**Real Time Java Interview Companion**

**or**

**FAQ’s And Answers.**

**(Freshers and Experienced Students)**

**By Mr.Anji Reddy (Real time expert).**

1)Who is the author of the java?

Ans)James Gosling. He is the founder of JAVA technology.

2)When java is came to the picture?

Ans)Java is a general purpose,high-level programming language developed by Sun Microsystems.A small team of engineers,known as the Green Team,initiated the language in 1991,Java was originally called “OAK” and the first public implementation was in 1995. And it made promise of “Write Once and Run Anywhere”.

3)How many editions are there in JAVA?

Ans)In Java There are three types of editions are there

i)JSE(java Standard Edition).

ii)JEE(Java Enterprise Edition)

iii)JME(Java Mobile Edition or Micro Edition).

4)Why Java is so popular ?

Ans) Java is so popular, because of two main reasons.

i)Plat form Independence.

ii)Object Oriented Language.

5)what is platform Independence ?

Ans)Any thing (means programe) which is not specific to the Operating System can be called as platform Independency. Platform independency is also called build once,run any where.java is one of the most popular plat form independent language.once we compile a java programe and build a jar,we can the jar in any operating system-where JVM is installed.

Ex:Java technology

6)what is platform dependency ?

Ans)Any thing (means programe) which is specific to the Operating System can be called as platform Independency.

Ex:c,c++…etc

7)what are the important differences between C, C++ and Java ?

Ans)

i)Java is platform independent .C,C++ is not platform independent.

ii)Java is a pure Object Oriented Language.In C,C++ are partial Object Oriented languages.

iii)C++ has pointers.java has no concept called pointers.

iv)In c++,programmer has to handle memory management.In java ,JVM takes of remving objects from memory,using a Garbage Collection.

v)C++ supports multiple Inheritance.Java does not support Multiple Inheritance.

8)What is the role for a ClassLoader in Java ?

Ans)A java programe is made up of a number of custom classes(written by programmers like us) and core classes(which come pre-packed with java).when programe is executed,JVM needs to load the content of all the needed class.JVM uses a ClassLoader to find the classes.

There are three class loaders are there.

i)System Class Loader

ii)Extension Class Loader

iii)Bootstrap Class Loader

iv)when jvm needs to find a class,it starts with system class loader,if it is not found,it checks with Extension Class Loader.if it not found ,it goes to the Bootstrap class loader.if class is still not found,then jvm throws ClassNotFoundException.

9)What are the wrapper classes ?

Ans)Wrapper are the classes using which we can wrap the concept of object on the top of the primitive data . and represent the primitive data in the form of it’s equalent object.

In order to represent 8 primitive data types of java there are 8 wrapper classes.and they are

i)Byte

ii)Integer

iii)Short

iv)Long

v)Float

vi)Double

vii)Character

viii)Boolean

Ex:Integer integer=new Integer(66);//int

Integer integer=new Integer(“66”);//String

10)what are the different utility methods present in wrapper classes ?

Ans)A number of utility methods are defined in wrapper classes to create and convert them primitives.

Value of method:providing another way of creating a wrapper object

Integer hundred=Integer.valueOf(“100”);

xxxValue methods:xxxValue methods help in creating primitives.

Integer integer=Integer.valueOf(57);

int primitive=integer.intValue();

float primitivefloat=integer.floatValue();

parseXxx methods:parseXxxmethods are similar to valueOf but they return primitive values.

Int hundered=Integer.parseInt(“100”);

11)what are the primitive data types and their default values ?

Ans) In java there are 8 primitive data types are there, they are

i)byte🡪1 byte(size)🡪default value is ‘0’.

ii)short🡪2 bytes🡪default value is ‘0’.

iii)int🡪4 bytes🡪default value is ‘0’.

iv)long🡪8 bytes🡪default value is ‘0’.

v)float🡪4 bytes🡪default value is ‘0.0’.

vi)double🡪8bytes🡪default value is ‘0.0’.

vii)char🡪2bytes🡪default value is ‘space’.

viii)boolean🡪1 bit🡪default value is ‘false’.

12)what is the boxing and unboxing ?

Ans)

Boxing:

The concept of representing the primitive data in the form of it’s equalent and corresponding object(wrapper class object) know as boxing.

Ex:

String s1=”123”;

Integer i1=new Integer(s.trim());//this is called a boxing,and trim() is used for to elemenating white spaces.

Un boxing:

The concept of getting back the actival primitive data present in side the object back into it’s equalent and corresponding primitive data value is know as unboxing.

Ex: String s1=”123”;

Integer i1=new Integer(s.trim());//boxing

Int x=i1.intValue();//un boxing

X=x+1;

13) what is the autoboxing and autounboxing ?

Ans)The concept of JVM automatically represent the primitive data in the form it’s equalent and corresponding wrapper class object is known as auto boxing.

The concept of JVM automatically getting back the primitive data present inside wrapper class object back into it’s equalent and corresponding primitive data known as autounboxing.

Ex:

Public class Wdemo{

Public static void main(String[] args)

{

Integer i1=13;//inter nally jvm is converting automatically new Integer(13) this autoboxing.

Int x=i1;

X=x+1;

System.out.println(x);

Integer i2=9;

I2=i2+1;//i2=new Integer(i2.intValue()+1) this is auto unboxing.

System.out.println(i2);

}

}

14)what is the diff between String and StringBuffer classes?

Ans)String is the immutable object,that means once String class object is created we can not modified.any modification on String object creates a new Object.

String str3=”abc”;

Str3.concat(“ed”);

System.out.println(str3);// o/p=abc

Note:that the value of str3 is not modified in the above example.the result should be assigned to a new reference variable.

String concat=str3.concat(“de”);

System.out.println(concat);//o/p=abcde

StringBuffer is mutable object that means we can modifying an object.

StringBuffer br=new StringBuffer(“abc”);

br.append(“de”);

System.out.println(br);//o/p=abcde.

15)Can you give examples of different utility methods in string class ?

Ans)String class defines a number of methods to get information about the String content.

Ex:String str=”abcdefghijk”;

System.out.println(str.charAt(2));output:print the ‘c’.

System.out.println(str.length());out put 11.

System.out.println(str.toString());out put abcdefghi.

System.out.println(“ABC”.equalsIgnoreCase(“abc”));

System.out.println(“abcdefghij”.substring(3));//cdefghij.

System.out.println(“abcdefghij”.substring(3,7));//defg

16)Explain the toString method ?

Ans)toString method is used for to print the content of the an Object.if toString method is not Overriden in a class,the default toString method from Object class is invoked.this would print some hashcode as shown in the example below.

Ex:class Animal{

String name,type;

Public Animal(String name,String type)

{

this.name=name;

this.type=type;

}

Run the following code.

Animal animal=new Animal(“Tommy”,”Dog”);

System.out.println(animal);//Animal@f7e6a96

Out put does not shows the content of animal,to shows the content of the animal object,we can override the default implementation of toString method provided by Object class.

Ex:

class Animal{

String name,type;

Public Animal(String name,String type)

{

this.name=name;

this.type=type;

}

Public String toString()

{

Return “Animal[name=”+name+”,type=”+type+”]”;

}

}

Run the following code.

Animal animal=new Animal(“Tommy”,”Dog”);

System.out.println(animal);//Animal [name=Tommy,type=Dog]

Note:out put shows the content of the animal object.

17)what is the diff between equals method and ‘==’operator ?

Ans)equals method is used for to compare the content of the an object where as ‘==’ operator is used for to compare the references of an objects.

Ex:

String s1=new String(“abcd”);

String s2=new String(“abcd”);

If(s1.equals(‘s2’))

System.out.println(“both are same”);

else

System.out.println(“both are not same”);

Output:both are same.

If(s1==s2)

System.out.println(“both are same”);

else

System.out.println(“both are not same”);

Output:both are not same.

18)what are the important things to consider when implementing equals method ?

Ans)any equals implementation should satisfy these properties:

i)Reflexive:for any reference value x,x.equals(x) returns true.

ii)Symmetric:For any reference values x and y,x.equals(y) should be return true if and only if y.equals(x) return true.

iii)Transitive:For any reference values x,y and z.if x.equals(y) return true and y.equals(z) returns true,then x.equals(z) must be return true.

iv)for any non-null reference value x, x.equals(null) should return false.

19)what is casting ?

Ans)Casting is used when we want to convert one data type to another type,is nothing but a casting.

There are two types of casting are there.

i)implicit type casting.

ii)explicit type cating.

20)what is implicit type casting ?

Ans)If jvm convert a value of one datatype into it’s equalent another data type automatically by it self with our concern then we call that as implicit type casting.

Whenever it is required to assigne a smaller value to the bigger location then implicit type casting is applicable.

Ex:long l=13;

Note:whenever jvm is finding above statement ,13 number is accepted in the form of 32 bit and internally it is converted in the form of long type. This is called a implicit type casting.

21)what is explicit type casting ?

Ans)what is explicit type casting ?

Ans)if jvm converts a value one data type into it’s equalent value of another data type explicitly according to our specification then we call that explicit type casting.

Whenever it is required to assigne a bigger value to the smaller location then explicit type casting is applicable.

Ex: char ch=(char)97;

System.out.println(ch);

Output: a

22)How variables are initialized in java ?

Ans)Instance variables and static variables are always initialized with default values.default values for numeric types is ‘0’.floating point types is ‘0.0’.boolean is ‘false’ .char is ‘space’ and Object reference variable is null.

Local and block variables are NOT initialized compiler.

Ex:

Public class VariableInitialization{

Public static void main(String[] args)

{

Player player=new Player();

System.out.println(player.score);//out put is zero.

System.out.println(player.name);//out put is null

Int local;//not initialized

//System.out.println(local);//compilation error.

String value1;//not initialized

//System.out.println(value1);//compilation error.

String value2=null;

System.out.println(value2);//null-no problem

}

}

Public class Player{

Int score;

String name;

}

23)what is a nested if else ?Can you explain with an example ?

Ans)Look at the example below.The code in first if condition which is true is executed.if none of the if conditions are true,then code in else is executed.

Ex:

class IfElseTest

{

public static void main(String[] args)

{

int z=15;

if(z==10)

{

System.out.println("Z is 10");//not executed

}

else if(z==12){

System.out.println("Z is 12");//not executed

}

else if(z==15)

{

System.out.println("Z is 15");//executed.Rest of the if else are skipped.

}

else{

System.out.println("Z is Some thing else");//not executed

}

z=18;

if(z==10)

{

System.out.println("Z is 10");//not executed

}

else if(z==12){

System.out.println("Z is 12");//not executed

}

else if(z==15)

{

System.out.println("Z is 15");//not executed

}

else{

System.out.println("Z is Some thing else");// executed

}

}

}

**Arrays:**

24)What is an array ?

Ans)Arrays are collection of elements of same data type of fixed size.(Ex:new int[5]).

The size of the array and the memory space allocated for the array can not change or altered,under any case.

Arrays are internally predefined structures of JVM.whenever jvm finds squre brackets,it understand to create an Object.

25)How do you declare and create an array ?

Ans)To declare the an we have to use the following syntax.

<datatype><reference>[];

Ex: int marks;

int [] temperatures;

Declaration of an array should not include size.

//int values[5];//compilation error.

Declaring 2D array examples.

Ex: int[][] matrix;

int[] matix2[]; //in valid.

Lets now look at how to create an array.

marks=new int[5];

Declaring and creating an array in a same line.

Int marks2=new int[5];

26)Can the size of an array be changed dynamically ?

Ans)Once an array is created,its size can not be changed.

27)Can you create an array without defining size of an array ?

Ans)size of an array is mandatory to create an array.

Ex: marks=new int[];//compiler error.

28)What are the default values in an array ?

Ans)new arrays are always initialized with default values.

Ex: int marks2=new int[5];

System.out.println(marks2[0]);

Default values are:

byte,short,int,long---->0

float,double---->0.0

boolean is false.

Object🡪null.

29)How do you loop around an array using enhanced for loop ?

Ans)we can read the following example,and understanding the enhanced for loop.

Ex:int ar1[]={1,2,3,4,5,6};

for(int i:ar1)

{

System.out.println(i);

}

Note:we call such type of loops are called enhanced for loop.

30)How do you print the content of an array ?

Ans)printing 1D array

Int marks[]={25,30,35,40,45};

System.out.println(marks);//some address printed

System.out.println(Arrays.toString(marks));

Printing 2D array.

import java.util.Arrays;

class TwoDarray

{

public static void main(String[] args)

{

int[][] matrix={{1,2,3,4},{5,6,7,8}};

System.out.println(matrix);

System.out.println(Arrays.toString(matrix));

System.out.println(Arrays.deepToString(matrix));

//matrix[0] 1D Array.

System.out.println(matrix[0]);

System.out.println(Arrays.toString(matrix[0]));

}

}

31)How to compare two arrays ?

Ans)If we want compare two arrays,we have to use static method called equals available in Arrays class.if two arrays are same values with same positions then it returns true other wise it returns false.

Ex:

Int[] number1={1,2,3};

Int[] number2={4,5,6};

System.out.println(Arrays.equals(number1,number2));//false

Int[] number3={1,2,3};

System.out.println(Arrays.equals(number1,number3));//true

32)How do you sort an array ?

Ans)if we want sort an array, we can use utility method called sort method.given by Arrays class.

Ex:

Int rollnos[]={12,5,7,9};

Arrays.sort(rollnos);

System.out.println(Arrays.toString(rollnos));

o/p:[5,7,9,12]

33)what is the diff between length variable and length function ?

Ans)The difference is length variable is finding the total size of an array,where as length function is finding the particular string size.

Ex:

String[] s={“oracle”,”java”,”cpp”};

System.out.println(s.length);//3

System.out.println(s[0].length());//6

34)What is difference between ArrayIndexOutfOBounds and ArrayStoreException?   
Ans)ArrayIndexOutOfBoundsException comes when your code tries to access an invalid index for a given array e.g. negative index or higher index than length - 1. While, ArrayStoreException comes when you have stored an element of type other than type of array, as shown in below example**.**  
EX:

class Demo

{

public static void main(String[] args)

{

int[] primes = new int[10];

//primes[0] = "a"; //compile time error

Object[] names = new String[3];

names[0] = new Integer(1); // ArrayStoreException at runtime

}

}

35) **Difference between a[] and []a in Java?**  
Ans)You can declare an array in Java by either prefixing or suffixing[] with variable. There is not much difference between them if you are not creating more than one variable in one line, but if you do then it creates different types of variables, as shown in following example :

Ex:

Int a[],b;//a is int array and b is integer variable.

Int[] c,d;//here c and d both are int array.

Enum:

36)what is an Enum ?

Ans)Enum allowes Specifying a list of values for a Type.Consider the example below. It declares an enum Seasion with 4 possible values.

Ex:

enum Seasion{

WINTER,SPRING,SUMMER,FALL

};

Note:enum is the key word.

37)How do you create an enum from String value ?

Ans)if we want create an enum from string value we have to use valueOf() function.

Ex:

Seasion seasion=Seasion.valueOf(“FALL”);

To finding string value from enum,we have to use name() method.

Ex:System.out.println(seasion.name());

o/p:FALL

38)What is an Ordinal?

Ans)java assigns default ordinals to an enum in order.However,it is not recommended to use ordinals to perform logic.

Ex:

System.out.println(Seasion.WINTER.ordinal());//0

System.out.println(Seasion.SPRING.ordinal());//1

System.out.println(Seasion.SUMMER.ordinal());//2

System.out.println(Seasion.FALL.ordinal());//3

39)How do you compare two Enums ?

Ans)If we want compare two enums,we have to use ==,or we can use equals method.

Ex:

Seasion seasion1=Seasion.FALL;

Seasion seasion2=Seasion.FALL;

System.out.println(seasion1==seasion2);//true

System.out.println(seasion1.equals(seasion2));//true

40)Can you switch statement around an enum?

Ans)Yes, we can use switch statement around enum.

Ex: enum Season

{

WINTER,SPRING,SUMMER,FALL

};

public class Demo1

{

public static void main(String[] args)

{

Season s;

s=Season.FALL;

switch(s)

{

case WINTER:

System.out.println("5 degrees");

break;

case SPRING:

System.out.println("10 degrees");

break;

case FALL:

System.out.println("20 degrees");

break;

case SUMMER:

System.out.println("30 degrees");

break;

default:

System.out.println("25 degrees");

break;

}

}

};

41)what is the super class of every class in java ?

Ans)Every class in java is a sub class of the class Object.when we create a class we inherit all the methods and properties of Object class.Let’s look at asimple example.

String str=”Testing”;

System.out.println(str.toString());

System.out.println(str.hashCode());

In the above example,toString() and hashCode() for String class are inherited from Object class.

In entire java the root class is Object of Object class.

42)can super class reference variable can hold an object of sub class ?

Ans)Yes.look at the following example.

Public class Actor{

Public void act()

{

System.out.println(“Act”);

}

Public class Hero extends Actor{

Public void fight()

{

System.out.println(“fight”);

}

}

Public class Comedian extends Actor{

Public void performComedy()

{

System.out.println(“comedy”);

}

Actor actor1=new Comedian();

Actor actor2=new Hero();

43)is multiple inheritance allowed in java ?

Ans)Java doesn’t support multiple inheritance.because multiple classes are coming to the sub class with same prearity.

Ex:class Dog extends Animal,Pet{

}

44)what is a polymorphism and how many typs?

Ans)poly means many and marphism means forms or functionalities.

The concept of defining multiple functionalities with the same name associated with the same oject is known as poly marphism.

There are two types of polymorphism are there.

i)Static or compile time polymorphism.

ii)Dynamic or run time polymorphism

45)what is a static polymorphism ?

Ans)The cocept of defining multiple functions with the same class but with the diff parameters of diff data types is known as static polymarpism.

Ex:

Class Poly1{

Void funA()

{

System.out.println(“funA()”);

}

Void funA(int x)

{

System.out.println(“funA(int x)”);

}

Void funA(int x,int y)

{

System.out.println(“funA(int x,int y)”);

}

Public static void main(String[] args)

{

Poly1 p=new Poly1();

Int x=6;

p.funA(6);

}

}

46)what is the diff between static polymorphism and dynamic poly marphism ?

Ans)with respct to static poly marphism,out of multiple functions with the same name which function as to be executed would be desided at the time of compilation.where as dynamic polymorphism which function is executed is desided at run time. But not compilation time.

47)can you give the example of dynamic polymorphism ?

Ans)the following example is represented as dynamic poly marphism.

Class P

{

Void funP(int x)

{

Int sum=0;

For(int i=0;i<=x;i++)

{

Sum=sum+i;

}

System.out.println(“The sum is:”+sum);

}

}

Class Q extends P

{

Void funP(int x)

{

Int sq=x\*x;

System.out.println(“Sq of x numbers:”+sq);

}

}

Class R extends P

{

Void funP(int x)

{

Int fact=1;

For(int i=1;i<=x;i++)

{

Fact=fact\*i;

}

System.out.println(“factorial of given number:”+fact);

}

}

Class DynaTest{

p.s.v.main()

{

Int x=Integer.parseInt(args[0].trim());

Int y=Integer.parseInt(args[1].trim());

P p1=null;

If(y>0&&y<=10)

{

P1=new P();

}

If(y>10&&y<=20)

{

P1=new Q();

}

If(y>20)

{

P1=new R();

}

P1.funP(x);

}

}

48)what is the use of instanceof operator in java ?

Ans) The **java instanceof operator** is used to test whether the object is an instance of the specified type (class or subclass or interface).

The instanceof in java is also known as type *comparison operator*because it compares the instance with type. It returns either true or false. If we apply the instanceof operator with any variable that has null value, it returns false.

