen at compile time.**
* **As ‘java.lang.String’ is a more specific type than ‘java.lang.Object’.**
* **In this case the method which takes ‘String’ as a parameter is chosen.**
* **If we have one more method which accept Integer object, As ‘java.lang.String’ and ‘java.lang.Integer’ is a more specific type than ‘java.lang.Object’,but between ‘java.lang.String’ and ‘java.lang.Integer’ none is more specific.**
* **In this case the Java is unable to decide which method to call. So compiler error will throw.**

**15.Write the program to find the first non repeated char from the string**

**Package:** com.interview.programs.string

**Program:** FirstNonRepeatingChar.java

1. **Write the program to split the number as integer and multiply untill single digit like below**

**39 → (3x9=) 27 →**

**Step 2(2x7=) 14 →**

**Step 3(1x4=) 4**

1. **Sort the names in descending order & find the sum of integer squares**

**class Test {**

**// Bhuvaneswari,Sundar, Komal**

**// Sundar,Komal,Bhuvanesari**

**List<String> names = new ArrayList<>();**

**names.add("Bhuvana");**

**names.add("Sundar");**

**names.add("Komal");**

**names.stream().sorted((n1,n2)->n2.compareTo(n1)).forEach(System.*out*::println);**

**Integer var = Stream.*of*(2,4,6,8).max(Integer::*compare*).get();**

**System.*out*.println("=====max value==="+var);**

**Integer min = Stream.*of*(2,4,6,8).min(Integer::*compare*).get();**

**System.*out*.println("=====min value==="+min);**

**//2,4,6,8**

**//Sum of Squrs**

**IntSummaryStatistics sum = Stream.*of*(2,4,6,8).map(i->i\*i).collect(Collectors.*summarizingInt*(Integer::intValue));**

**System.*out*.println("=====max value==="+sum.getMax());**

**System.*out*.println("=====min value==="+sum.getMin());**

**System.*out*.println("=====sum value==="+sum.getSum());**

**System.*out*.println("=====count value==="+sum.getCount());**

**}**

**}**

1. **Write a program to find given two strings are anagram or not**

**public boolean getAnagram(String s1, String s2){**

**Character[] c1 = s1.toCharArray();**

**Character[] c2 = s2.toCharArray();**

**Arrays.sort(c1);**

**Arrays.sort(c2);**

**if(c1.length == c2.length){**

**/\*for(int i=0;i<c1.length;i++){**

**if(c1[i]!=c2[i])**

**return false;**

**}**

**return true;**

**\*/**

**Return Arrays.equals(c1,c2);**

**}else**

**return false;**

**}**

1. **Write a program to distribute T number of candies to n number of students based on scores**
2. **Remove duplicates in the list using stream**

**List<Integer> newList = list.stream()**

**.distinct()**

**.collect(Collectors.toList());**

1. **Find the subfolder from given array whose sum Is given sum**

**package com.example.interview;**

**import java.util.ArrayList;**

**import java.util.Collection;**

**import java.util.Collections;**

**import java.util.HashMap;**

**import java.util.List;**

**import java.util.Map;**

**public class Test {**

**public static void main(String[] args) {**

**Integer arr[] = {1, 4, 20, 3, 10, 5};**

**Integer sum = 33;**

**Integer num=0;**

**Integer start=0;**

**Integer end=0;**

**for(int i =0;i<arr.length;i++) {**

**num=arr[i];**

**start=i;**

**for(int j =i+1;j<arr.length;j++) {**

**num=num+arr[j];**

**if(num>sum)**

**break;**

**if(num==sum) {**

**end=j;**

**System.out.println("====start=="+start+"===end===");**

**display(start,end,arr);**

**}**

**}**

**}**

**}**

**public static void display(Integer start,Integer end,Integer arr[] ) {**

**for(int i=start;i<=end;i++) {**

**System.out.println(arr[i]);**

**}**

**}**

**}**

1. **Find the second largest number from array using stream**

**public** **class** Test {

**public** **static** **void** main(String[] args) {

Integer arr[] = { 20, 1, 40, 20, 3, 10, 5, 40, 20 };

List<Integer> list = Arrays.*asList*(arr);

Integer max1 = list.stream().max(Integer::*compare*).get();

Integer num = list.stream().filter(i -> i != max1).max(Integer::compareTo).get();

System.***out***.println("====max value using two lines===" + num);

Integer max = list.stream().sorted(Collections.*reverseOrder*()).distinct().limit(2).skip(1).findFirst().get();

System.***out***.println("====max value===" + max);

System.***out***.println(list.stream().sorted(Collections.*reverseOrder*()).collect(Collectors.*toList*()));

// [40, 40, 20, 20, 20, 10, 5, 3, 1]

System.***out***.println(list.stream().sorted(Collections.*reverseOrder*()).distinct().collect(Collectors.*toList*()));x

//[40, 20, 10, 5, 3, 1]

System.***out***.println(

list.stream().sorted(Collections.*reverseOrder*()).distinct().limit(2).collect(Collectors.*toList*()));

//[40, 20]

System.***out***.println(list.stream().sorted(Collections.*reverseOrder*()).distinct().limit(2).skip(1)

.collect(Collectors.*toList*()));

//[20]

}

}

1. **What Is the output of below program**

int test() {

  try {

    return 1;

  }

  finally {

    return 2;

    SOP("finally");

  }

  return 3;

}

**it will throw unreachable code compilation error on SOP(finally) and return 3 statement. Because in finally return statement is there. If we removed those statement the output is return 2.**

1. **Output of below program**

**class** A {

**void** m1() {

System.***out***.println("class A");

}

}

**class** B **extends** A {

**void** m1() {

System.***out***.println("class B");

}

}

**public** **class** Test **extends** B {

**void** m1() {

System.***out***.println("class C");

}

**public** **static** **void** main(String args[]) {

A a=**new** B();

Test c=(Test)a;

c.m1();

}

}

* It will throw below run time exception
* Exception in thread "main" java.lang.ClassCastException: class com.example.interview.B cannot be cast to class com.example.interview.Test (com.example.interview.B and com.example.interview.Test are in unnamed module of loader 'app')

at com.example.interview.Test.main(Test.java:33)

* **Because child reference cannot used to hold parent object**

1. **How to find the magic number**
2. **How to sort the HashMap based on keys**