Ex: public class MainClass {

public static void main(String[] a) {

String s = "Hello";

if (s instanceof java.lang.String) {

System.out.println("true");

} else {

System.out.println("false");

}

}

}

49)what is an abstract class ?

Ans)an abstract class is partially implemented and partially unimplemented structure.which containes method definations with out bodies and also method definations along with it’s bodies,they are known as abstract classes.

🡪object can not be created directly by using new operator.

Ex:abstract class Abs1{

int i,j;

abstract public void funA();

void funB()

{

System.out.println(“inside fun2() of class Abs1”);

}

}

50)How do you define an abstract method ?

Ans)An abstract methods does not containe body.An abstract methods does not have any implementation.the implementation of abstract methods shoud be provided in sub class.

Ex:

abstract void funA();

abstract void funB();

🡪abstract methods can be declared only in abstract class.in the example below,funA() gives a compiler error NormalClass is not abstract.

Ex:

Class NormalClass{

abstract void funA();

}//compilation error

51)What is a coupling ?

Ans)Whenever we want communicate one object to another object between two components(classes) there is a some collaboration is done. this collaboration is called coupling.

🡪Coupling mainly devided into two types.

i)Tightly coupled ii)loosely coupled.

52)what is a tightly coupled and give an example?

**Tight-Coupling:-**

1.    While creating complex application in java, the logic of one class will call the logic of another class to provide same service to the clients.  
  
2.    If one class calling another class logic then it is called collaboration.  
  
3.    When one class is collaborating with another class then there exists tight-coupling between the two classes.  
  
4.    If one class wants to call the logic of a second class then they first class need an object of second class it means the first class create an object of second class.  
  
5.    For example, if we have two classes called traveller and car, traveller class is calling logic of car class; in this case traveller class creates an object of car class.  
  
6.      In the above traveller class and car classes, car class object of dependency for traveller object.  
  
**Example:-**

|  |  |
| --- | --- |
| Picture | Picture |

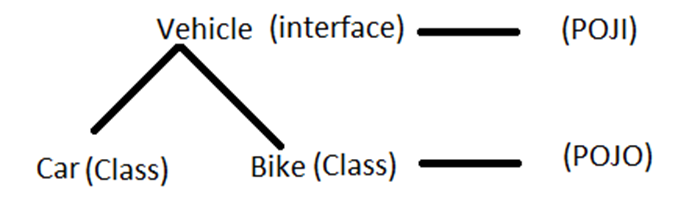
7.    In the above example traveller object is tightly coupled with car object because in place car object  if you want to use bike object then, we need to make changes in Traveller class  
  
    Example :-

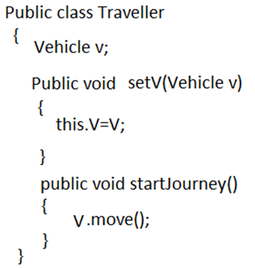
|  |  |
| --- | --- |
| Picture | Picture |

53)what is a loosely coupled and give an example?

**Loose-Coupling:-**

.  1.  In Loose-Coupling, when one object is depending on another class object, some external entity will provide that dependency object to the main object that external object we call as a Container.  
  
2.    In order to get loose-coupling between objects the following two rules are required  
  
1.    The classes should follow POJI/POJO model.  
  
2.    Apply dependency injection mechanism.  
  
 For example:-





3.    In the above traveler class, an external entity injects either car (or) Bike object.  
  
4.    In traveler, these are no changes required we are shifting the dependency from car to a Bike.  
  
5.    In the above traveler class, we are token vehicle reference, so that an external object (Container) can injects either car object (or) Bike object, depends on requirement if a traveler.  
  
6.    In spring frame work, spring container follows dependency injection mechanism and injects the dependency objects required for a main object.  
  
7.    Spring frame work is much success because of one of the main reason is it promotes Loose-Coupling between the objects.

54)what is a encapsulation ?

Ans)The concept of binding the data along with it’s related and corresponding functions is known as encapsulation.

🡪encapsulation is hiding the implementation of a class behind a well defined interface.

Approach-1

In this approach we create a public variable score.The main method directly accesses the score variable,update it.

Ex:

Public class CricketScore{

Public int score;

}

CricketScore scorer=new CricketScore();

Scorer. Scorer= scorer. Scorer+4;

Approach-2

In this approach,we make score as private and access value through get and set methods.however,the logic of adding 4 to the score is performed in the main method.

Ex:

Public class CricketScore{

private int score;

public int getScore(){

return score;

}

Public void setScore(int score)

{

This.score=score;

}

}

class CricketScore1

{

public static void main(String[] args)

{

CricketScore scorer=new CricketScore();

int score=scorer.getScore();

scorer.setScore(score+4);

System.out.println(scorer.getScore());

}

}

55)what is the method Overloading ?

Ans)The concept of method Overloading is same function name with different parameter is called as method Overloading.

Ex:

Class Foo{

Public void doIt(int number)

{

}

Public void doIt(String name)

{

}

}

56)what is method Overriding ?

Ans)creating the sub class method with same signature as that of a method in super class is called method Overriding.

Ex:

Public class MethodOverriding

{

Public void funA()

{

System.out.println(“in side funA() of class MethodOverriding”);

}

}

Class MethodOverriding1 extends MethodOverriding

{

Public void funA()

{

System.out.println(“inside funA() of class MethodOverriding1”);

}

}

When we compile and execute the above programe the out put is: inside funA() of class MethodOverriding1

57)what is the inner class ?

Ans)

Whenever we declared one class inside another class is nothing but a inner class.consider the following example.

Class OuterClass{

Public class InnerClass{

}

}

58)what is the static inner class ?

Ans)a class declared directly inside another class and declared as static also called as nested class.consider the following example

Class OuterClass{

Public static class InnerClass{

}

}

59)what is the ananmus inner class ?

Ans)what is the class name we don’t know exactly such type of classes we call the annomus inner classes.

Ex:

Connection con=DriverManager.getConnection(“jdbc:oracle:thin:@localhost:1521:XE”,”uname”,”password”);

Here DriverManager.getConnection just returns the one object,but what is the class name we don’t know such type of classes we call them as ananmus inner classes.

60)can you create an inner class inside a method ?

Ans)Yes.an inner class can be declared directly inside a method.consider the following example

Class OuterClass{

Public void exampleMethod(){

Class MethodLocalInnerClass{

}

}

}

Constructors:

61)what is a default Constructor ?

Ans)Default constructor is the constructor that is provided by the compiler.it has no arguments.in the example below,there are no constructors defined in the Animal class.Compiler provides us with a default constructor,which helps us create an instance of animal class.

Ex:

Public class Animal{

String name;

Public static void main(String[] args)

{

Animal animal=new Animal();

}

}

Here compiler is providing the default constructor.

62)How do you call a super class constructor from a constructor ?

Ans)if we want call super class constructors,explicitly we have to use super key ward.

Ex:

class A

{

A()

{

System.out.println("i am the a class");

}

};

class SuperDemo extends A

{

SuperDemo()

{

super();

}

public static void main(String[] args)

{

SuperDemo d=new SuperDemo();

System.out.println("Hello World!");

}

}

63)How to call the super class parameterized constructor ?

Ans)using super() method and passing the some argument,then we can call the super class constructors.

Ex:

class A

{

A(int x)

{

System.out.println("i am the a class");

}

};

class SuperDemo extends A

{

SuperDemo()

{

super(9);

}

public static void main(String[] args)

{

SuperDemo d=new SuperDemo();

System.out.println("Hello World!");

}

}

64)can a constructor be called directly from a method ?

Ans)A constructor can not be explicitly called from any method except another constructor.

Ex:

Class Animal{

Public Animal(){

}

Public void method()

{

Animal();//compiler error.

}

}

65)Is a super class constructor called even when there is no explicit call from a sub class constructor ?

Ans)if a super class constructor is not explicitly called from a sub class constructor.super class constructor is automatically invoked from a sub class constructor.

Ex:

Consider the example bellow.

Class Animal{

Public Animal(){

System.out.println(“animal constructor”);

}

}

Class Dog extends Animal{

Public Dog(){

System.out.println(“Dog constructor”);

}

Class Cat extends Dog{

Public Cat(){

System.out.println(“cat constructor”);

}

}

Public class ConstrucorExample{

p.s.v.main(String[] args){

Cat c=new Cat();

}

}

Interfaces:

66)what is an interface ?

Ans)interface is an structor,it containes all abstract un implemented methods.but not bodies of methods.

Or it containes fully un implemented methods.

67)How do you define an Interface ?

Ans)an interface is declared by using the keyword interface.for example,

Public abstract interface Flyable{

Public abstract void fly();

}

//public abstract are not necessary.

68)How do you implement an interface ?

Ans)we can define a class implementing the interface by using the implements keyword.

Ex:

Public abstract interface Flyable{

Public abstract void fly();

}

Public class Aeroplane implements Flyable{

Public void fly(){

System.out.println(“aeroplane is flying”);

}

}

69)Can you tell a little bit more about interfaces?

Ans)variables in an interface are always public,static,final.variables in an interface can not be declared private.

Ex:

interface ExampleInterface{

int value1=10;

public int value2=20;

public static int value3=30;

public static final int value4=40;

}

Interface methods are by default public and abstract.A concreate method con not be created in an interface.

70)Can you extend an interface ?

Ans)an interface can extend another interface.consider an example

interface SubInterface extends ExampleInterface1{

void method3();

}

71)can a class extend multiple interfaces ?

Ans)A class can implement multiple interfaces.it should implement all the methods declared in all interfaces beging implemented.

Ex:

interface Idemo

{

void funA();

void funB();

}

interface Idemo1

{

void funC();

void funD();

}

class Itest implements Idemo,Idemo1

{

public void funA()

{

System.out.println("inside funA() of class Itest");

}

public void funB()

{

System.out.println("inside funB() of class Itest");

}

public void funC()

{

System.out.println("inside funC() of class Itest");

}

public void funD()

{

System.out.println("inside funD() of class Itest");

}

public static void main(String[] args)

{

Idemo d1=new Itest();

d1.funA();

d1.funB();

//d1.funC();

//d1.funD();

}

}

Access Modifiers:

72)what is default class modifier ?

Ans)A class is called a default Class is when there is no access modifier specified on a class.

Default classes are visible inside the same package only.

Default access is also called package access.

Ex:

Package com.durga;

Class DefaultAccessClass

{

}

73)Another class in same package:Has access to default class ?

Ans)Yes, we can access.

Ex:

Package com.durga;

Class AnotherClassInSamePackage{

DefaultAccessClass defaultAccess;

}

74)class in different package:No access to default class.?

Ans)It is not passible to access the default class other packages.

Ex)package com.durga.tech;

Public class ClassInDiffrentPack{

DefaultPackage defaultpack;//compilation error

}

75)what are the diff method access specifier ?

Ans)In java there are four types access specifiers are there.

i)private

ii)public

iii)protected

iv)default.

76)what is a private access specifier and give the scope ?

Ans)private is the access specifier or key ward.

Scope of the private access specifier is ,variables and methods can be accessed only in the class they are declared.

Ex:public class AccessDemo{

Private int i,j;

}

These I and j doesn’t allowed out side of that class under any case.

77)what is the default access specifier and give an example ?

Ans)any member of the class define with out any access specifier,would be considered as default,thus any thing define with out access specifier we consider as default.

Ex:

Public class AccessDemo{

Int i,j;

}

Here I and j are default access specifier.

78) what is the protected access specifier and give an example ?

Ans)protected will act as public with in the same package and protected will act as private outside package.but protected will act as public out side the package only with respect to the sub class objects.

Ex:

Package p1

Public class Access

{

Int i,j;

Protected int k;

}

Package p2;

Import p1.Access;

Public class Access1 extends Access

{

p.s.v.main(){

Access1 a=new Access1();

S.o.p(a.k);

}

}

79)what is the public access specifier and give an example ?

Ans)public members of a class could be available to any thing outside of the class and package.

In order to make the members of a class available to any thing and any where out side the class and package,we mention public.

Ex:

Public class Access

{

Public int k,l;

}

These variables are available any where.

80)what is the use of a final modifier on a class ?

Ans)Final class cannot be extended.Example of final class in java is the String class.final class is used,if we want prevents re-use of the class.

Ex:

Final public class FinalClass{

}

Class ExtendingFinal extends FinalClass{

}//getting compilation error.

81)what is the use of a final mpdifier on a method ?

Ans)Final methods can not be Overriden.consider the following example.

Ex:

Public class FinalMethod{

final void finalMethod(){

}

}

Public class Subclass extends FinalMethod{

final void finalMethod(){//compilation error.

}

}

82)what is a final variable ?

Ans)Once initialized,the value of a final variable can not be changed.

Ex:

Class FinalVariable{

final int finalValue=5;

//finalValue=10;//compilation error.

}

83)what is the final argument ?

Ans)final arguments value cannot be modified.consider the following example.

Ex:

class FinalArgument{

void testMethod(final int finalargument)

{

System.out.println(finalargument);

//finalargument=6;//compilation error.

}

public static void main(String[] args)

{

FinalArgument f=new FinalArgument();

f.testMethod(7);

}

}

84)what is a static variable ?

Ans)static variables and methods are class level variables and methods.there is only one copy of the static variable for the enter class.each instance of the class will not have a unique copy of a static variable.

Ex:

Public class Cricketer{

Private static int count;

Public Cricketer(){

Count++;

}

Static int getCount(){

Return count;

}

Public static void main(String[] args){

Cricketer c1=new Cricketer();

Cricketer c2=new Cricketer();

Cricketer c3=new Cricketer();

Cricketer c4=new Cricketer();

System.out.println(Cricketer.getCount());

}

}

85)what are variable arguments or varargs ?

Ans)Variable arguments allow calling a method with different number of parameters.consider the following example.the sum method can be called with 1 int parameter or 2 int parameters or more int parameters.

Ex:

class VarArg

{

public int sum(int...numbers)

{

int sum=0;

for(int number:numbers)

{

sum+=number;

}

return sum;

}

public static void main(String[] args)

{

VarArg v=new VarArg();

System.out.println(v.sum(1,2,3));

System.out.println(v.sum(1,6,3));

}

}

86)what are asserts used for ?

Ans)Assertion are introduced in java 1.4.They enable you to validate assumptions.if assert failes,AssertionError is thrown.

Ex:

Private int computeSimpleInterest(int principal,float interest,int year)

{

Assert(principal>0)

Return 100;

}

87)What is a garbage collection ?

Ans)Garbage collection is a name given to automatic memory management in java.Aim of Garbage collection is to keep as much of heap available for the programe as possible.JVM removes objects on the heap which is no longer have reference from the heap.

88)can you garbage ollection with an example ?

Ans)let’s see the following example.

Void method()

{

Calendar calender=new GregorianCalender(2000,10,30);

System.out.println(calender);

}

An Object of the class GregorianCalender is created on the heap by the first line of the function with one reference variable calendar

After the function ends execution,the refence variable calendar is no longer valid.hence there is no references to the object created in the method.

89)When garbage collection run ?

Ans)the garbage collection runs,the following two switch vations.

i)when available memory on the heap is low.

ii)when cpu is free.

90)what are best practices on garbage collection ?

Ans)programmatically,we can request JVM to run Garbage collection by calling System.gc() method.

JVM might throw an OutOfMemoryException when memory is full and no objects on the heap are eligible for garbage collection.

91)what are Initialization blocks ?

Ans)There are two initialization blocks

i)static initialize:code that runs when a class is loaded.

ii)Instance Initializer:code that runs when a new object is created.

92)what is a static initializer ?

Ans)look at the following example.

class InitializerExample

{

static int count;

int i;

static{

System.out.println("static initializer");

//i=6;//compilation error

System.out.println("count when static initializer run is"+count);

}

public static void main(String[] args)

{

InitializerExample e1=new InitializerExample();

InitializerExample e2=new InitializerExample();

InitializerExample e3=new InitializerExample();

System.out.println("Hello World!");

}

}

93)what is an instance initialize blocks ?

Ans)look at the following example ?

class InitializerExample

{

static int count;

int i;

{

System.out.println("Instanc initializer");

i=6;//compilation error

count= count+1;

System.out.println("count when instace initializer run is"+count);

}

public static void main(String[] args)

{

InitializerExample e1=new InitializerExample();

InitializerExample e2=new InitializerExample();

InitializerExample e3=new InitializerExample();

System.out.println("Hello World!");

}

}

94)what is reflection ?

Ans)Reflection is an mirror image,which is used for to finding the internal details of classes and interface.for example we can find out what is the super class object of all the classes and how many interfaces are implemented by the particular class. Such type of information we can find out by using reflection api.

95)Give the an example of,how to find out super class of all the classes ?

Ans)

import java.lang.reflect.\*;

class Test1

{

public static void main(String[] args)throws Exception

{

Class c=Class.forName("java.lang.Integer");

Class sc=c.getSuperclass();

String str=sc.getName();

System.out.println("Hello World!"+str);

}

}

96)Give an example of how to finding,how many interfaces are implemented by particular class ?

Ans)

import java.lang.reflect.\*;

class Test2

{

public static void main(String[] args)throws Exception

{

Class c=Class.forName(args[0]);

Class itf[]=c.getInterfaces();

for(int i=0;i<itf.length;i++)

{

String str=itf[i].getName();

System.out.println(str);

}

System.out.println("Hello World!");

}

}

97)what are regular Expressions ?

Ans)Regular expressions make parsing,scanning and splitting a string very easy.we will first look at how you can evaluate a regular expressions in java.using patter,Matcher and Scanner classes.we will then look into how to write regular expressions.

98)what is Tokenizing ?

Ans)Tokenizing meansn splitting a string into several sub strings based on delimiters.for example delimiter; splits the string ac;bd;def;e into four sub strings ac,bd,def, and e.

99)can you give an example of tokenizing?

Ex:

import java.util.\*;

class StringToken

{

private static void tokenize(String string,String regexp)

{

String[] tokens=string.split(regexp);

System.out.println(Arrays.toString(tokens));

}

public static void main(String[] args)

{

tokenize("ac;bd;def;e",";");

System.out.println("Hello World!");

}

}

100)How you can tokenize using scanner class ?

Ex:

import java.util.\*;

class StringTokenUsingScanner

{

private static void tokenizeUsingScanner(String string,String regexp)

{

Scanner scanner=new Scanner(string);

scanner.useDelimiter(regexp);

List<String> matches=new ArrayList<String>();

while(scanner.hasNext())

{

matches.add(scanner.next());

}

System.out.println(matches);

}

public static void main(String[] args)

{

tokenizeUsingScanner("ac;bd;def;e",";");

}

}

101)How do you add hours to a date Object ?

Ans)consider the following example,

import java.util.\*;

class PrintDate

{

public static void main(String[] args)

{

Date date=new Date();

date.setTime(date.getTime()+6\*60\*60\*1000);

System.out.println(date);

System.out.println("Hello World!");

}

}

102)How do you format Date Objects ?

Ans)Formatting Dates is done by using DateFormat class .let’s look at a few example

import java.util.\*;

import java.text.DateFormat;

class PrintDate

{

public static void main(String[] args)

{

Date date=new Date();

date.setTime(date.getTime()+6\*60\*60\*1000);

System.out.println(DateFormat.getInstance().format(date));

System.out.println("Hello World!");

}

}

103)what is the use of calendar class in java ?

Ans)Calender class is used in java to manipulate dates.Calender class provides easy ways to add or reduce days,months or years from a date. It also provide lot of details about a date(which day of the year ?)which week of the year ?...etc)

104)How do you get Instance of Calendar class in java ?

Ans)Calender class cannot be created by using new Operator.The best way to get an instance of Calendar class is by using getInstance().This is static method in a Calendar class.

Ex)

Calendar calendar=Calendar.getInstance();

105)Can you explain some of the important methods in calendar class ?

Ans)setting day,month or year on a calendar object is simple.call the set method with appropriate constant for Day,Month or Year.

Ex:

calendar.set(Calendar.DATE,24);

calendar.set(Calendar.MONTH,8);

calendar.set(Calendar.DATE,2010);

if we want get Calender related information we have to use get() method.

System.out.println(calendar.get(Calendar.YEAR));

System.out.println(calendar.get(Calendar.MONTH));

System.out.println(calendar.get(Calendar.DATE));

System.out.println(calendar.get(Calendar.WEEK\_OF\_MONTH));

System.out.println(calendar.get(Calendar.WEEK\_OF\_YEAR));

System.out.println(calendar.get(Calendar.DAY\_OF\_YEAR));

106)Why do we need collections in java ?

Ans)Array is collection of elements with same data type and fixed size. Where as collections are dynamically we can increase at run time

🡪Arrays are supports both Objects and primitive data types also.where as collections are support only Objects but not primitives.

🡪Arrays are predefined structures of JVM.where as in collections many libraries are available.

107)what are the important methods that are declared in the Collection interface ?

Ans)Most important methods declared in the collection interface are the methods to add and remove an element. add method allows adding an element to a collection and delete method allowes deleting an element from a collection.

Ex:

interface Collection<E> extends Iterable<E>

{

boolean add(E paramE);

boolean remove(Object paramObject);

int size();

boolean isEmpty();

void clear();

--------------------------

}

108) can you explain briefly about the List Interface ?

Ans)List interface extends Collection interface.List interface is available java.util.\*; package.list collectios are allowed duplicate elements and it will displayed Same order.

Ex:

Class ListDemo{

p.s.v.main(){

List list=new ArrayList();

List.add(“ravi”);

List .add(“soono”);

S.o.p(list);

}

}

109) can you explain briefly about the Set Interface ?

Ans)Set interface extends Collection interface.Set interface is available java.util.\*; package.Set collectios are not allowed duplicate elements and it will displayed Same order my be or may not.

Ex:

Class ListDemo{

p.s.v.main(){

List list=new ArrayList();

List.add(“ravi”);

List .add(“soono”);

S.o.p(list);

}

}

110) can you explain briefly about the Map Interface ?

Ans)Map is an interface available in java.util.\*;package,Map interface does not extends Collection interface.Map collection are maintaines collection of elements in an unordered way and Map collections doesn’t allowed duplicate keys.by using map collections we can perform the operations.

i)adding an element or value along with it’s key.

ii)Accessing an element by mentioned it’s key.

iii)Updating an element by mentioned it’s key.

iv)Deleting an element by mentioned it’s key.

111)Give the an example of Map collections ?

Ans)import java.util.\*;

Class Hmap

{

p.s.v.main(){

Map<Integer,String> m=new HashMap<Integer,String>();

m.put(1234,”murali”);

m.put(2345,”raj”);

m.put(3242,”nani”);

Set<Integer> s1=m.key();

Iterator<Integer> i1=s1.iterator();

While(i1.hasNext())

{

Int x=i1.next();

String v=m.get(x);

S.o.p(x+” ”+v);

}

}

112)what is the difference between Set and SortedSet ?

Ans)SortedSet interface extends the Set interface.Both Set and SortedSet dom not allow duplicate elements.Main difference between Set and SortedSet is- an implementation of SortedSet interface maintains its elements in a sorted Order. Set interface does not guarantee any order.

113)Give the an example of Set and SortedSet ?

Ans)Consider the following example.

import java.util.\*;

public class SortedSetTest {

public static void main(String[] args) {

// Create the sorted set

SortedSet set = new TreeSet();

// Add elements to the set

set.add("30");

set.add("24");

set.add("40");

// Iterating over the elements in the set

Iterator it = set.iterator();

while (it.hasNext()) {

// Get element

Object element = it.next();

System.out.println(element.toString());

}

}

}

import java.util.\*;

public class SetTest {

public static void main(String[] args) {

// Create the sorted set

Set set = new HashSet();

// Add elements to the set

set.add("30");

set.add("24");

set.add("40");

// Iterating over the elements in the set

Iterator it = set.iterator();

while (it.hasNext()) {

// Get element

Object element = it.next();

System.out.println(element.toString());

}

}

}

114) what is the difference between Map and SortedMap ?

Ans)SortedMap interface extends Map interface.in addition,an implementation of SortedMap interface maintaines keys in a sorted order where as Map doesn’t mataines the keys any order.

115)Give the an example of Map and SortedMap?

Ans)

import java.util.Comparator;

import java.util.HashMap;

import java.util.Map;

import java.util.TreeMap;

import java.util.\*;

public class SortMapExample {

public static void main(String[] args) {

//creating unsorted map of employee id as a key and employee name as a value

Map unsortMap = new HashMap();

unsortMap.put(10, "Ashraf");

unsortMap.put(5, "Sara");

unsortMap.put(6, "Mohamed");

unsortMap.put(20, "Esraa");

unsortMap.put(1, "Bahaa");

unsortMap.put(7, "Dalia");

unsortMap.put(8, "Amira");

unsortMap.put(99, "Ahmed");

unsortMap.put(50, "Sama");

unsortMap.put(2, "Nada");

unsortMap.put(9, "Osama");

System.out.println(unsortMap);

}

}

import java.util.Comparator;

import java.util.HashMap;

import java.util.Map;

import java.util.TreeMap;

import java.util.\*;

public class SortMapExample {

public static void main(String[] args) {

//creating unsorted map of employee id as a key and employee name as a value

SortedMap sortMap = new TreeMap();

sortMap.put(10, "Ashraf");

sortMap.put(5, "Sara");

sortMap.put(6, "Mohamed");

sortMap.put(20, "Esraa");

sortMap.put(1, "Bahaa");

sortMap.put(7, "Dalia");

sortMap.put(8, "Amira");

sortMap.put(99, "Ahmed");

sortMap.put(50, "Sama");

sortMap.put(2, "Nada");

sortMap.put(9, "Osama");

System.out.println(sortMap);

}

}

116)Explain briefly about Queue Interface ?

Ans)Queue interface extends Collection interface.Queue interface is typically used for implementation of holding elements in order some processing.

Queue interface offers methods peek() and poll() which get the element at head of the queue.the difference is that poll() method removes head of the queue.peek() method doesn’t remove the head of the queue.

Ex:

interface Queue extends Collection{

E poll();

Epeek();

}

117)Explain briefly about Iterator ?

Ans)Iterator is an interface.using Iterator interface we can iterate the collection elements.all the collections define a method iterator() that gets an iterator of the collection.

hasNext() checks if there is another element in the collection being iterated.next() gets the next element.

Ex:

Public interface Iterator{

Boolean hasNext();

E next();

}

118)Explain about ArrayList and Example ?

Ans)ArrayList is an implemented class of List interface.So arraylist stores the elements in insertion order.

Elements can be inserted and removed from arraylist based on their positions.

For example:

List<Integer> list=new ArrayList<Integer>();

List.add(6);

List.add(5);

119)Can an ArrayList have Duplicate elements ?

Ans)ArrayList can have duplicates.

Ex:

List<String> list=new ArrayList<String>();

List.add(“Sachin”);

List.add(“Dravid”);

List.add(“Ganguly”);

List.add(“Sachin”);

120)How do you iterate around an ArrayList using Iterator ?

Ans)below example shows how to iterate around an arraylist.

Ex:

List<String> list=new ArrayList<String>();

While(list.hasNext())

{

String str=list.next();

S.o.p(str);

}

}

121)How do you sort an ArrayList ?

Ans)consider the below example to sort an ArrayList.It uses the Collections.sort method.

import java.util.\*;

class ArrayListDemo

{

public static void main(String[] args)

{

List<String> numbers=new ArrayList<String>();

numbers.add("two");

numbers.add("three");

numbers.add("five");

numbers.add("one");

numbers.add("six");

System.out.println(numbers);

Collections.sort(numbers);

System.out.println(numbers);

System.out.println("Hello World!");

}

}

122)How do you sort elements in an ArrayList using Comparable interface?

Ans)Consider the following example,

import java.util.\*;

class Cricketer implements Comparable<Cricketer>

{

int runs;

String name;

public Cricketer(String name,int runs)

{

super();

this.name=name;

this.runs=runs;

}

public String toString(){

return name+""+runs;

}

public int compareTo(Cricketer that)

{

if(this.runs>that.runs)

{

return 1;

}

if(this.runs<that.runs)

{

return -1;

}

return 0;

}

}

class ComparableTest

{

public static void main(String[] args)

{

List<Cricketer> cricketers=new ArrayList<Cricketer>();

cricketers.add(new Cricketer("Bradman",9996));

cricketers.add(new Cricketer("Sachin",14000));

cricketers.add(new Cricketer("Dravid",12000));

cricketers.add(new Cricketer("ponting",11000));

System.out.println(cricketers);

Collections.sort(cricketers);

System.out.println(cricketers);

}

}

123) How do you sort elements in an ArrayList using Comparator interface?

Ans)Consider the following example,

import java.util.\*;

class Cricketer implements Comparator<Cricketer>

{

int runs;

String name;

public Cricketer(String name,int runs)

{

super();

this.name=name;

this.runs=runs;

}

public String toString(){

return name+""+runs;

}

public int compare(Cricketer cricketer1,Cricketer cricketer2)

{

if(cricketer1.runs>cricketer2.runs)

{

return -1;

}

if(cricketer1.runs<cricketer2.runs)

{

return 1;

}

return 0;

}

}

class ComparableTest

{

public static void main(String[] args)

{

List<Cricketer> cricketers=new ArrayList<Cricketer>();

cricketers.add(new Cricketer("Bradman",9996));

cricketers.add(new Cricketer("Sachin",14000));

cricketers.add(new Cricketer("Dravid",12000));

cricketers.add(new Cricketer("ponting",11000));

System.out.println(cricketers);

Collections.sort(cricketers,new Cricketer());

System.out.println(cricketers);

}

}

124)How do you convert List to Array?

Ans)we can use toArray(String) function.

List<String> numbers1=new ArrayList<String>();

numbers1.add(“one”);

numbers1.add(“two”);

numbers1.add(“three”);

numbers1.add(“four”);

String[] numbers1Array=new String[numbers1.size()];

numbers1Array=numbers1.toArray(numbers1Array);

S.o.p(Arrays.toString(numbers1Array));

125)How do you convert an Array to List ?

Ans)String values[]={“value1”,”value2”,”value3”};

List<String> valuesList=Arrays.asList(values);

System.out.println(valueslist);

126)what is Vector class ? How is it different from an ArrayList?

Ans)Vector has the same operations as an ArrayList.However,all methods in Vector are Synchronized.So we can use Vector,if we share a list between two threads.then we can synchronized.

127)what is LinkedList ? what interfaces does it implement ? how is it different from an ArrayList ?

Ans)LinkedList extends List and Queue.LinkedList has the same operations as an ArrayList.

ArrayList uses an Array kind of structure to store elements.However, search of an ArrayList is faster than LinkedList.

LinkedList uses a linked representation.each objects holds a link to the next element,hence insertion and deletion are faster than ArrayList.But searching is slower.

128)Can you give examples of classes that implement the Set interface?

Ans)The implemented classes of Set interface is HashSet ,LinkedHashSet and TreeSet.

129)what is a HashSet?

Ans)HashSet implements Set interface.So HashSet does not allow duplicates.However,HashSet does not support ordering.

Ex:

Set<String> hashset=new HashSet<String>();

hashset.add(“sachin”);

hashset.add(“dravid”);

hashset.add(“sachin”);

System.out.println(hashset);//duplicates doesn’t allowed.

130)what is a LinkedHashSet? How is different from a HashSet?

Ans) LinkedHashSet implements Set interface and similar operations to HashSet.but the difference is LinkedHashSet maintaine the insertion order.that means in which order we insert the elements same order is displayed.where as HashSet is prints diff order.

131)what is a TreeSet? How is diff from a HashSet?

Ans)TreeSet implements Set,SortedSet and NavigableSet interfaces.TreeSet is similar to HashSet.except that it stores elements in sorted Order.

Ex:

Set<String> treeset=new TreeSet<String>();

treeset.add(“Sachin”);

treeset.add(“Dravid”);

treeset.add(“Ganguly”);

S.o.p(treeset);

132)Can you give examples of implementations of NavigableSet?

Ans)NavigableSet interface was implements TreeSet class.and values are sorted.

Ex:

TreeSet<Integer> numbertreeset=new TreeSet<Integer>();

numbertreeset.add(55);

numbertreeset.add(25);

numbertreeset.add(35);

numbertreeset.add(5);

numbertreeset.add(45);

NavigableSet interface has following method.

Lower method finds the highest element lower than specified element.floor method finds the highest element lower than or equal to specified element.

Ex:

S.o.p(numberstreeset.lower(25));//5

S.o.p(numberstreeset.floor(25));//25

S.o.p(numberstreeset.higher(25));//35

S.o.p(numberstreeset.ceiling(25));//25

133)what are the diff implementations of a Map interface?

Ans)The important implemented classes of Map interface is HashMap and TreeMap.

134)What is HashMap ?

Ans)HashMap is an implemented class of Map interface.and HashMap is storing the values in the form of key,values pairs.

Map<String,Integer> hashmap=new HashMap<String,Integer>();

hashmap.put(“sachin”,2000);

hashmap.put(“dravid”,1000);

hashmap.put(“lara”,800);

135)what are the diff methods in a HashMap ?

Ans)The different methods of hashmap is get and put.by using getmethod we can find out specified values by passing key.where as by using put method we can store the elements in a HashMap.

Ex:S.o.p(hashmap.get(“lara”));//80

136) what is a TreeMap? How is different from a HashMap ?

Ans)TreeMap is similar to HashMap except that it stores keys in sorted order.It implements NavigableMap interface and SortedMap interfaces along with Map interface.

Ex:

TreeMap tm = new TreeMap();

// Put elements to the map

tm.put("Zara", new Double(3434.34));

tm.put("Mahnaz", new Double(123.22));

tm.put("Ayan", new Double(1378.00));

tm.put("Daisy", new Double(99.22));

tm.put("Qadir", new Double(-19.08));

137)what are Generics ?

Ans)Generics are used to create Generic classes and Generic methods which can work with different Types(Classes).

138)why do we need Generics ? can you give an example of how generics make a program simple ?

Ans)Consider the following example.

class MyList{

private List<String> values;

void add(String value){

values.add(value);

}

void remove(String value){

values.remove(value);

}

}

MyList can be used to store a list of Strings Only.

MyList mylist=new MyList();

mylist.add(“value 1”);

mylist.add(“value 2”);

consider the example with Generics,let’s replace String with T and create a new class.Now the MyListGeneric class can be used to create a list of integer or list of Strings

class MyListGeneic<T>{

private List<T> values;

void add(T value){

values.add(value);

}

void remove(T value){

values.remove(value);

}

MyListGeneic<String> myliststring=new MyListGeneric<String>();

myliststring.add(“value 1”);

myliststring.add(“value 2”);

MyListGeneic<Integer> mylistinteger=new MyListGeneric<Integer>();

mylistinteger.add(1);

mylistinteger.add(2);

139)How do you declare a Generic Class ?

Ans)we can declare the generic class as like as followes.

class MylistGeneric<T>

{

}

Here instated of T, we can use any valid identifier.

140)what are the restrictions in using generic type that is declared in a class declaration?

Ans)If we declare a Generic as class declaration,it can be used any where atype can be used in a class,method(return type,argument),member variable etc.

141)How can we restrict Generics to a subclass of particular class ?

Ans)In MyListGeneric,Type T is defined as part of class declaration.any java Type can be used a type for this class.if we would want to restrict the types allowed for a Generic Type,we can use a Generic Restrictions.consider the example below.in declaration of the class,we specified a constraint “T extends Number”.we can use the class MyListRestricted with any class extending Number-Float,Integer,Double…etc

class MyListRestricted<T extends Number>{

private List<T> values;

void add(T value){

values.add(value);

}

void remove(T value){

values.remove(value);

}

MyListRestricted<Integer> myListRestricted=new MyListRestricted<Integer>();

myListRestricted.add(1);

myListRestricted.add(2);

here String not valid substitute.

142)How can we restrict Generics to a super class of particular class ?

Ans)In MyListGeneric,type T is defined as part of class declaration.any java Type can be used a type for this class.if we would want to restrict the types allowed for a Generics type,we can use Generic Restrictions.in declaration of the class,we specified a constraint “T super Number”.we can use the class MyListRestricted with any class that is a super class of Number class.

143)How to define a class as multiple Generic elements?

Ans)consider the following example

class Gdemo<K,V>

{

K x;

V v;

void fun1(K k1)

{

x=k1;

}

void fun2(V v1)

{

Y=v1;

}

Public static void main(String[] args){

Gdemo<String,Integer> d1=new Gdemo<String,Integer>();

d1.fun1(“John”);

d1.fun2(113);

S.o.p(d1.x);

S.o.p(d1.y);

}

}

144)Explain about Exception Handling with an example?

Ans)The concept of identifying a logical error or an exception and catching that exception and assigne that exception object to the reference of the corresponding exception class,is known as exceptionhandling.

It is the procedure using which we can avoide exception reaching to the JVM.

Ex:

try{

int z=x/y;

S.o.p(“Z:”+z);

}catch(ArithmeticException e)

{

S.o.p(e);

}

145)What is the use of finally block in exception handling?

Ans)Finally blocks are the blocks,which are going to be get executed complsary,irrespective of the executions.for example we are getting the connection object from database to front end after that must and should close the connection object even though we are not perform any operation.

146)what are the key wards are available in exception handling ?

Ans)In exception handling there are six types of keywards are available

1)try

2)catch

3)throws

4)throw

5)finally

6)assert

147)In what kind of scenarios, a finally block is not executed?

Ans)the finally block is not executed in two scenarios.

i)if exception is thrown in finally.

ii)if JVM Crashes in between(for example,System.exit()).

148)Is finally block executed even when there is a return statement in the try block?

Ans)Yes.finally block is executed first,then later try is executed.consider the example.

Ex:

class Palindrome

{

public static void main(String args[])

{

System.out.println(Palindrome.test());

}

public static int test()

{

try {

//return 0;

return 100;

}

finally {

System.out.println("finally trumps return.");

}

}

}

149)Is a try block without corresponding catch block allowed ?

Ans)Yes.try allowed with out catch block.but we can use either catch or finally block.

Ex)consider the following example,

class Demo

{

public static void main(String[] args)

{

int a=10,b=2;

try{

int z=a/b;

}

finally{

//some code.

}

System.out.println("Hello World!");

}

}

150)explain the hierarchy of exception related classes in java?

Ans)Throwable is the super class of all the error handling classes.

Below class definations shows the pre-defined exception hierarchy in java

Class Error extends Throwable{}

Class Exception extends Throwable{}

Class RuntimeException extends Exception

Below classes definations shows creation of programmer defined exception in java

Class CheckedException1 extends Exception{}

Class CheckedException2 extends CheckedException1{}

Class UnCheckedException extends RuntimeException{}

Class UnCheckedException2 extends UnCheckedException{}

151)what is diff between an Error and an Exception ?

Ans)Error is used in situations when there is nothing a programmer can do about an error.

Ex:StackOverflowError,OutOfMemoryError.Exception is used when a programmer can handle the exception.

152)What is the diff b/w a checked exception and an un-checked exception ?

Ans)checked exception:-At the compilation compiler checks whether the statement is proven to generate any exception.if compiler finds a statement proven to generate an exception which is not the subclass of RuntimeException.then compiler checkes whether that statement is present inside the try and catch or not,if not then compiler rise the an compilation error. This is comes under checked exception.

Uncheckedexception:-if compiler finds a stmt that is proven to generate an exception.which is the sub class of RuntimeException.then compier would not check whether that stmt is present inside the try and catch.this type of exceptions are called unchecked exceptions.

153)How do you throw a checked Exception from a Method ?

Ans)consider the example below,The method addAmounts throws a new Exception.however,it gives us a copilation error because Exception is a Checked Exception.

Ex)class AmoutAdder{

static Amount addAmounts(Amount amount1.Amount amount2){

if(!amount1.currency.equals(amount2.currency)){

throw new Exception(“currencies don’t match”);//compiler error.

}

return new Amount(amount.currency,amount1.amount+amount2.amount);

}

}

Note:-it gives the compilation error.

Example with throws definition.

Ex)class AmoutAdder{

static Amount addAmounts(Amount amount1.Amount amount2)throws Exception{

if(!amount1.currency.equals(amount2.currency)){

throw new Exception(“currencies don’t match”);//compiler error.

}

return new Amount(amount.currency,amount1.amount+amount2.amount);

}

}

Note:-It compile success fully.

154)How do you create a custom exception classes?

Ans)we can create a custom exception by extending Exception or by extending RuntimeException class.if we extend Exception class, it will be a checked exception class.if we extends RuntimeException class,then we create an unchecked exception class.

155)Give the an example of custom exception class ?

Ans)consider the following example.

class InvalidAgeException extends Exception

{

public String toString()

{

String s1="Invalid Age Supplied by the end user.....!";

return s1;

}

}

class Emp

{

public float getPention(int age,float esal)throws Exception

{

float pention=0;

if(age>100||age<50)

{

throw new InvalidAgeException();

}else{

pention=(age\*esal)/100;

return pention;

}

}

}

class John

{

public static void main(String[] args)

{

int age=Integer.parseInt(args[0]);

float esal=Float.parseFloat(args[1]);

Emp e1=new Emp();

try{

float pen=e1.getPention(age,esal);

System.out.println("Employee Salary:"+pen);

}catch(Exception e2)

{

e2.printStackTrace();

}

System.out.println("age:"+age);

System.out.println("esal:"+esal);

}

}

156)What is the use of throws key ward?

Ans)The first usage of throws key ward we can explicitly mention that the function is proven to transfer it’s unhandled exceptions to the calling place.when we handle at calling place.

The second usage of throws key ward is to avaide try and catch blocks,w.r.to stmts proven to generate checked exception.

157)How should the exception catch blocks be ordered ?

Ans)specific Exception catch blocks should be before the catch block of Exception. For example ArithmeticException, NullPointerException, ArrayIndexOutOfBoundsException should be before Exception.below code gives a compilation error.

Ex:

class Exdemo1

{

public static void main(String[] args)

{

int i=Integer.parseInt(args[0].trim());

int j=Integer.parseInt(args[1].trim());

int k=Integer.parseInt(args[2].trim());

int l=Integer.parseInt(args[3].trim());

A a=null;

try{

int x=i/j;

if(x>5)

a=new A();

a.funA();

int ar[]=new int[k];

ar[l]=x;

}catch(Exception e)

{

e.printStackTrace();

}

catch(ArithmeticException e1)

{

e1.printStackTrace();

}

catch(NullPointerException e2)

{

e2.printStackTrace();

}

catch(ArrayIndexOutOfBoundsException e3)

{

e3.printStackTrace();

}

System.out.println("i value is:"+i);

System.out.println("j value is:"+j);

System.out.println("k value is:"+k);

System.out.println("l value is:"+l);

}

}

158)what is difference between throw and throws key wards ?

Ans)throw keyward is used to explicitly throw an exception,where as throws key ward declare the an exception.

Checked exceptions can not be propagated with throw keyward.where as checked exceptions can be propagated with throws key ward.

throw is used with in the method, where as throws is used at method signature.

159)What is StackOverflowError?

Ans) The StackOverFlowError is an Error Object thorwn by the Runtime System when it Encounters that your application/code has run out of the memory. It may occur in case of recursive methods or a large amount of data is fetched from the server and stored in some object. This error is generated by JVM.

160) What are the possible combination to write try, catch finally block?

Ans)

1 try{

//lines of code that may throw an exception

}catch(Exception e){

//lines of code to handle the exception thrown in try block

}finally{

//the clean code which is executed always no matter the exception occurs or not.

}

2 try{}finally{}

3 try{

}catch(Exception e){

//lines of code to handle the exception thrown in try block

}

161)what are the basic methods in File class ?

Ans)

create afile object

File file=new File(“filename.txt”);

Check if the file exists or not

System.out.println(file.exists());

If file does not exist creates it and returns true,if file exists,returns false.

System.out.println(file.createNewFile());

162)How do you handle directories in java?

Ans)In java we can handle directories and files as like as followes.

File directory=new File(“src/com/durga”);

To print full directory path,we can use the following methods.

System.out.println(directory.getAbsolutepath());

System.out.println(directory.isDirectory());

163)How do you write to a file using FileWriter class?

Ans)we can write to a file using FileWriter class,as like as followes,

FileWriter filewriter=new FileWriter(“durga.txt”,true);

filewriter.write(“How are you?”);

filewriter.flush();

filewriter.close();

164)How do you read from a file using FileReader class ?

Ans)we can read the file as like as followes.

FileReader fr=new FileReader(“file”);

char[] temp=new char[25];

s.o.p(fr.read(temp));

s.o.p(Arrays.toString(temp));

fr.close();

165)what is the use of BufferedWriter and BufferedReader classes in java ?

Ans) BufferedWriter and BufferedReader provide better buffering in addition to basic file writing and reading operations.using BufferedWriter and BufferedReader classes we can read and write large amount of data very fast.

166)what is the use of PrintWriter class?

Ans)printwriter is an advanced method to write the data.using we can write the data according to required format.generally this is used at server side code.for example in a servlet code.we can write as like as followes.

PrintWriter pw=res.getWriter();

167)Give the example,how to read the data from a console?

Ans)consider the following example,we can read the data from a console.

Ex: import java.io.\*;

class ReadConsole

{

public static void main(String[] args)

{

DataInputStream dis=new DataInputStream(System.in);

System.out.println("Pls enter the data");

try{

String str=dis.readLine();

System.out.println(str);

}catch(Exception e1)

{

e1.printStackTrace();

}

}

};

168) Give the example,how to write the data on a file ?

Ans)consider the following example,

import java.io.\*;

class WriteFile

{

public static void main(String[] args)

{

BufferedReader br=new BufferedReader(new InputStreamReader(System.in));

System.out.println("pls enter the data");

try{

FileOutputStream fos=new FileOutputStream("app.txt",true);

String data=br.readLine();

byte[] ba=data.getBytes();

fos.write(ba);

fos.close();

}catch(Exception e)

{

e.printStackTrace();

}

}

}

169)what is Serialization ?

Ans)The concept of serialization helps us to save and retrieve the state of an object.

Serialization🡪Convert object state to some internal object representation.

De-Serialization🡪The reverse.convert internal representation to object.

The two important methods are

ObjectOutputStream.writeObject();//serialize and write to file

ObjectInputStream.readObject();//read from file and deserialize.

170)How do you serialize an object using serializable interface ?

Ans)To serialize an Object it should be implement Serializable interface.Serializable interface doesn’t have any methods.it is an marker interface.

The ObjectOutputStream class is used to serialize an Object. The following SerializeDemo program instantiates an Employee object and serializes it to a file.

When the program is done executing, a file named employee.ser is created. The program does not generate any output, but study the code and try to determine what the program is doing.

Ex:

import java.io.\*;

public class SerializeDemo

{

public static void main(String [] args)

{

Employee e = new Employee();

e.name = "Duga";

e.address = "Sr nagar,hyd";

e.SSN = 11122333;

e.number = 101;

try

{

FileOutputStream fileOut =

new FileOutputStream("D:/yogicorejava/package/temp/employee.ser");

ObjectOutputStream out = new ObjectOutputStream(fileOut);

out.writeObject(e);

out.close();

fileOut.close();

System.out.printf("Serialized data is saved in /tmp/employee.ser");

}catch(IOException i)

{

i.printStackTrace();

}

}

}

171) The following DeserializeDemo program deserializes the Employee object created in the SerializeDemo program. Study the program and try to determine its output

import java.io.\*;

public class DeserializeDemo

{

public static void main(String [] args)

{

Employee e = null;

try

{

FileInputStream fileIn = new FileInputStream("D:/yogicorejava/package/temp/employee.ser");

ObjectInputStream in = new ObjectInputStream(fileIn);

e = (Employee) in.readObject();

in.close();

fileIn.close();

}catch(IOException i)

{

i.printStackTrace();

return;

}catch(ClassNotFoundException c)

{

System.out.println("Employee class not found");

c.printStackTrace();

return;

}

System.out.println("Deserialized Employee...");

System.out.println("Name: " + e.name);

System.out.println("Address: " + e.address);

System.out.println("SSN: " + e.SSN);

System.out.println("Number: " + e.number);

}

}

172)What do you do if only parts of the object have to be serialized ?

Ans)we mark all the properties of the object which should not be serialized as transient.Transient attributes in an object are not Serialized.we can make a variable transient.consider the following example

Class Rectangle implements Serializable{

Public Rectangle(int length,int breadth)

{

This.length=length;

This.breadth=breadth;

Area=length\*breath;

}

Int length;

Int breadth;

Transient int area;

}

173)are the constructors in an object invoked when it is de-serialized?

Ans)NO.when a class is de-serialized,initialization does not take place.The state of the object is retained as it is.

174)are the values of static variables stored when an object is serialized ?

Ans)Static variables are not part of the object,they are not serialized.

175)why we are not create the System class Object directly ?

Ans)Consider the following example,the System class defined as like as followes.that is the reason we can not create System class Object directly by using new operator.

Ex:

class Access1

{

private int x,y;

private Access1(){

System.out.println("inside constructor of class Access1");

}

public void fun1(){

System.out.println("inside fun1() of class Access1");

}

public static Access1 getAccess1(){

Access1 a1=new Access1();

return a1;

}

}

class Test6

{

public static void main(String[] args){

Access1 a1=Access1.getAccess1();

a1.fun1();

}

}

176)How to access inner class members or functions ?

Ans)Consider the following example,we can access the inner class members as like as followes.

Ex:

class Outer1

{

int i,j;

class Inner1

{

void funIn(){

System.out.println("inside funIn() of class Inner1");

}

}

void funOut(){

//Inner1 in=new Inner1();

//in.funIn();

System.out.println("inside funOut() of class Outer1");

}

public static void main(String[] args)

{

Outer1.Inner1 in=new Outer1().new Inner1();

in.funIn();

//in.funOut();

}

}

177)what is the need for threads in java ?

Ans)MultiThreading is one of the concept in multitasking,multitasking is the concept of multiple functions are executed simalteniously,that means processor or cpu only executed only one stmt at a time but not more then one.in this case one function is executed completely then only another function is executed until unless second function doesn’t executed in this case un necessarly time is wated.for example if one function is executed inside that function one io related operation is found then cpu goes to idle state.and DMA circuit is executed that io related operation in this case CPU is idle state.in order to over come that problem we can go for threading concept.and we can eliminate the idle state of the cpu.

178)How do you create a thread ?

Ans)we can create a threads in two ways,

i)By extending Thread class we can create the threads.

ii)By implementing Runnable interface we can create the threads.

179)How do you create a thread by extending thread class ?

Ans)we can create a thread by extends Thread class and we can provide implementation of run method given by the Runnable interface.consider the following example,to print the 1 to 100 numbers.

Class ThradA extends Thread{

Public void run()

{

For(int i=1;i<=100;i++)

{

System.out.println(i);

}

}

}

180)How do you create a thread by implementing Runnable interface ?

Ans)Threads can also create by implementing Runnable interface and implementing the method run(),given by the runnable interface.consider the following example

Ex:

Class ThreadA implements Runnable{

Public void run(){

for(int i=1;i<=100;i++){

System.out.println(i);

}

}

}

181)How do you run a Thread in java ?

Ans)Running the threads in java slightly different approach.if we want call the run method,we need to use start().this method was given by the Thread class.activally start method responsibility is to handovering the run method to jvm and jvm is running the thread.

Ex:

Class ThradA extends Thread{

Public void run()

{

For(int i=1;i<=100;i++)

{

System.out.println(i);

}

}

p.s.v.main(){

ThreadA a1=new ThreadA();

a1.start();

}

}

182)what are the different states of a thread ?

Ans)The different states of thred is as followes,

i)NEW

ii)READY STATE

iii)RUNNING

iv)BLOCKED/WAITING

v)TERMINATED/DEAD.

183)How many methods are available in Runnable interface?

Ans)Runnable interface having only one method that is run method.

184)How many run methods we can define in one class?

Ans)we can define only one run method,in one class but not allowed multiple run methods in a single class.

185)what is a scheduling ?

Ans)The concept of scheduling is out of multiple functions,The JVM is assigne particular time per one thread assigne the time per particular thread is nothing but scheduling.

186)what is priority of a thread ?How do you change the priority of a thread ?

Ans)priority means,preference.we use priorites for controlling the rescheduling of threads,when the threads are in the wait states.

🡪notify() would be notifying based on the priorities of threads mentioned.

🡪Priorities ranges from 1 to 10,one would be consider as lowest priority and 10 would be consider as the heighest priority.the default priority of thread is 5.

Ex: class ThradA extends Thread{

public void run()

{

for(int i=1;i<=100;i++)

{

System.out.println(i);

}

}

}

class ThradB extends Thread{

public void run()

{

for(int i=100;i<=200;i++)

{

System.out.println(i);

}

}

}

class PriarityTest

{

public static void main(String[] args){

ThradA t1=new ThradA();

t1.setPriority(10);

ThradB t2=new ThradB();

t2.setPriority(9);

t1.start();

t2.start();

}

}

187)what is a synchronization of threads ?

Ans) When we start two or more threads within a program, there may be a situation when multiple threads try to access the same resource and finally they can produce unforeseen result due to concurrency issue. For example if multiple threads try to write within a same file then they may corrupt the data because one of the threads can overrite data or while one thread is opening the same file at the same time another thread might be closing the same file.So there is a need to synchronize the action of multiple threads and make sure that only one thread can access the resource at a given point in time.

188)can you give an example of a synchronized block ?

Ans) Synchronized block can be used to perform synchronization on any specific resource of the method.

Suppose you have 50 lines of code in your method, but you want to synchronize only 5 lines, you can use synchronized block.

If you put all the codes of the method in the synchronized block, it will work same as the synchronized method.

Points to remember for Synchronized block

Synchronized block is used to lock an object for any shared resource.

Scope of synchronized block is smaller than the method.

Syntax to use synchronized block

synchronized (object reference expression) {

//code block

}

Ex: class Table{

void printTable(int n){

synchronized(this){//synchronized block

for(int i=1;i<=5;i++){

System.out.println(n\*i);

try{

Thread.sleep(400);

}catch(Exception e){System.out.println(e);}

}

}

}//end of the method

}

class MyThread1 extends Thread{

Table t;

MyThread1(Table t){

this.t=t;

}

public void run(){

t.printTable(5);

}

}

class MyThread2 extends Thread{

Table t;

MyThread2(Table t){

this.t=t;

}

public void run(){

t.printTable(100);

}

}

class TestSynchronizedBlock1{

public static void main(String args[]){

Table obj = new Table();//only one object

MyThread1 t1=new MyThread1(obj);

MyThread2 t2=new MyThread2(obj);

t1.start();

t2.start();

}

}

189)Can a static method be synchronized?

Ans)Yes.Consider the example below.

Synchronized static int getCount(){

return count;

}

Static methods and blocks are synchronized on the class.instance methods and blocks are synchronized on the instance of the class.i.e an object of the class.static synchronized methods and instance synchronized methods don’t affect eatch other.

190)what is the use of join method in threads?

Ans)join method is an instance method on the Thread class. Let’s see a small example to understand what join method does.

Ex:class ThreadA extends Thread{

int sum;

public void run(){

for(int i=0;i<=50;i++) {

sum=sum+i;

}

System.out.println(sum);

}

}

Class ThreadB extends Thread{

ThtreadA t1;

ThreadB(ThreadA t1){

This.t1=t1;

}

public void run(){

for(int i=100;i<=200;i++)

{

try{

If(i==150){

t1.join();

}

}catch(Exception e){e.printStackTrace();}

System.out.println(t1.sum);

}

}

}

191)Describe a few other important methods in threads ?

Ans)sleep(),sleep is a static method in Thread class.sleep method can throw a InterruptedException.sleep method causes the thread in execution to go to sleep for specified number of milliseconds.

yield(),yield method is changes the state of thread from RUNNING to RUNNABLE.

start(),this method is used for to handovering the run method to JVM.

192)What is deadlock ?

Ans)Let’s consider a situation where thread1 is waiting for thread2 and thread2 is waiting for thread1.this situation is called a deadlock.in a deadlock situation,both these threads would wait for one another.

193)what are the important methods in java for inter-thread communication ?

Ans)important methods are wait(),notify() and notifyAll().

194)what is the use of wait method ?

Ans)Below code shows how wait is used.wait method is defined in the object class.This causes the thread to wait until it is notified.

synchronized(thread){

thread.start();

thread.wait();

}

195)what is the use of notify method?

Ans)Below code shows how notify is used.notify method is defined in the Object class.This causes the object to notify other waiting threads.

synchronized(this){

calculateSumUptoMillon();

notify();

}

196)what is the use of notifyAll method?

Ans)if more then one thread is waiting for an object,we can notify all the threads by using notifyAll method.

Ex: thread.notifyAll()

197)What is JDBC API and when do we use it?

Ans)Java DataBase Connectivity API allows us to work with relational databases. JDBC API interfaces and classes are part of java.sql and javax.sql package. We can use JDBC API to get the database connection, run SQL queries and stored procedures in the database server and process the results.JDBC API is written in a way to allow loose coupling between our Java program and actual JDBC drivers that makes our life easier in switching from one database to another database servers easily.

198) What are different types of JDBC Drivers?

Ans)There are four types of JDBC drivers. Any java program that works with database has two parts, first part is the JDBC API and second part is the driver that does the actual work.

First driver class is:sun.jdbc.odbc.JdbcOdbcDriver.

Second Driver Class:oracle.jdbc.driver.OracleDriver.

Thired driver class:thired driver class is connection pooling concept,if we want get the connection object through middleware server we can get the connection object.

Fourth driver class:oracle.jdbc.driver.OracleDriver.

199) What is JDBC Connection? Explain steps to get Database connection in a simple java program?

Ans)JDBC Connection is like a Session created with the database server. You can also think Connection is like a Socket connection from the database server.

Creating a JDBC Connection is very easy and requires two steps:

i)Register and Load the Driver: Using Class.forName(), Driver class is registered to the DriverManager and loaded in the memory.

ii)Use DriverManager to get the Connection object: We get connection object from DriverManager.getConnection() by passing Database URL String, username and password as argument.

Ex:import java.sql.\*;

class Dbclient{

public static void main(String[] args)throws Exception{

Class.forName(“oracle.jdbc.driver.OracleDriver”);

Connection con=DriverManager.getConnection(“jdbc:oracle:thin:@localhost:1521:XE”,”durgatech”,”durgatech”);

System.out.println(“connection is success………!”);

Con.close();

}

}

200) What is the use of JDBC DriverManager class?

Ans)JDBC DriverManager is the factory class through which we get the Database Connection object. When we load the JDBC Driver class, it registers itself to the DriverManager, you can look up the JDBC Driver classes source code to check this.

Then when we call DriverManager.getConnection() method by passing the database configuration details, DriverManager uses the registered drivers to get the Connection and return it to the caller program.

201) How to get the Database server details in java program?

Ans)We can use DatabaseMetaData object to get the database server details. When the database connection is created successfully, we can get the meta data object by calling getMetaData() method. There are so many methods in DatabaseMetaData that we can use to get the database product name, it’s version and configuration details.

Ex)DatabaseMetaData metaData = con.getMetaData();

String dbProduct = metaData.getDatabaseProductName();

202)What is a Jdbc Statement ?

Ans)Statement is an predefined interface is used for to send the sql queries from front end to back end application and this interface available to the java.sql.\*; package .this inter face was given by the some of the important methods like execute(),executeQuery() and executeUpdate()……etc.using those methods we can send the sql queries.if we want getting the statement object we need to use the following pice of code.

Ex)Statement st=con.createStatement();

203) What is the difference between execute, executeQuery, executeUpdate?

Ans)The above three methods are given by the Statement interface.Generally

execute() is used for to perform the ddl operations like a create,drop,rename,alter and truncate and return type of this method is Boolean that means either true or false.

executeQuery() is used for to perform dql operations like a select operation and return type of this methods is ResultSet,ResultSet is an interface available to the java.sql.\*;

executeUpdate() is used for to perform dml operations like a insert,delete and update methods.and return type of this method is integer.

204) What is JDBC PreparedStatement?

Ans)JDBC PreparedStatement object represents a precompiled SQL statement. We can use it’s setter method to set the variables for the query.Since PreparedStatement is precompiled, it can then be used to efficiently execute this statement multiple times. PreparedStatement is better choice that Statement because it automatically escapes the special characters and avoid SQL injection attacks.

205) What are the benefits of PreparedStatement over Statement?

Ans)Some of the benefits of PreparedStatement over Statement are:

PreparedStatement helps us in preventing SQL injection attacks because it automatically escapes the special characters.

PreparedStatement allows us to execute dynamic queries with parameter inputs.

PreparedStatement is faster than Statement. It becomes more visible when we reuse the PreparedStatement or use it’s batch processing methods for executing multiple queries.

PreparedStatement helps us in writing object Oriented code with setter methods whereas with Statement we have to use String Concatenation to create the query. If there are multiple parameters to set, writing Query using String concatenation looks very ugly and error prone.

206) What is JDBC ResultSet?

Ans)JDBC ResultSet is like a table of data representing a database result set, which is usually generated by executing a statement that queries the database.

ResultSet object maintains a cursor pointing to its current row of data. Initially the cursor is positioned before the first row. The next() method moves the cursor to the next row. If there are no more rows, next() method returns false and it can be used in a while loop to iterate through the result set.

207) What are different types of ResultSet?

Ans)There are different types of ResultSet objects that we can get based on the user input while creating the Statement. If you will look into the Connection methods, you will see that createStatement() and prepareStatement() method are overloaded to provide ResultSet type and concurrency as input argument.

There are three types of ResultSet object.

ResultSet.TYPE\_FORWARD\_ONLY: This is the default type and cursor can only move forward in the result set.

ResultSet.TYPE\_SCROLL\_INSENSITIVE: The cursor can move forward and backward, and the result set is not sensitive to changes made by others to the database after the result set was created

ResultSet.TYPE\_SCROLL\_SENSITIVE: The cursor can move forward and backward, and the result set is sensitive to changes made by others to the database after the result set was created.

Based on the concurrency there are two types of ResultSet object.

ResultSet.CONCUR\_READ\_ONLY: The result set is read only, this is the default concurrency type.

ResultSet.CONCUR\_UPDATABLE: We can use ResultSet update method to update the rows data.

208) How to use JDBC API to call Stored Procedures?

Ans)Stored Procedures are group of SQL queries that are compiled in the database and can be executed from JDBC API. JDBC CallableStatement can be used to execute stored procedures in the database. The syntax to initialize CallableStatement is;

Ex) import java.sql.\*;

class ProcedureDemo

{

public static void main(String[] args)throws Exception

{

Class.forName("oracle.jdbc.driver.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE","naga","naga");

CallableStatement cst=con.prepareCall("{call proc1(?,?)}");

cst.setInt(1,1002);

cst.registerOutParameter(2,Types.INTEGER);

cst.execute();

int esal=cst.getInt(2);

System.out.println(esal);

con.close();

}

}

209) What is JDBC Batch Processing and what are it’s benefits?

Ans)Sometimes we need to run bulk queries of similar kind for a database, for example loading data from CSV files to relational database tables. As we know that we have option to use Statement or PreparedStatement to execute queries. Apart from that JDBC API provides Batch Processing feature through which we can execute bulk of queries in one go for a database.

JDBC API supports batch processing through Statement and PreparedStatement addBatch() and executeBatch() methods.

210) What is JDBC Transaction Management and why do we need it?

Ans) Transaction is a unit of work performed by front-end application to back-end application.Generally this concept is used for transportation projects and financial domain related projects.

Ex) importjava.sql.\*;

import java.io.\*;

classTransationMgmtDemo

{

public static void main(String[] args)

{

Connection con=null;

try{

Class.forName("oracle.jdbc.driver.OracleDriver");

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","durga","durga");

con.setAutoCommit(false);

Statement st=con.createStatement();

st.executeUpdate("insert into emp values(222,'gurga',8000)");

st.executeUpdate("insert into emp values(555,'reddy',9000)");

st.executeUpdate("insert into emp values(666,'sita',10000)");

con.commit();

System.out.println("Transaction success=======!");

}

catch(Exception e)

{

try{

con.rollback();

System.out.println("Transaction failure=======!");

e.printStackTrace();

}

catch(Exception e1)

{

e1.printStackTrace();

}

}

}

211) How to rollback a JDBC transaction?

Ans)We can use Connection object rollback() method to rollback the transaction. It will rollback all the changes made by the transaction and release any database locks currently held by this Connection object.

212)What is JDBC Savepoint?

Ans)Sometimes a transaction can be group of multiple statements and we would like to rollback to a particular point in the transaction. JDBC Savepoint helps us in creating checkpoints in a transaction and we can rollback to that particular checkpoint.

213) What is CLOB and BLOB datatypes in JDBC?

Ans)CLOB stands for character large object,using clob data type we can store the files and we can retrieve the files from a data base.and BLOB stands for Binary large object and it is used for to inserting the images and retrieve the images from data base.

214)Give the an example of CLOB data type?

Ans) consider the following example,

import java.sql.\*;

import java.io.\*;

public class ClobDemo

{

public static void main(String[] args)throws Exception

{

Class.forName("oracle.jdbc.driver.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","durgatech","durgatech");

System.out.println("con=============:"+con.getClass().getName());

PreparedStatement pst=con.prepareStatement("insert into web\_inf values(?,?)");

pst.setString(1,"app5");

File f=new File("web.xml");

FileReader fis=new FileReader(f);

pst.setCharacterStream(2,fis,(int)f.length());

pst.executeUpdate();

System.out.println("web application is stored success fully");

con.close();

}

}

215)Give an example of BLOB data type ?

Ans)Consider the following example,

import java.sql.\*;

import java.io.\*;

class BlobDemo1

{

public static void main(String[] args)throws Exception

{

Class.forName("oracle.jdbc.driver.OracleDriver");

System.out.println("driver is loaded.....");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","durgatech","durgatech");

System.out.println("driver is loaded.....");

PreparedStatement pst=con.prepareStatement("insert into company values(?,?)");

pst.setString(1,1001);

File f=new File("abc.gif");

FileInputStream fis=new FileInputStream(f);

pst.setBinaryStream(2,fis,(int)f.length());

pst.executeUpdate();

System.out.println("Employee Image is inserted success fully!");

con.close();

}

}

216) What do you understand by DDL and DML statements?

Ans)Data Definition Language (DDL) statements are used to define the database schema. Create, Alter, Drop, Truncate, Rename statements comes under DDL statements and usually they don’t return any result.

Data Manipulation Language (DML) statements are used to manipulate data in the database schema. Select, Insert, Update, Delete, Call etc are example of DML statements.

217) What is difference between java.util.Date and java.sql.Date?

Ans)java.util.Date contains information about the date and time whereas java.sql.Date contains information only about the date, it doesn’t have time information. So if you have to keep time information in the database, it is advisable to use Timestamp or DateTime fields.

218)what is a RowSet ?

Ans) JDBC RowSet is an interface of javax.sql.rowset interface. This interface is wrapper around a ResultSet object that makes possible to use resultSet as java beans object. It can be one bean that available for composing of an application. Because a it continually maintain a connection JDBC connection to the database.

Another advantage of JDBC RowSet is that it is used to makes ResultSet object scrollable and updateable. By default all the RowSet object are scrollable and updateable.

219)What is the diff b/w web server and application server?

Ans) What is a web Server

An webserver is an application, which provides runtime support for web applications,created by using servlet or jsp technologies

A web server will provide a web container and that web container will provide services of a web server to web applications

A webserver does not accept any enterpise component created through ejb technology.for example tomcat server.

What is a appliation server:-

An applicationserver is an application which provides run time support for both webapplications and also for enterpriseapplications

An applicationserver provides both webcontainr and ejbcontainer, to provide services to both web and ejb components

One difference b/w a web server and application server is ,a web server will provide only a web container but an application server provides both web and ejb containers.for example weblogic server.

220) What is the difference between GET and POST method?

Ans)We can send limited data with GET method and it’s sent in the header request URL whereas we can send large amount of data with POST because it’s part of the body.

GET method is not secure because data is exposed in the URL and we can easily bookmark it and send similar request again, POST is secure because data is sent in request body and we can’t bookmark it.

GET is the default HTTP method whereas we need to specify method as POST to send request with POST method.

221) What is MIME Type?

Ans)The “Content-Type” response header is known as MIME Type. Server sends MIME type to client to let them know the kind of data it’s sending. It helps client in rendering the data for user. Some of the mostly used mime types are text/html, text/xml, application/xml etc.

222) What is a web application and what is it’s directory structure?

Ans)Web Applications are modules that run on server to provide both static and dynamic content to the client browser. Java provides web application support through Servlets and JSPs that can run in a servlet container and provide dynamic content to client browser.pls refer to the directory structure to video session.

223) What is a servlet?

Ans)Java Servlet is server side technologies to extend the capability of web servers by providing support for dynamic response and data persistence.

The javax.servlet and javax.servlet.http packages provide interfaces and classes for writing our own servlets.

All servlets must implement the javax.servlet.Servlet interface, which defines servlet lifecycle methods. When implementing a generic service, we can extend the GenericServlet class provided with the Java Servlet API. The HttpServlet class provides methods, such as doGet() and doPost(), for handling HTTP-specific services.

224) What are the advantages of Servlet over CGI?

Ans)Servlet technology was introduced to overcome the shortcomings of CGI technology.

Servlets provide better performance that CGI in terms of processing time, memory utilization because servlets uses benefits of multithreading and for each request a new thread is created, that is faster than loading creating new Object for each request with CGI.

Servlets and platform and system independent, the web application developed with Servlet can be run on any standard web container such as Tomcat, JBoss, Glassfish servers and on operating systems such as Windows, Linux, Unix, Solaris, Mac etc.

225) What are common tasks performed by Servlet Container?

Ans)Servlet containers are also known as web container, for example Tomcat. Some of the important tasks of servlet container are:

i)whenever we are making the request,first container loading the servlet class and it creates that class object.

ii)after that the container calling the life cycle methods automatically.initially init() and service() are called.

iii)when the second client is making the request,init() doesn’t called only service() is called.

iv)whenever stoping the server then automatically container is going to be called destroy method.

226) What is ServletConfig object?

Ans)javax.servlet.ServletConfig is used to pass configuration information to Servlet. Every servlet has it’s own ServletConfig object and servlet container is responsible for instantiating this object. We can provide servlet init parameters in web.xml file or through use of WebInitParam annotation. We can use getServletConfig() method to get the ServletConfig object of the servlet.

227) What is ServletContext object?

Ans)javax.servlet.ServletContext interface provides access to web application parameters to the servlet. The ServletContext is unique object and available to all the servlets in the web application. When we want some init parameters to be available to multiple or all of the servlets in the web application, we can use ServletContext object and define parameters in web.xml using <context-param> element. We can get the ServletContext object via the getServletContext() method of ServletConfig.

228) What is difference between ServletConfig and ServletContext?

Ans)Some of the differences between ServletConfig and ServletContext are:

ServletConfig is a unique object per servlet whereas ServletContext is a unique object for complete application.

ServletConfig is used to provide init parameters to the servlet whereas ServletContext is used to provide application level init parameters that all other servlets can use.

We can’t set attributes in ServletConfig object whereas we can set attributes in ServletContext .

229) What is Request Dispatcher?

Ans)RequestDispatcher interface is used to forward the request to another resource that can be HTML, JSP or another servlet in same application. We can also use this to include the content of another resource to the response. This interface is used for inter-servlet communication in the same context.

There are two methods defined in this interface:

void forward(ServletRequest request, ServletResponse response) – forwards the request from a servlet to another resource (servlet, JSP file, or HTML file) on the server.

void include(ServletRequest request, ServletResponse response) – includes the content of a resource (servlet, JSP page, HTML file) in the response.

We can get RequestDispatcher in a servlet using ServletContext getRequestDispatcher(String path) method. The path must begin with a / and is interpreted as relative to the current context root.

230) Do we need to override service() method?

Ans)When servlet container receives client request, it invokes the service() method which in turn invokes the doGet(), doPost() methods based on the HTTP method of request. I don’t see any use case where we would like to override service() method. The whole purpose of service() method is to forward to request to corresponding HTTP method implementations. If we have to do some pre-processing of request, we can always use servlet filters and listeners.

231)What is difference between GenericServlet and HttpServlet?

Ans)GenericServlet is protocol independent implementation of Servlet interface whereas HttpServlet is HTTP protocol specific implementation. Most of the times we use servlet for creating web application and that’s why we extend HttpServlet class. HttpServlet class extends GenericServlet and also provide some other methods specific to HTTP protocol.

232)What is the inter-servlet communication?

Ans)When we want to invoke another servlet from a servlet service methods, we use inter-servlet communication mechanisms. We can invoke another servlet using RequestDispatcher forward() and include() methods and provide additional attributes in request for other servlet use.

233)How do we call one servlet from another servlet?

Ans)We can use RequestDispatcher forward() method to forward the processing of a request to another servlet. If we want to include the another servlet output to the response, we can use RequestDispatcher include() method.

234)What is servlet attributes and their scope?

Ans)Servlet attributes are used for inter-servlet communication, we can set, get and remove attributes in web application. There are three scopes for servlet attributes – request scope, session scope and application scope.

ServletRequest, HttpSession and ServletContext interfaces provide methods to get/set/remove attributes from request, session and application scope respectively.

235)How many methods are available in Servlet interface ?

Ans)In servlet technology,the servlet interface having five life cycle methods are available,they are

i)init()

ii)service()

iii)destroy()

iv)getServletInfo()

v)getServletConfig()

🡪the above three methods are life cycle methods and remaining two methods are non-life cycle methods.

236)What are different methods of session management in servlets?

Ans)Session is a conversional state between client and server and it can consists of multiple request and response between client and server. Since HTTP and Web Server both are stateless, the only way to maintain a session is when some unique information about the session (session id) is passed between server and client in every request and response.

Some of the common ways of session management in servlets are:

HTML Hidden Field

Cookies

URL Rewriting

Session Management API

237)How to configure servlet inside the web.xml file ?

Ans)we can configure the servlet class inside the web.xml file as like as followes.

<web-app>

<servlet>

<servlet-name>mysrv</servlet-name>

<servlet-class>MyServlet</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>mysrv</servlet-name>

<url-pattern>/srv</url-pattern>

</servlet-mapping>

</web-app>

238)How to configure the filter inside the web.xml file ?

Ans)we can configure the filter class as like as followes.

<filter>

<filter-name>myFilter</filter-name>

<filter-class>MyFilter</filter-class>

</filter>

<filter-mapping>

<filter-name>myFilter</filter-name>

<url-pattern>/filter</url-pattern>

</filter-mapping>

239)when we configure filter and servlet inside the configuration file,container is loading first which one ?

Ans)whenever we configure filter and servlet, first the container is loading filter class then after that container is loading the servlet class.that means first the container is given to the preference to load the filter classes only.

240) How to get the IP address of client in servlet?

Ans)We can use request.getRemoteAddr() to get the client IP address in servlet.

241)What is JSP and why do we need it?

Ans)JSP stands for JavaServer Pages. JSP is java server side technology to create dynamic web pages. JSP is extension of Servlet technology to help developers create dynamic pages with HTML like syntax.

We can create user views in servlet also but the code will become very ugly and error prone. Also most of the elements in web page is static, so JSP page is more suitable for web pages. We should avoid business logic in JSP pages and try to use it only for view purpose. JSP scripting elements can be used for writing java code in JSP pages but it’s best to avoid them and use JSP action elements, JSTL tags or custom tags to achieve the same functionalities.What is JSP and why do we need it?

242) What are JSP lifecycle methods?

JSP lifecycle methods are:

1)jspInit(): This method is declared in JspPage and it’s implemented by JSP container implementations. This method is called once in the JSP lifecycle to initialize it with config params configured in deployment descriptor. We can override this method using JSP declaration scripting element to initialize any resources that we want to use in JSP page.

2)\_jspService(): This is the JSP method that gets invoked by JSP container for each client request by passing request and response object. Notice that method name starts with underscore to distinguish it from other lifecycle methods because we can’t override this method. All the JSP code goes inside this method and it’s overridden by default. We should not try to override it using JSP declaration scripting element. This method is defined in HttpJspPage interface.

3)jspDestroy(): This method is called by container when JSP is unloaded from memory such as shutting down application or container. This method is called only once in JSP lifecycle and we should override this method to release any resources created in JSP init method.

243) Which JSP lifecycle methods can be overridden?

Ans)We can override jspInit() and jspDestroy() methods using JSP declaration scripting element. We should override jspInit() methods to create common resources that we would like to use in JSP service method and override jspDestroy() method to release the common resources.

244) What are different types of comments in JSP?

JSP pages provide two types of comments that we can use:

Ans)HTML Comments: Since JSP pages are like HTML, we can use HTML comments like <-- HTML Comment -->. These comments are sent to client also and we can see it in HTML source. So we should avoid any code level comments or debugging comments using HTML comments.

JSP Comments: JSP Comments are written using scriptlets like <%-- JSP Comment --%>. These comments are present in the generated servlet source code and doesn’t sent to client. For any code level or debugging information comments we should use JSP comments.

245)What is Scriptlet, Expression and Declaration in JSP?

Ans)Scriptlets, Expression and Declaration are scripting elements in JSP page using which we can add java code in the JSP pages.

A scriptlet tag starts with <% and ends with %>. Any code written inside the scriptlet tags go into the \_jspService() method. For example;

<%

Date d = new Date();

System.out.println("Current Date="+d);

%>

Since most of the times we print dynamic data in JSP page using out.print() method, there is a shortcut to do this through JSP Expressions. JSP Expression starts with <%= and ends with %>.

<% out.print("Durga"); %> can be written using JSP Expression as <%= "Durga" %>

Notice that anything between <%= %> is sent as parameter to out.print() method. Also notice that scriptlets can contain multiple java statements and always ends with semicolon (;) but expression doesn’t end with semicolon.

JSP Declarations are used to declare member methods and variables of servlet class. JSP Declarations starts with <%! and ends with %>.

For example we can create an int variable in JSP at class level as <%! public static int count=0; %>.

246)What are JSP implicit objects?

Ans)We have 9 implicit objects that we can directly use in JSP page.

1)out Object

2)request Object

3)response Object

4)config Object

5)application Object

6)session Object

7)pageContext Object

8)page Object

9)exception Object

247)Which implicit object is not available in normal JSP pages?

Ans)JSP exception implicit object is not available in normal JSP pages and it’s used in JSP error pages only to catch the exception thrown by the JSP pages and provide useful message to the client.

248) What are the benefits of PageContext implicit object?

Ans)JSP pageContext implicit object is instance of javax.servlet.jsp.PageContext abstract class implementation. We can use pageContext to get and set attributes with different scopes and to forward request to other resources. pageContext object also hold reference to other implicit object.

This is the only object that is common in both JSP implicit objects and in JSP EL implicit objects.

249)How to ignore the EL expression evaluation in a JSP?

Ans)We can ignore EL evaluation in JSP page by Using page directive as

<%@ page isELIgnored="true" %>

250)what are the important jsp predefined action tags?

Ans)The important predefined action tags are

i)<jsp:forward>

ii)<jsp:include>

iii)<jsp:useBean>

iv)<jsp:setProperty>

v)<jsp:getProperty>

vi)<jsp:param>

vii)<jsp:plugin>

viii)<jsp:fallback>…….etc

251)How do traditional web applications work ?

Ans)The traditional web applications are working based on Http Request and Http Response cycle.following are the steps.

i)when user initiates an action in the browser,a http request is created by the browser.

ii)web server creates a HttpServletRequest based on the content of Http Request.

iii)The web application handles the HttpServletRequest.

iv)A HttpServletResponse is returned.This is converted to Http Response.

v)The Http Response is rendered by the browser.

252)what is MVC pattern ?

Ans)MVC stands for model ,view and controller.it is a software architectural pattern for implementing user interfaces.

Controller:controller means it is an bash of the entire application.that means every request and response is take care about the controller.

Model:Model means it is an business components,that means we can use model components we can develop the business logic.

View:View means to presenting the data.

253)How does struts 1.x handle the request ?

Ans)i)User clicks on a link in an HTML page.

ii)Servlet controller(ActionServlet) receives the request,forwards it to RequestProcessor

iii)RequestProcessor looks up struts-config.xml and routes to an action.

iv)Action makes calls to populate the Model.

v)Action forwards to a view resource.

vi)Request Processor looks up the mapping for the requested resource and forwards to the appropriate jsp page.

vii)jsp file is invoked and sent to the browser as HTML.

254)How does Struts2 handle the request ?

Ans)i)user sends a request for the action.

ii)initial request goes to the servlet container

iii)Container looks up web.xml file and finds the frontcontroller.

iv)container invokes the front controller.in the beginning of struts2,FilterDispatcher was used as frontcontroller,since struts 2.1 on wards it is recommended to use StrutsPreparedAndExecuteFilter.

v)Controller interacts with ActionMapper and invokes the ActionProxy.

vi)ActionProxy forwards the request to the ActionInvocation

vii)ActionInvocation invokes all the interceptors in the chain.

viii)Result is sent back to the ActionInvocation and a HttpServletResponse is generated.

255)How is Struts2 diff from Struts1.x ?

Ans)i)in 1.x,declarative validations are done by using validator framewok.where as in 2.x validatios are done by using Xwork2 frame work.the reason is,it supports validations through annotation also.

ii)in 1.x action class is not a thread safe,where as 2.x by default action class is thread safe.

iii)in 1.x we have only a jsp as a new technology,it means we can not send other type of view except either html or jsp.where as struts 2.x all the view technologies are supported like velocity,facelets and prime faces….etc.

256)can you explain more about interceptors in struts2 ?

Ans)1)interceptors intercepts the requests,and provides some additional processing before and after the execution of action classs.

2)common functionalities required by action classes are implemented as interceptors.this makes the action class lightweight.

3)The framework provides a default set of interceptors.we can write our own custom interceptor classes also.

4)The interceptor which is executed before and after the action is executed.to provide pre-processing and post-processing of request.

257)can you provide an example configuration of an interceptor in struts2?

Ans)in the below example,interceptor1 and interceptor2 are configured for the action login.

<interceptors>

<interceptor name=”interceptor1” class=”Interceptor1\_class\_name”/>

<interceptor name=”interceptor2” class=”Interceptor2\_class\_name”/>

</interceptors>

<action name=”login” class=”LoginAction”>

<interceptor-ref name=”interceptor1”/>

<interceptor-ref name=”interceptor2”/>

<result name=”input”>login.jsp</result>

<result name=”success”>success.jsp</result>

</action>

258)How do you create a common set of Interceptors which apply to every request in struts 2?

Ans)consider the below example,defaultstack is now a common set of interceptors.it can be used on all the actions as needed.

<interceptors>

<interceptor-stack name=”defaultStack”>

<interceptor-ref name=”exception”/>

<interceptor-ref name=”servlet-config”/>

<interceptor-ref name=”prepare”/>

<interceptor-ref name=”checkbox”/>

<interceptor-ref name=”params”/>

<interceptor-ref name=”conversionError”/>

<interceptors/>

We can configure the defaultStack as the default for all actions.

<default-interceptor-ref name=”defaultStack”/>

Other option is to configure it for every action

<action name=”login” class=”LoginAction”>

<interceptor-ref name=”defaultStack”/>

</action>

259)Can you give couple of examples of common interceptors in spring2?

Ans)The important common interceptors are debugging,fileUploading,i18n,logger,tokens……etc

260)How do you configure global exception handiling in struts2 ?

Ans)Instated of writing try and catch in every method to handle exceptions, struts2 provides a global exception handling mechanism.if the exception is not caught by any of the implemented methods, the global exception handling catches the exception and shows the error page.This is also called declarative error handling.consider the following example,

<global-exception-mapping>

<exception-mapping exception=”java.lang.exception” result=”exception”/>

</global-exception-mapping>

<global-result>

<result name=”exception”>/error.jsp</result>

</global-result>

261)what are the different view technologies supported by struts2?

Ans)The following different view technologies are supported by the struts2 frame work

Java server pages(include JSTL and JSF),Velocity,FreeMaker,face lets…..etc.

262)what is ValueStack ?

Ans)The valueStack is a storage area holding all data associated with processing of a request.The ValueStack is per-request.Any value placed on the stack during request processing is accessible later in the processing of same request.Data is placed on the valuestack by various interceptors.OGNLis the preferred way of accessing values on the ValueStack.

263)What is OGNL ?

Ans)The Object-Graph Navigation Language is a powerful expression language that is used to reference and manipulate data on the ValueStack.OGNL also helps in data transfer and type conversion.

264)What is struts-default package ?

Ans)The default built-in package in struts frame work is struts-default.xml.

265)what are the aware of interfaces in struts 2.x ?

Ans)In struts 2.x ,we have the following five Aware of interfaces are available.

i)ApplicationAware interface

ii)SessionAware interface

iii)ServletRequestAware interface

iv)ServletResponseAware interface

v)ParameterAware interface.

266) What are the two types of validations supported by Validator FrameWork?

Ans: Validator Framework is used for form data validation. This framework provides two types of validations:

1)Client Side validation on user’s browser

2)Server side validation.

267) What’s DynaActionForm?

Ans) DynaActionForm is a special type of actionForm class (sub-class of ActionForm Class) that’s used for dynamically creating form beans. It uses configuration files for form bean creation.

268)In struts1.x how many action forms are available ?

Ans)In struts there are totally 6 types forms are available.

i)org.apache.struts.action.ActionForm

ii) org.apache.struts.action.DynaActionForm

iii) org.apache.struts.validator.ValidatorForm

iv) org.apache.struts.validator.DynaValidatorForm

v) org.apache.struts.validator.ValidatorActionForm

vi) org.apache.struts.validator.DynaValidatorActionForm

269)How many types of action classes are available in struts 1.x ?

Ans)In struts there are six types of action classes are available,using which we can implement business logic.

i)Action

ii)DispatchAction

iii)LookupDispatchAction

iv)SwitchAction

v)ForwardAction

vi)IncludeAction.

270)what is the use of validate() and reset() methods ?

Ans)validate method is used for to apply validations on the user input and reset method is used for to initialize or reset the bean parameters with default values,before storing the new values to the variables.

271)what is a dependency Injection ?

Ans)Dependency Injection mechanism means as a java programmer responsibility is to developing java classes and Objects are created and injecting to our application is take care about spring framework. Injecting an objects into our application is nothing but dependency injection or inversion of control.

272)why is spring one of the most popular java related frameworks ?

Ans)Spring framework enables development of loosely coupled classes based on well defined interfaces.

🡪By using spring frame work testing is very simple.

🡪Beauty of spring framework is that it provides great integration support with other frameworks like a struts,jsp,jpa..etc

273)what are the different modules in spring frameworks?

Ans)In spring mainly seven modules are available,they are

i)spring IOC module or dependency injection mechanism

ii)Spring with jdbc module

iii)Spring with orm module

iv)Spring with aop module

v)Spring with Jee module

vi)Spring with MVC module.

vii)Spring with WebMVC module.

274)can you give an overview of a web application that is implemented using spring related modules?

Ans)Different layers of a web application can be implemented using different spring modules.

Service &business layers

Core business logic using simple POJOs managed by Spring IOC container.

Transaction management using Spring AOP.

Integration laye

Spring ORM to integrate with Database(jpa,ibaties).

Spring Ws to consume web services.

Web layer

Spring MVC to implement MVC pattern.

Spring ws to expose web services.

275)what are the major features in different versions of Spring ?

Ans)In Spring 2.5 annotation-driven configuration possible.

In spring 3.0 made great use of the java 5 improvements in language.

In spring 4.0 is the first version to fully support java 8 features.and minimum version of spring4 is java6.

276)what are the latest specifications supported by spring 4.0?

Ans)Spring framework4.0 supports the java EE specifications.

i)JMS

ii)JTA

iii)JPA

iv)Bean validations.

277)can you describe some of the new features in spring 4.0?

Ans)i)Spring websocket module provides supported for communication to web applications

ii)Spring 4 focused on servlet 3.x environ ment

iii)@RestController annotation is introduced in Spring MVC applications.

iv)spring 4 supported by the json format…..etc.

278)what is auto-wiring ?

Ans)auto-wiring is one of the concept,using this concept we can inject the inner beans automatically inside outer bean with out configuring explicitly,when we are injecting like that then we can say that bean auto-wiring.if we want inject like that first we need to enable the bean auto wiring concept in our application,if we want enable along with bean configuration we can use autowire attribute.

279)What are the auto-wire values are available in sprin frame work?

Ans)in spring framework there are five types of auto-wire values are available.

i)byName

ii)byType

iii)constructor

iv)autodetect

v)none.

280)How to configure the Spring bean in Spring configuration file ?

Ans)Consider the following example,

public class WelcomeBean

{

private String message;

public void setMessage(String message)

{

this.message=message;

}

public String getMessage()

{

return message;

}

public void show()

{

System.out.println(message);

}

}

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN 3.0//EN" "http://www.springframework.org/dtd/spring-beans-3.0.dtd">

<beans>

<bean id="id1" class="WelcomeBean">

<property name="message">

<value>WelcomeToDurgasoftware solutions..................</value>

</property>

</bean>

</beans>

281)How do you represent stateful bean in spring ?

Ans)Stateful beans are represented in spring with a prototype scope.A new instance is created every time a request for bean is made.for example,

Ex:<bean id=”id1” class=”com.durga.bean.SampleBean” scope=”prototype”>

282)How do you use values defined in a property file in an application context.xml ?

Ans)consider the following example,we can configure the datasource,url,username and password.the properties file name is database-connection.properties.

<bean class=”org.springframework.beans.factory.config.PropertyPlaceholderConfigurer”>

<property name=”locations” value=”classpath:database-connection.properties”/>

</bean>

<bean id=”datasource” class=”org.apache.commons.dbcp.BasicDataSource”>

<property name=”driverClassName” value=”${jdbc.driverClassName}”/>

<property name=”url” value=”${jdbc.url}”/>

<property name=”username” value=”${jdbc.username}”/>

<property name=”password” value=”${jdbc.password}”/>

</bean>

283)How do you implement cross cutting concerns in a web applications ?

Ans)Functionality spanning multiple layears of an application are called cross cutting concerns.for example logging,security,declarative transactions.

Cross cutting concerns are best implemented using Aspect Oriented Programming.AOP enables us to apply the cross cutting features across multiple classes.

284)what is an Aspect and Pointcut in AOP ?

Ans)aspect🡪aspect is nothing but name of the service,in aop technically we call them aspect.

Ex:logging,transaction management,advice is nothing but implementation of aspect is nothing but advice.

Pointcut:point cut is used for which method we want apply the service,those methods we can apply the services

285)what are the diff types of AOP advices ?

Ans)In AOP mainly there are four types of advices are available,

i)beforeadvice:is used for before executing the business method this advice is executed.

ii)afteradvice:is used for after executing the business method this advice is executed.

iii)aroung advice:around advice is the combination of before and after advices.

iv)throws advice:is used for whenever business logic is rise the an exception then then this service is executed.

286)How do you define transaction management for Spring-Hibernate integration ?

Ans)First step is to define a transaction manager in the xml file,

<bean id=”id1” class=”org.springframework.orm.hibernate.HibernateTransactionManager”/>

<tx:annotation-driven/>

Next, we can add the @Transactional annotation on the method which need to part of a transaction.

@Transactional(readOnly=true)

Public class CustomerDaoImpl implements CustomerDao{}

287)what are the utility methods available to test JDBC classes ?

Ans)countRowsInTable(..)&deleteFromTables(..).

countRowsInTableWhere(..)&deleteFromTablesWhere(..).

dropTables(..).

288)How do you setup a Session Factory to integrate Spring and Hibernate ?

Ans)consider the following example,

<bean id="id2" class="org.springframework.orm.hibernate3.LocalSessionFactoryBean">

<property name="dataSource">

<ref bean="id1"/>

</property>

<property name="mappingResources">

<list>

<value>student.hbm.xml</value>

</list>

</property>

<property name="hibernateProperties">

<props>

<prop key="hibernate.dialect">org.hibernate.dialect.Oracle9Dialect</prop>

<prop key="hibernate.show\_sql">true</prop>

<prop key="hibernate.hbm2ddl.auto">update</prop>

</props>

</property>

</bean>

289)How does request flow happen in Spring MVC ?

Ans)all the requests arrive at the DispatcherServlet (front controller).

🡪when request is sent from a browser then a front controller servlet of springMVC called Dispatcher Servlet traps the given request

🡪Dispatcher servlet takes the help of HandlerMappingBean,to find a suitable controller for Handling the given request.

🡪Dispatcher servlet deligates the given request to the ControllerBean

🡪The Method in which either “business” or intermediate logic defined for a request is executed.

🡪The method of controller bean returns a ModelAndView(mav) object.

🡪DispatcherServlet calles a ViewResolverBean to find the appropriate view for sending the response.

🡪The DispatcherServlet Forwards the request a view.

🡪Finally the response generated by the view will be a displayed on browser.

290)can you list a few advantages of using spring MVC framework ?

Ans)Spring MVC supports diff view technologies like a jsp,velocity,free maker…..etc.

Spring MVC generate diff formats of content like a XML,JSON……etc.

Highly convenient tag library

Flexible to integrate all other frame works like a struts,JSF…etc.

291)Give examples of important Spring MVC annotations?

Ans)The important Spring MVC annotations are

i)@Controller🡪is used for at class level

ii)@RequestMapping🡪is used for at method level or class level

iii)@PathVariable🡪is used for at argument level.

292)How do you integrate Spring MVC with tiles?

Ans)Tiles helps us to define the layout for a web page.we can integrate Spring MVC with tiles by configuring TilesConfigurer.

<bean id=”id1” class=”org.spfw.web.servlet.view.tiles2.TilesConfigurer” p:definitions=”/WEB-INF/tiles-defs/templates.xml”/>

<bean id=”id2” class=”org.spfw.web.servlet.view.UrlBasedViewResolver” p:viewClass=”org.spfw.web.servlet.view.tiles2.TilesView”/>

293)How do you configure Spring MVC web application to use UTF-8 encoding for handling forms ?

Ans)using org.spfw.web.filter.CharacterEncodingFilter,

<filter>

<filter-name>encoding-filter</filter-name>

<filter-class>org.spfw.web.filter.CharacterEncodingFilter</filter-class>

<init-param>

<param-name>encoding</param-name>

<param-value>UTF-8</param-value>

</init-param>

</filter>

<filter-mapping>

<filter-name>encoding-filter</filter-name>

<url-pattern>/\*</url-pattern>

</filter-mapping>

294)How do you enable Spring security for a web application ?

Ans)Spring security is used to implement Authenication and Authorization for a web application.

<filter>

<filter-name>springSecurityFilterChain</filter-name>

<filter-class>org.spfw.web.filter.DelegatingFilterProxy</filter-class>

</filter>

<filter-mapping>

<filter-name> springSecurityFilterChain </filter-name>

<url-pattern>/\*</url-pattern>

</filter-mapping>

295)what is the advanced container in spring framework ?

Ans)Advanced container of Spring framework is ApplicationContext,this is an predefined interface,available to the org.springframework.context.\*.and this interface was implemented by the following three classes.1)ClassPathXmlApplicationContext() 2)FileSystemXmlApplicationContext() 3)XmlWebApplicationContext().all these classes are available to the org.springframework.context.support.\*.

296)How do you configure the frontcontroller in web.xml file ?

Ans)consider the following example,

<web-app>

<servlet>

<servlet-name>springapp</servlet-name>

<servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>

<load-on-startup>1</load-on-startup>

</servlet>

<servlet-mapping>

<servlet-name>springapp</servlet-name>

<url-pattern>\*.do</url-pattern>

</servlet-mapping>

</web-app>

297)what is the StoredProcedure usage in spring framework ?

Ans)The usage of storedprocedure class to class procedure,then we can use storedprocedure class.when we are using the procedures our application performance is improved.instated of hitting directly that data base,we can call the procedure and procedure will take care about to call the oracle table.

298)What are the important methods in jdbctemplate class ?

Ans)The important methods of JdbcTemplate class is as follwes,

i)execute()🡪is used for to call the ddl operations like drop,rename,alter,truncate,create.the return type of this method is void.

ii)update()🡪is used for to call the dml operations like delete,update,insert.the return type of this method is int.

iii)Query methods,using this methods we can select the data from database.for example the query methods are a)queryForInt(),b)queryForLong(),c)queryForObject()…..etc.

299)what are the dependency-check values in spring framework ?

Ans) Validating the dependency:- in springfw, either in explicit wiring or in autowiring if all properties are not configured in xml, but still spring continer creates a n object of the bean class.

By default springcontainer doesn’t verify whether call properties are set in xml (or) not.

If we enable the dependency validation then spring container verifies whether all dependencies are set (or) not if not then container doesn’t create an obj and throws an exception.

To enable dependency validation, we need to add dependency-check attribute for the <bean>

The different values of dependency-check attributes are

1. No(default) 2) simple 3) objects 4) all

**Simple:-**in this case spring container verifies for primitives and collections are set (or) not. If not set then exception will be thrown.

**Object:-**in this case, spring container verifies whether objects are set (or) not, if not the container throws an exception.

**All:-**in this case the container verifies for both primitives collections and objects.

300)what are the predefined life cycle methods in spring framework ?

Ans)in spring there two predefined life cycle methods are available,1)afterPropertiesSet() and destroy().these method are given by the InitializingBean(i) and DisposableBean(i).these two are pre defined interfaces available to the org.springframework.beans.factory.\*;

Life cycle methods means automatically called by the container such type of methods we call them as life cycle methods.

301)what is JPA ?How is Hibernate related to JPA?

Ans)JPA(Java Persistance API) is a specification of how the object relational mapping should be done.java is object oriented programming language.jpa defines specification,but not implementation.

Hibernate is an implementation of the JPA specification.all annotations are given by the jpa and implementation was taken by the hibernate frame work.if directly use hibernate annotations ,we can’t easily switch to anoter ORM framework.

302)what is Hibernate ?what does it help a programmer do ?

Ans)Hibernate is an ORM frame work.an object relational mapping frame work.

Java is an object oriented programming language.data from java objects need to be stored in relational tables,hibernate provides an alternative way of storing data from java objects into a relational database.

303)Explain high level architecture of Hibernate framework ?

Ans)The below points are represents the high level points of hibernate architecture,

i)SessionFactory🡪it is an interface available to the org.hibernate.\*;SessionFactory is an highlevel object in hibernate framework.

ii)create the session object using Session interface,available to the org.hibernate.\*;

iii)create the java pojo class object and persist that object

iv)Transaction is used for unit of work to commit or roll back the data.it is available to the org.hibernate.\*;

304) can you give an example of an entity mapping with hibernate ?

Ans)consider the example,

import javax.persistence.\*;

@Entity

@Table(name="student\_table")

public class Student

{

@Id

@Column(name="stno")

private Integer sno;

@Column(name="sname",length=10)

private String sname;

@Column(name="stadd",length=10)

private String address;

public void setSno(Integer sno)

{

this.sno=sno;

}

public Integer getSno()

{

return sno;

}

public void setSname(String sname)

{

this.sname=sname;

}

public String getSname()

{

return sname;

}

public void setAddress(String address)

{

this.address=address;

}

public String getAddress()

{

return address;

}

}

305)How do you indicate to hibernate that a table will not be updated through the application?

Ans)@Immutable annotation to indicate to hibernate that a table will not be updated from the application.This allows hibernate to make some minor performance optimizations.

306)How can you generate a unique identifier using hibernate ?

Ans)Hibernate can create identifier values automatically,for example

i)IDENTITY:supports identity column in db2,mysql and Sybase.

ii)SEQUENCE🡪is used to generate sequence.

iii)CUSTOMGENERATOR

iv)AUTO.

307)How do you implement single table per class hierarchy strategy with hibernate ?

Ans)consider the following example,

<hibernate-mapping>

<class name="com.javatpoint.mypackage.Employee" table="emp121" discriminator-value="durga">

<id name="id">

<generator class="increment"></generator>

</id>

<discriminator column="type" type="string" length="15"></discriminator>

<property name="name" length="15"></property>

<subclass name="com.javatpoint.mypackage.Regular\_Employee" discriminator-value="reg\_emp">

<property name="salary" length="15"></property>

<property name="bonus" length="15"></property>

</subclass>

<subclass name="com.javatpoint.mypackage.Contract\_Employee" discriminator-value="con\_emp">

<property name="pay\_per\_hour" length="15"></property>

<property name="contract\_duration" length="15"></property>

</subclass>

</class>

</hibernate-mapping>

308)How do you specify a one to many unidirectional relationship with hibernate?

Ans)consider the following example,to implement one to many relation ship.

Public class Parent{

@Id

@GeneratedValue

Private long id;

@OneToMany

Private Set<Child> children;

}

Public class Child{

@Id

@GeneratedValue

Private long id;

Private String name;

}

309)How do you specify a one-to-many bidirectional relationship with hibernate?

Ans)consider the below example,if we want implement bidirectional relation ship,we can use @OneToMany and @ManyToOne annotations.

Public class Parent{

@Id

@GeneratedValue

Private long id;

@OneToMany

Private Set<Child> children;

}

Public class Child{

@Id

@GeneratedValue

Private long id;

Private String name;

@ManyToOne

Private Parent parent;

}

310)How do you implement pagination in hibernate ?

Ans)To implemt the pagination concept,we have to use setFirstResult() and setMaxResults() methods given by the Criteria interface.

Ex:Criteria crit=session.createCriteria(Product.class);

Crit.setFirstResult(50);

Crit.setMaxResult(100);

List crit.list();

311)what is a lazy association ?

Ans)Let’s consider a parent table associated with a corresponding child table.when we load the parent table,should be load content of the child relation ship.in lazy association the child relationship is loaded when it is need.This is default configuration in hibernate.

312)How can you do automatic schema generation from hibernate mappings ?

Ans)if we want generate the schema in oracle database automatically,we need to use hibernate provided ddl commands in our hibernate configuration file.in hibernate there are three commands are available 1)create 2)create-drop 3)update.generally update command is used in our applications.

Ex:<property name=”hibernate.hbm2ddl.auto”>update</property>

313)what is the use of Schema Validator?

Ans)SchemaValidator tool can be used to verify if the mapping configuration matches the existing datadase structure.

314)suggest some Hibernate Best Practices?

Ans)i)identify natural keys.

ii)place each class mapping in its own file.

iii)externalize query strings to make applications more portable.

iv)prefer lazy fetching for associations.

v)use bidirectional associations.

315)what is HQL and what is use it?

Ans)Hql stands for Hibernate Query language,using hql we can perform bulk of operations,that means we can select at time number of records and we can delete at time number of records and we can update all the records at a time.

🡪This is an object oriented type of sql,that means while constructing hql queries we can replace the database table names with pojo class names and database columns we can replace the pojo class property name.

316)How to select the bulk of recods by using HQL?

Ans)Conside the following example,using this code we can fetch the bulk of records.

Session session=factory.openSession();

Query qry=session.createQuery(“from Product p”);

List l=qry.list();

Iterator it=l.iterator();

While(it.hasNext())

{

Product p=(Produt)it.next();

System.out.println(p.getXXX());

}

317)Hibernate 2.x supported by the DML operation or not ?

Ans)Hibernate 1.x and 2.x doesn’t supported by the DML operations.

318)what is a native sql ?

Ans)This is one type of query language,using this language we can migrate the jdbc projects into hibernate is very simple.

Native sql was supported by the ordinary sql queries also and it is supported by the data base dependent queries.

319)what is a Entity query?

Ans)whenever we are adding the addEntity() to existing native sql query then we can say that entity queries.the pupose of entity query is to convert the our resultset object records into pojo class type then we can use entity queries.

For example

SQLQuery qry=session.createSQLQuery(“select \*from employee”).addEntity(Employee.class);

This type of queries we call them entity queries.

320)what is scaler queries in hibernate?

Ans)Generally while select the records from database to front end the hibernate framework was searching the meta information about the data base columns.at the time hibernate takes the help from ResultSetMetaData,when the hibernate is taken the help from ResultSetMetaData the application performance is decreasing.in order to increase the performance of the an application we need to add addScaler method on exiting Native sql query.

Ex:

SQLQuery qry=session.createSQLQuery(“select eid,ename from employee”).addScaler(“eid”,Hibernate.INTEGER).addScaler(“ename”,Hibernate.STRING);

This type of queries we call them as ScalerQueries.

321)what is a projections ?

Ans)projection is one of the concept in ibernate framework,using this concept we can retrieve the partial records from data base to front end.

🡪projection concept is given by the two important things.

i)Projection(i),it is an pre defined interface,available in org.hibernate.criterion.\*;

ii)Projections,is the predefined class available to the org.hibernate.criterion.\*;this class was given by the list of static methods.using those methods we can put the more conditions.

322)How to select the partial records using projections concept ?

Ans)Consider the following example,

Session session = factory.openSession();

Criteria crit = session.createCriteria(Product.class);

ProjectionList pl=Projections.projectionList();

pl.add(Projections.sum("price"));

pl.add(Projections.avg("price"));

pl.add(Projections.rowCount());

crit.setProjection(pl);

List l=crit.list();

Iterator it=l.iterator();

if(it.hasNext())

{

Object[] row=(Object[])it.next();

System.out.println(row[0]+" "+row[1]+" "+row[2]);

}

323)what is the diff b/w Hql and Criteria ?

Ans)

i)Hql is performance wise is slow compared to critera api.

ii)By using Hql we can’t put more conditions on selected data where as using criteria we can put the more conditions.

iii)By using Hql we can perform ddl and dml operations where as by using criteria we can perform only selected operations.

324)what is a named query ?

Ans)Named query is one of the concept in hibernate frame work,using this concept we can perform bulk operations.

🡪Generally this concept is used for instated of writing each and every time we can write in one place and reuse that query,wherever we want.

325)where we can configure the named Query?

Ans)Generally we can configure the named queries in hibernate mapping file,for example.

hibernate.hbm.xml

<DTD…….>

<hibernate-mapping>

<class name="str.Product" table="products">

<id name="productId" column="pid" />

<property name="proName" column="pname" length="10"/>

<property name="price"/>

</class>

<!--<query name="q1">

<![CDATA[delete from Product p where p.price>:durga]]>

</query>-->

<query name="q1">

<![CDATA[select p from Product p where p.price>:durga]]>

</query>

</hibernate-mapping>

326)what is the diff between update method and merge method ?

Ans)Both are used for to updating the data in a database,but the difference is update method is related to particular session where as merge method is related to previous session and current session and combined both and gives to the an updation to database

327)what are the cascade values in hibernate frame work?

Ans)In hibernate mainly the following cascade values are available

1)none (default)

2)save

3)update

4)save-update

5)delete

6)all

328)what is the single row operation and multirow operation?

Ans)By using hibernate frame work we can perform two types of operations

i)single row operations

ii)multi row operations

🡪single row operation means we can perform the operations at time single row.but not multiple rows.where as multirow operation means we can perform at time multiple rows.

329)How many types of caching mechanism are available in hibernate framework ?

Ans)In hibernate mainly there are two types of caching mechanism are available

1)first level caching.

2)second level caching

330)what is the diff between first level caching and second level caching ?

Ans)The main diff between first level caching and second level caching is first level cache is available until we are closing the session object,once we closed session object then automatically first level cache is gone.where as second level cache is available until how much time we want maintaine the objects in buffer area that much of time we can maintaine the objects in buffer area.

331)as an programmer,what are designe principles you focus on ?

Ans)while designing the an applications,as a java programmer we have to follow the following principles.

i)Runs all test cases.

ii)Minimize Duplication

iii)Maximize Clarity.

iv)Keep it small.

332)what are the modern programming practices which lead to very good application ?

Ans)First of all:unit testing and mocking.we are in the age of continuous integration and delivery,and the basic thing that enables those is having a good set of unit test in place.

Second in line is Automated integration tests.Automated integrationtests is the second imported bullet in enabling continuous delivery.

333)what are the typical things you would need to consider while designing the business layer of a Java EE web application ?

Ans)let’s consider the following consideration,

i)Should I have a service layer acting as a faced to the business layer?

ii)Can I separate any of the business logic into separate component or service ?

iii)Do I use a Domain Object Model ?

iv)Do I need caching ?if so at what level?

v)Does service layer need to handle all exceptions ?or shall we leave it to the web layer?

334)what are the things that you would need to consider when designing the access layer of the web application ?

Ans)

i)Do we want to use a jpa based object mapping framewok(hibernate) or query based mapping framework or simple Spring DO?

ii)How do you communicate with external systems?Web services or JMS? If web services,then how do we handle object xml mapping?JAXB or XML Beans?

iii)How do you handle connections to database?these days its an easy answer,leave it to the application server configuration of data source.

iv)ensure that performance and scalability is taken care of in all the decisions you make.

335)what are the things that you would need to consider when designing the web layer?

Ans)

i)first question is do we want to use a modern front end javascript framework like anjularJS? If the answer is yes,most of this discussion does not apply.if the answer is no,then proceed?

ii)should we use a MVC framework like Spring MVC,struts or should we go for a java based framework like wicket or vaadin?

iii)what should be the view technology ? jsp,jsf or Template based(Velocity,free maker)?

iv)Do you want AJAX functionality ?

v)Do we need to expose external web services?

336)what are the important features of IDE Eclipse?

Ans) Enterprise Java Tooling

It's a broad topic, but Eclipse has some of the best tooling available for JEE projects. It's not my usual type of development, but recently I needed to get setup to developer on an enterprise project.

Software Updates

It tends to get more than it's fair share of bad press, but for me, p2, the software update mechanism is essential.

Being able to simply add in an update site, and get the latest version of a framework or plug-in makes the developers life really easy.

337)what are the best practices for build tool maven?

Ans)use archetypes as much as possible.archetypes are good start for generating projects.based on spring,SpringMVC,struts,Hibernate and wide variety of other projects. Also it is a good practice to create maven archtype for the components we create repeatedly.

Some of the maven best practices are,

i)proper dependency mgmt.

ii)Group related dependencies.

iii)exclude test dependencies from final ear.

iv)Have a parent pom.

v)Use profiles as needed.

338)what is a uml ?

Ans) Unified Modeling Language, a standard language for designing and documenting a system in an object-oriented manner. It has nine diagrams which can be used in design document to express design of software architecture.

339) Can you explain primary and secondary actors?

Ans)Actors are further classified in to two types primary and secondary actors. Primary actors are the users who are the active participants and they initiate the user case, while secondary actors are those who only passively participate in the use case.

340) Can you explain class diagrams?

Class diagram

-------------------

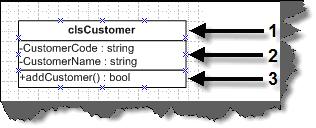
Class is basically a prototype which helps us create objects. Class defines the static structure of the project. A class represents family of an object. By using Class we can create uniform objects.

In the below figure you can see how the class diagram looks. Basically there are three important sections which are numbered as shown in the below. Let’s try to understand according to the numbering:

Class name: This is the first section or top most section of the Class which represents the name of the Class (clsCustomer).

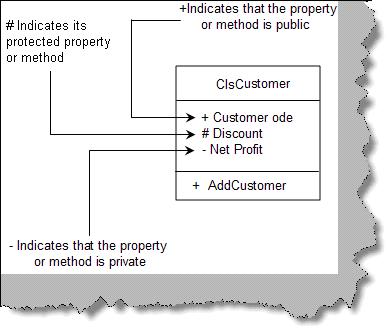
Attributes: This is the second section or the middle section of the class which represents the properties of the system.

Methods: This section carries operation or method to act on the attributes.



341) How do we represent private, public and protected in class diagrams?

Ans)In order to represent visibility for properties and methods in class diagram we need to place symbols next to each property and method as shown in figure ‘Private, Public and Protected’. ‘+’ indicates that it’s public properties/methods. ‘-‘indicates private properties which means it can not be accessed outside the class. ‘#’ indicate protected/friend properties. Protected properties can only be seen within the component and not outside the component.



342)what are the major three types of modeling used?

Ans)major three types of modelings are structural,behavioral,and architectural.

343) What is SDLC?

Ans)SDLC is Software Development Life Cycle. SDLC of a system included processes that are Use case driven, Architecture centric and Iterative and Incremental. This Life cycle is divided into phases. Phase is a time span between two milestones. The milestones are Inception, Elaboration, Construction, and Transition. Process Workflows that evolve through these phase are Business Modeling, Requirement gathering, Analysis and Design, Implementation, Testing, Deployment. Supporting Workflows are Configuration and change management, Project management.

344) What are the different kinds of modeling diagrams used?

Ans)Following modeling diagrams are commonly used.

i)Use case diagram

ii)Class Diagram

iii)Object Diagram

iv)Sequence Diagram

v)State chart Diagram

vi)Collaboration Diagram

vii)Activity Diagram

viii)Component diagram

ix)Deployment Diagram

345) What are Messages?

Ans)A message is the specification of a communication, when a message is passed that results in action that is in turn an executable statement.

346)What is an Use Case?

Ans)A use case specifies the behavior of a system or a part of a system, Use cases are used to capture the behavior that need to be developed. It involves the interaction of actors and the system.

347) an you explain ‘Extend’ and ‘Include’ in use cases?

Ans)‘Extend’ and ‘Include’ define relationships between use cases.

Include: Include relationship represents an invocation of one use case by the other. If you think from the coding perspective its like one function been called by the other function.

Extend: This relationship signifies that the extending use case will work exactly like the base use case only that some new things will inserted in the extended use case.

348) Can you explain sequence diagrams?

Ans)Sequence diagram shows interaction between objects over a specific period time. The message flow is shown vertically in waterfall manner i.e. it starts from the top and flows to the bottom. Dashed lines represent the duration for which the object will be live. Horizontal rectangles on the dashed lines represent activation of the object. Messages sent from a object is represented by dark arrow and dark arrow head. Return message are represented by dotted arrow.

349) What is the waterfall model ?

Ans)The Waterfall Model was first Process Model to be introduced. It is also referred to as a linear-sequential life cycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed fully before the next phase can begin. This type of model is basically used for the for the project which is small and there are no uncertain requirements.

350) What are the Disadvantages of Spiral model?

Ans)The following dis advantages are available in spiral model,

1)Can be a costly model to use.

2)Risk analysis requires highly specific expertise.

3)Project’s success is highly dependent on the risk analysis phase.

4)Doesn’t work well for smaller projects.

351)what are the important responsibilities of architect?

Ans) Architects design residential and commercial structures, of course, but their responsibilities don't stop there. An architect must also think about a building’s style, safety and sustainability to ensure it meets the needs of its occupants as well as the requirements of state and federal regulators. Architects work with both government agencies and private clients.

352)How should an ideal architect be like ?

Ans) Most important qualities I look for in an Architect are

Ans)Impeccable Credibility : Somebody the team looks up to and aspires to be.

Super diagnostic skills : The ability to do a deep dive on a technology issue. When developers are struggling with a problem (having tried different things), Can he/she provide a fresh pair of eyes to look at the same problem?

Forward Thinker and Proactive : Never satisfied with where we are. Identifies opportunities to add value fast.

Great Communication : Communication in the widest sense. Communicating the technical aspects to the stakeholders, project management, software developers, testers, etc.

353)How do you ensure that the Code Quality is maintained?

Ans)More than everything else, code quality is an attitude. Either, the team has it or not. The attitude to refactor when something is wrong. The attitude to be a boy scout. As an architect, it is important to create an environment where such an attitude is appreciated. (There are always bad sheep, who take the code quality to such depth that it is not fun anymore, but I like them more than developers who keep churning out bad code).

354)How do Agile and Architecture go hand in hand?

Ans)First of all I m a great believer that agile and architecture go hand in hand. I do not believe agile means no architecture. I think agile brings in the need to separate architecture and design. For me architecture is about things which are difficult to change : technology choices, framework choices, communication between systems etc. It would be great if a big chunk of architectural decisions are made before the done team starts. There would always be things which are uncertain. Inputs to these can come from spikes that are done as part of the Done Scrum Team.But these should be planned ahead.

355) How do you ensure the team is following sound engineering practices?

Ans)I ask the following questions:

How often is code committed?

How often is code released?

How often do builds break? Are they immediately fixed?

How often is code deployed?

What steps are part of continuous integration build? Is deployment and automation suite part of it?

Does the team develop vertical slices when implementing a new functionality?

356)what are the new features in java 5?

Ans)

i)Generics

ii)Enhanced for loop

iii)Autoboxing/un boxing

iv)Typesafe Enums

v)Varargs

vi)static imports

vii)concurrent collections

viii)copy on write

ix)compare and swap

x)locks

357)what are the new features in java 6?

Ans)java 6 has very few important changes in terms of api’s.there are a few performance improvements but none significant enough to deseve a mention.

358)what are the new features in java 7?

Ans)New features in java 7 are:

i)Diamond Operator.Example:Map<String,List<Trade>> trades=new TreeMap<>();

ii)using String in switch statement

iii)Automatic resource management:try(resources\_to\_be\_cleant){//your code}

iv)Numeric literals with underscores.

v)improved exception handling:multiple catches in same block-catch(ExceptionOne|ExceptionTwo|ExceptionThree e).

vi)File change notifications.

359)what are the new features in java 8?

Ans)The new features in java 8 are

Lamda Expressions.

Nashorn:javascript engine that enables us to run javascript to run on a jvm.

String.join() function

Streams.

360)what is a web service ?

Ans)Simplest definition is a service offered over the web(internet).whenever we browse the internet,we are making use of innumerable web services.Even the simplest attempt to load a google.com websites involves web services.

When we type in google.com in the browser,following steps happen in the back ground.

i)Browser invokes a get request on google.com.

ii)google.com returns a Http response with the Html content to show.

iii)Browser understands the html content and renders it for the user.

361)what are the important parts of a web service?

Ans)In web service mainly three important parts are available,

i)Service Provider:Google.com is the service provider,handles the request and sends a response back.

ii)Service Consumer:Browser is the service consumer,Creates request.invokes service and processes the response.

iii)Data exchange format.i.e SOAP and JSON.

362) What are the advantages of web services?

Ans)Interoperability: By the help of web services, an application can communicate with other application developed in any language.

Reuability: We can expose the web service so that other applications can use it.

Modularity: By the help of web service, we can create a service for a specific task such as tax calculation etc

363)What are the different types of web services?

Ans)There are two types of web services:

i)SOAP

ii)RESTful

364)what are SOAP web services ?

Ans) 🡪 it is a independent component, Not a technology, is a message format to be exchanged b/w a SR & SProv

🡪 Exchanging structure msgs b/w (xml docu) b/w a SR & SResource

🡪 SOAP is an XML document to exchange the data b/w SR & SP in the form of messages.

SOAP is a kind of XML document.

SOAP is not a protocol, it is a message format

SOAP is defined as message format for exchanging data in a distributed environment in the form of messages.

🡪 SOAP specification is given by W3C

🡪 SOAP first version 1.1 & current version is 1.2

SOAP messages are transferred as an internal part of HTTP protocol . so there is no fire wall problem for SOAP message.

A SOAP request is message, is inserted into body part of http request document and then that http request document will be sent to the server by http protocol.

365)what is a WSDL ?

Ans)WSDL is an acronym for Web Services Description Language.

WSDL is a xml document containing information about web services such as method name, method parameter and how to access it.

WSDL is a part of UDDI. It acts as a interface between web service applications.

WSDL is pronounced as wiz-dull.

366)what is marshalling and unmarshalling? How do JAXB and XML Beans help with this?

Ans) JAX B🡪 introduced Jdk -1.5 (1.1)

2.0🡪 contains annotations (Jdk.16 with annotations)

JAX B uses🡪 it is easy to convert java obj to xml & xml is convert into java obj.

In JAX B 🡪 marshalling🡪 convert java obj to java obj Java Architctur for XML Binding (JAXB)

* JAXB is used for converting a java object into an xml file or an xml file to a java object.
* 🡪 JAXB is introduced in jdk1.5 and JAXB 2.0 is made vary simple with annotations and introduced in jdk.16
* Before JAXB java developers used the xml parsr API’s like SAX & DOM for reading the xml files.
* Working with SAX & DOM API’s is complex, so SUN introduced JAXB is an alternate for SAX & DOM api’s
* In JAXB converting a java object into an xml file is called marshalling and converting an xml file into a java object is called un-marshalling

367)what are the java frameworks that you can use to build web services?

Ans)One frequently used framework is Spring Web Services(also called Spring WS).

368)How do you handle security for web services?

Ans)security for web services has to be handled at two levels,

i)Transport level

ii)Application level.

🡪at transport level,ssl is used to exchange certificates.this ensure that the server and client are mutually authenticated. It is possible to use one way ssl authentication as well.

🡪At the application level, security is implemented by transferring encrypted information in the message header.This helps the server to authenticate the client and be confident that the message has not been tampered with.

369)what are the opular java specifications for web services?

Ans)popular java specifications for SOAP web services is JAX-WS.Rest web service specifications are JAX-RS.

370)what is service oriented architecture(SOA)?

Ans)service Oriented is an architectural style where applicactions are built on top of language neutral,loosely coupled,independent,reusable components.

Major components of SOA are

i)User interface layer:user interface for web/desktop applications.

ii)Business process layer:This represents the layer where the use cases are on top of the services.

iii)service:services and components used to build the services.such as various frameworks.

371)what is REST web service?

Ans) 🡪Rest is an architectural structure.

🡪Restful web services, which are accessable over web.

🡪A Restful web services is a collection of resources,where each resource is identified through a unique uri and which are accessable using that uri.

🡪A Restful web services can be accessed directly from the browser or by creating a client application in different languages.

🡪Representation is a data format like text/html,text/xml,application/json etc…..and state transfer is nothing but transfering the data with representation to the client.

372)what are important constraints for a RESTful Web Services?

Ans) The five important constraints for RESTful Web Service are

Client - Server : There should be a service producer and a service consumer.

The interface (URL) is uniform and exposing resources. Interface uses nouns (not actions)

The service is stateless. Even if the service is called 10 times, the result must be the same.

The service result should be Cacheable. HTTP cache, for example.

Service should assume a Layered architecture. Client should not assume direct connection to server - it might be getting info from a middle layer - cache.

373)what is Richardson maturity model ?

Ans) Richardson Maturity Model defines the maturity level of a Restful Web Service. Following are the different levels and their characteristics.

Level 0 : Expose SOAP web services in REST style. Expose action based services (http://server/getPosts, http://server/deletePosts, http://server/doThis, http://server/doThat etc) using REST.

Level 1 : Expose Resources with proper URI’s (using nouns). Ex: http://server/accounts, http://server/accounts/10. However, HTTP Methods are not used.

Level 2 : Resources use proper URI's + HTTP Methods. For example, to update an account, you do a PUT to . The create an account, you do a POST to . Uri’s look like posts/1/comments/5 and accounts/1/friends/1.

Level 3 : HATEOAS (Hypermedia as the engine of application state). You will tell not only about the information being requested but also about the next possible actions that the service consumer can do. When requesting information about a facebook user, a REST service can return user details along with information about how to get his recent posts, how to get his recent comments and how to retrieve his friend’s list.

374)what are the best practices in designing REST ful Api?

Ans) While designing any API, the most important thing is to think about the api consumer i.e. the client who is going to use the service. What are his needs? Does the service uri make sense to him? Does the request, response format make sense to him?

In Rest, we think Nouns (resources) and NOT Verbs (NOT actions). So, URI’s should represent resources. URI’s should be hierarchical and as self descriptive as possible. Prefer plurals.

Always use HTTP Methods. Best practices with respect to each HTTP method is described in the next question.

375)what are the best practices in using Http methods with Restful web services?

Ans) GET : Should not update anything. Should be idempotent (same result in multiple calls). Possible Return Codes 200 (OK) + 404 (NOT FOUND) +400 (BAD REQUEST)

POST : Should create new resource. Ideally return JSON with link to newly created resource. Same return codes as get possible. In addition : Return code 201 (CREATED) is possible.

PUT : Update a known resource. ex: update client details. Possible Return Codes : 200(OK)

DELETE : Used to delete a resource.

376)can you explain a little bit about jax-rs?

Ans) JAX-RS is the JEE Specification for Restful web services implemented by all JEE compliant web servers (and application servers).

Important Annotations:

@ApplicationPath("/"). @Path("users") : used on class and methods to define the url path.

@GET @POST : Used to define the HTTP method that invokes the method.

@Produces(MediaType.APPLICATION\_JSON) : Defines the output format of Restful service.

@Path("/{id}") on method (and) @PathParam("id") on method parameter : This helps in defining a dynamic parameter in Rest URL. @Path("{user\_id}/followers/{follower\_id}") is a more complicated example.

@QueryParam("page") : To define a method parameter ex: /users?page=10.

Useful methods:

Response.OK(jsonBuilder.build()).build() returns json response with status code.

Json.createObjectBuilder(). add("id",user.getId()); creates a user object.

377)what are the advantages of Restful web services?

Ans) Lightweight : Easy to consume from mobile devices also.

Easy to expose : Little or no restrictions on output format and communication protocol.

Most Restful services use HTTP protocol : Entire web is based on HTTP and is built for efficiency of HTTP. Things like HTTP caching enable Restful services to be effective.

High Performance : Less xml & soap overhead and More caching enable Restful services to be highly performant.

378)what is the difference between REST and SOAP Based services ?

Ans)1)The difference between rest and soap based is,when we are using restfull web service we can reduce the lot of xml configuration.where as when we are using soap it needs lot of xml configurations

2)Rest is given to the more performance,where as soap is very less performance.

Ans) First of all, REST is a set of architectural principles defining how a RESTful service should look look like. SOAP is a message exchange format. SOAP defines the structure of message to exchanged. How should the header be? How should the request content be? So, there is no real comparison between REST and SOAP.

To get a real comparison, I compare two popular implementation of these concepts.

Restful Sample Implementation : JSON over HTTP

SOAP Sample Implementation : XML over SOAP over HTTP

All comparison is between the Sample Restful and SOAP implementations described above.

REST is built over simple HTTP protocol. SOAP services are more complex to implement and more complex to consume.

REST has better performance and scalability. REST reads can be cached, SOAP based reads cannot be cached.

REST permits many different data formats (JSON is the most popular choice) where as SOAP only permits XML.

SOAP services have well defined structure and interface (WSDL).

SOAP is based on well defined standards (WS-Security, WS-AtomicTransaction and WS-ReliableMessaging).

379) Explain what is Java Design Pattern?

Ans)A design pattern is a language independent strategies for solving common object oriented design problem. It describes how to structure classes to meet a given requirement.

There are three types of design patterns are there

i)Creational patterns:concered with creation of Objects.prototype,singleton,Builder..etc.

ii)Structural pattern:concerened with structure of an objects and the relation ship between them,Decorator,Facade,Adapter..etc

iii)Behavioural pattern:concered with interaction between objects.strategy,Template method etc.

380)Explain what is creational design patterns and Factory pattern?

Ans)Creational design pattern: This pattern is used to define and describe how objects are created at class instantiation time.

Factory pattern: The factory pattern is used to create an object without exposing the creation logic to the client and refer to a newly created object using a common interface.

381) Explain in singleton pattern whether it is better to make the whole getinstance() method synchronized or just critical section is enough? Which one is preferable?

Ans)Synchronization of whole getinstance() method is costly and is only needed during the initialization on singleton instance, to stop creating another instance of Singleton. Therefore it is better to only synchronize critical section and not the whole method.

382)what is a structural designe pattern?

Ans)Structural design patterns are concerned with how classes and objects can be composed, to form larger structures.

The structural design patterns simplifies the structure by identifying the relationships.

These patterns focus on, how the classes inherit from each other and how they are composed from other classes.

Types of structural design patterns

There are following 7 types of structural design patterns.

Adapter Pattern

Adapting an interface into another according to client expectation.

Bridge Pattern

Separating abstraction (interface) from implementation.

Composite Pattern

Allowing clients to operate on hierarchy of objects.

Decorator Pattern

Adding functionality to an object dynamically.

Facade Pattern

Providing an interface to a set of interfaces.

Flyweight Pattern

Reusing an object by sharing it.

proxy Pattern

Representing another object.

383) what is a behavioral designe pattern?

Ans) In software engineering, behavioral design patterns are design patterns that identify common communication patterns between objects and realize these patterns. By doing so, these patterns increase flexibility in carrying out this communication.

Behavioral patterns are concerned with the assignment of responsibilities between objects, or, encapsulating behavior in an object and delegating requests to it.

Behavioral Design Patterns

Chain of Responsibility

Command

Interpreter

Iterator

Mediator

Memento

Observer

State

Strategy

Visitor

Template Method

384) What are the general programming practices that are important with respect to performance in Java?

Ans)First and Foremost - NO premature optimizations. Any optimization decision should be based on numbers or past experience. Programmers waste enormous amounts of time thinking about, or worrying about, the speed of non critical parts of their programs, and these attempts at efficiency actually have a strong negative impact when debugging and maintenance are considered. We should forget about small efficiencies, say about 97% of the time: premature optimization is the root of all evil. Yet we should not pass up our opportunities in that critical 3%."

Minimise number of objects created:

Avoid String Concatenation : Use StringBuffer.

Avoid creating objects in Loops.

Consider patterns like Flyweight.

Use correct data structures:

Use the right collection for a situation.

Use a proper domain model.

Reduce web application overhead:

Small session size.

Use Caching where appropriate.

Close connections and Streams.

Tune you database:

Have indexes.

Tune your queries.

If you are using hibernate, understand the internals of hibernate. Avoid N+1 Selects Problem.

Enable statistics on Databases.

385) What are the best practices in Load & Performance Testing?

Ans)Following are the best practices in terms of load and performance testing.

Have clear performance objectives. That’s the single most important objective. Decide Peak Load, Expected Response Time, Availability Required before hand.

An application does not work on its own. It connects with a number of external interfaces. Establish clear performance expectations with the Interface Services

The next important thing is to ensure that you mirror your production environment. A load testing environment should be the same as your production environment. We will discuss the exact factors involved later in this article.

Validate early : Do performance testing as early as possible.

Make it a regular practice to use profilers in development environment. ex:JProfiler

Make sure team is not making premature optimizations. Any optimization decision should be based on numbers or past experience. In Donald Knuth's paper "Structured Programming With GoTo Statements", he wrote: "Programmers waste enormous amounts of time thinking about, or worrying about, the speed of non critical parts of their programs, and these attempts at efficiency actually have a strong negative impact when debugging and maintenance are considered. We should forget about small efficiencies, say about 97% of the time: premature optimization is the root of all evil. Yet we should not pass up our opportunities in that critical 3%."

Have Clear Strategy on How to Handle expected load. What would be the initial settings on the application server? Do you plan to use a clustered environment? Do you need a load balancer?

386) What are the important factors to consider to ensure while building up for the Load Test?

Ans)A Load test environment should mirror production environment as much as possible:

Application Configuration

Application Server Configuration : Datasource properties (connections etc), JVM Memory settings, etc.

Test Scenarios should mirror production usage. Load on different screens should mirror the usage in production.

Ensure that the user think time is taken into consideration in the load test script.

Consider the delays in interacting with other interfaces. If you are using stubs for interfaces, include the delay in.

All parts of the planned production environment (like Load Balancer) should be included.

Have same amount of data in the database as you have in production environment.

387) What are the important results from Profiling in Development environment?

Ans)JProfiler is a good profiling tool. The main result we expect to find from profiling is to identify the parts of the application where most request time is spent? Focus on the parts of the request which consume more than 5-10% of request time.

Waiting for connection?

Waiting for response from External Interface?

Running a query on the database?

Some loop on the application server?

388) Can you list some important features of JProfiler?

Ans)Important features are

Memory profiling

Heap Walker : See what are the objects in the Heap.

CPU profiling : Call tree ,HotSpots - most time consuming methods list & Method statistics

Thread profiling : Thread dumps

Monitor profiling : all waiting and blocking situations in the JVM

Telemetry views i.e. Graphs : Heap,Throughput, GC activity, CPU load & Database

389) What are the important features of Java Mission Control (Formerly JRockit Mission Control)?

Ans)Low overhead (even in production environment)

Captures: garbage collection pauses, memory and CPU usage, heap statistics

390) What are the important components in having a clear strategy to handle expected load?

Ans)Clear Deployment Topology

Initial Caching Strategy

Application Server : Max Memory and Min Memory Settings - Have a clear strategy on how to play around with these?

Database Connections - Statement Cache Size, Max Connections

391) What are the actions to reduce bottlenecks in an application?

Ans)Reduce demand

Introduce Caching.

Tuning Java Code.

Tuning Database. (Indexing, Optimizing Queries, Optimize Hibernate settings)

Tuning Application Server Configuration and Settings (Connection, Memory, GC etc).

Increase available resources

Horizontal or Vertical Scaling

More Memory

Better CPU

Reduce slowdown due to Synchronization

More effective collections

More effective locking.

392) What are the websphere tools available for performance tuning and bottleneck analysis?

Ans)Thread and Monitor Dump Analyzer for Java

Analyzing Java core files.

Finds Hangs, Deadlocks, Resource contention & Bottlenecks.

Garbage Collection and Memory Visualizer

Analyzing and visualizing verbose GC logs.

Flag possible memory leaks, Size the Java heap correctly, Select the best garbage collection policy.

HeapAnalyzer

Analyse Heap Dumps to find memory leaks.

PMI (Performance Monitoring Infrastructure)

Can be switched in the websphere admin console.

Results can be viewed in Tivoli Performance Viewer (TPV)(WAS admin console)

Monitors JDBC Connection Pools, JVM Runtime. HeapSize, Request Count, Average time taken by servlet etc

393)what is continuous Integration ?

Ans) Continuous Integration can be defined as “Building software and taking it through as many tests as possible with every change”.

394)Why is Continuous Integration important?

Ans)Two important reasons:

Defects found early cost less to fix : When a defect is found immediately after a developer codes it, it takes 10x times less time to fix it compared to finding the defect a month later.

Reduced Time to Market : Software is always tested. So, it is always ready to move to further environments.

395) How is Continuous Integration Implemented?

Ans)Different tools for supporting Continuous Integration are Hudson, Jenkins and Bamboo. Jenkins is the most popular one currently. They provide integration with various version control systems and build tools.

396) What are the success factors for Continuous Integration?

Ans)Implementing the tools for Continuous Integration is the easy part. Making best use of Continuous Integration is the complex bit. Are you making the best use of your continuous integration setup? Here are the things you would need to consider.

How often is code committed? If code is committed once a day or week, the CI setup is under utilised. Defeats the purpose of CI.

How is a failure treated? Is immediate action taken? Does failures promote fun in the team?

What steps are in continuous integration? More steps in continuous integration means more stability.

Compilation

Unit Tests

Code Quality Gates

Integration Tests

Deployment

Chain Tests

More steps in continuous integration might make it take more time but results in more stable application. A trade-off needs to be made.

Run Steps a,b,c on a commit.

Run Steps d & e once every 3 hours.

How long does a Continuous Integration build run for?

One option to reduce time taken and ensure we have immediate feedback is to split the long running tests into a separate build which runs less often.

397) What are the different things to consider regarding security of a web application?

Ans)Security related consideration can be split into these parts

User Authentication and Authorization

Web Related Issues

External Interfaces

Infrastructure Related Security

398) What are the important things to consider regarding user authentication and authorization?

Ans)Following are the important considerations:

Proper separation of authenticated and unauthenticated resources. These can be split into separate deployable units if possible.

Proper use of filters to ensure that the configuration for authenticated resources is centralized.

Use a proper framework like Spring Security to implement authorization.

399) What are the important factors to consider when exposing an application to Internet?

Ans)OWASP (Open Web Application Security Project) is normally a great starting point. Important factors to consider are

Validaton of user data : Ensure they are validated also in Business Layer.

SQL Injection : Never build sql queries using string concatenation. Use a Prepared Statement. Even better, use Spring JDBCTemplate or frameworks like Hibernate, iBatis to handle communication with database.

XSS - Cross Site Scripting : Ensure you check against a white list of input characters.

Avoid using Old versions of software

400) What are the Best Practices regarding handling security for a web application?

Ans)Best practices are:

Threat Modelling : Do threat modelling and understand the various security threats posed to the application

Static Security Analysis : Use a static security analysis tool like Fortify.

Educate Developers and Testers : Most important part. Developers and Testers should be aware of the latest security threats.

Dynamic Security Tests : Dynamic security tests done by a professional security testing team should be an important part of the release cycle. It is preferable to do this as early as possible.

401)What are the important features of Spring Batch?

Ans)Restartability : Easy to restart a batch program from where it failed

Different Readers and Writers : Provides great support to read from JMS, JDBC, Hibernate, iBatis etc. It can write to JMS, JDBC, Hibernate and more.

Chunk Processing : If we have 1 Million records to process, these can be processed in configurable chunks (1000 at a time or 10000 at a time).

Easy to implement proper transaction management even when using chunk processing.

Easy to implement parallel processing. With simple configuration, different steps can be run in parallel

**ALL THE BEST………………..!**