**Core java Task**

**Date 3/1/2018**

1. **Can abstract class have constructors in Java?**

**Ans :- yes**

1. **Can abstract class implements interface in Java? do they require to implement all methods?**

**Ans:- yes abstract class can implements interface in java but It is the responsibility of the first concrete class that has that abstract class as an ancestor to implement all of the methods in the interface.**

1. **Can abstract class be final in Java?**

**Ans**:- **no.**

**Because if you make an abstract class final class in java then**

* **must be extended in order to instantiated.**
* **can not be extended because it is final.**
* **Its contrast not allowing java to make abstract class as a final**

1. **Can abstract class have static methods in Java?**

**Ans:- yes but is not abstract with static its just static like**

**Static float areaCircle(int r)**

**{**

**Statements;**

**}**

1. **Can you create instance of abstract class?**

**Ans:- no. because it can not contain always complete implementation.**

1. **Is it necessary for abstract class to have abstract method?**

**Ans:- no its 0 to 100% abstraction so normal & static both method can be inside abstract class.**

1. **Difference between abstract class and interface in Java?**

**Ans :- Abstract class**

**\* abstract class have abstract & non abstract methods**

**\* abstract class can not support multiple inheritace**

**\* abstract class have final,nofinal,static ,non static variable**

**\* abstract class provides implementation of interface**

**\* its declare with abstract key word**

**\* its not always 100% abstraction**

**Interface**

**\* it can only hold abstract method but now static & default method is possible after jdk 8**

**\* supports multiple inheritance**

**\* only static & final variable possible**

**\* interface keyword used to declare interface**

**\*100% abstraction achieved using interface.**

1. **When do you favor abstract class over interface?**

**Ans:- depends on requirement & situation t use need to use interface & abstract class.**

1. **What is abstract method in Java?**

**Ans:-method has only signature at time of declaration called abstract method which is need to implements wherever we inherit purpose is hiding implementation .**

1. **Can abstract class contains main method in Java ?**

**Ans:-yes main is static method no need of object to call static method so abstract class can contain main methods.**

1. **what is static block in java?**

**Ans:-static block execute when class will load in javm its use in jdbc mostly**

**Class A**

**{**

**Int a;**

**Static**

**{**

**a=100;**

**System.out.println(“static bllock called …….”);**

**}**

**Public static void main(String args[])**

**{**

**System.out.println(A.a);**

**}**

**}**

1. **What is the need of static block?**

**Ans:- used to initialize static variable in java.**

1. **Can we overload static methods in java?**

**Ans:-it can be overload but can not override.**

1. **Can we call super class static methods from sub class?**

**Ans:-static will be called on base of type reference not the object created.**

1. **What is the difference between final and static keywords?**

**Ans:- static belongs to the class not instance this means there will be one copy of variable/method to be shared between instance of particular class.**

**Final is used to declare constant that values can not change entire finale method can not override as well fianal class ca not inherit.**

1. **Write a note on covariant return type with example code.**

**Ans:-covariant return type specifies that return type may difference from same direction as the subclass**

**Class Demo**

**{**

**Demo get()**

**{**

**return this;**

**}**

**}**

**Class MainDemo extends Demo**

**{**

**MainDemo get()**

**{**

**return this;**

**}**

**Void print()**

**{**

**System.out.println(“this is maindemo class method print ”);**

**}**

**Public static void main(String args[])**

**{**

**New MainDemo().get().print();**

**}**

**}**

1. **Write a note on Enum with example code.**

**Ans:-**

**enum in java is user define datatype which contains fix number or set of constants.**

**Java enum is public static final implicitly.**

**Enum is used for making weekday ,months that type of constants which will never change in their terms.**

**Feature of enum**

**Improve type safty**

**Can be used in switch**

**We can traversed it without using collection framework.**

**Enum is class type it has fields,methods,constructor**

**It may implements interface but cannot extends any class because by default it has Enum class extension from Wrapper.**

**Example :-**

**Class EnumDemo**

**{**

**enum Week{**

**WEEK1(5),WEEK2(4),WEEK3(6),WEEK4(5);**

**}**

**private int I;**

**private Week(int a)**

**{**

**i=a;**

**}**

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

**{**

**for(Week w:Week.values())**

**{**

**System.out.println(w.value);**

**}**

**}**

**}**

1. **Write a note on use of super keyword and super() method.**

**Ans:- super keyword**

**Used as a reference variable which is used to refer parent class instance variable,constructor & Methods.**

**Super() method**

**Used to call default constructor JVM call it implicitly whenever inheritance happen using supe method.**

1. **Write a code to implement abstraction using interface.**

**Ans:-**

**Interface exDemo**

**{**

**Static int count;**

**Static{count=0;}**

**Void display();**

**Default void print()**

**{**

**System.out.println(“I am default method of interface exDemo”);**

**}**

**Static void msg()**

**{**

**System.out.println(“I am static msg method of interface exDemo”);**

**}**

**}**

**Class DemoMain implements exDemo**

**{**

**Void dataCount()**

**{**

**count++;**

**System.out.println(count);**

**}**

**Void disply()**

**{**

**count++;**

**System.out.println(count);**

**}**

**Public static void main(String args[])**

**{**

**exDemo d=new DemoMain();**

**d.display();**

**d.print();**

**d.dataCount();**

**DemoMain.msg();**

**}**

**}**

**20)Write a Java program to sort a numeric array and a string array.**

**Ans:-**

**import java.util.Arrays;**

**public class SortIntArray**

**{**

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

**{**

**int[] arr = {13, 7, 6, 45, 21, 9, 2, 100};**

 String[] strNames = **new** String[]{"John", "alex", "Chris", "williams", "Mark", "Bob"};

Arrays.sor(strNames );

For(String s:strNames)

{

System.out.println(s);

}

**Arrays.sort(arr, 1, 5);**

**System.out.printf("Modified arr[] : %s",**

**Arrays.toString(arr));**

**}**

**}**

**21)Write a Java program to sum values of an array.**

**Ans:-**

**Import java.uti.Scanner;**

**Class Sum**

**{**

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

**{**

**Scanner sc=new Scanner(System.in);**

**int a[]=new int[5];**

**int sum;**

**for(int i=0;i<5;i++)**

**{**

**a[i]=sc.nextInt();**

**}**

**for(int i=0;i<5;i++)**

**{**

**Sum=sum+a[i];**

**}**

**System.out.println(“sum => ”sum);**

**}**

**}**

**22)Write a Java program to remove a specific element from an array.**

**Ans:-**

**import java.util.Scanner;**

**public class DeleteArrayElement**

**{**

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

**{**

**int size, i, del, count=0;**

**int arr[] = new int[50];**

**Scanner scan = new Scanner(System.in);**

**System.out.print("Enter Array Size : ");**

**size = scan.nextInt();**

**System.out.print("Enter Array Elements : ");**

**for(i=0; i<size; i++)**

**{**

**arr[i] = scan.nextInt();**

**}**

**System.out.print("Enter Element to be Delete : ");**

**del = scan.nextInt();**

**for(i=0; i<size; i++)**

**{**

**if(arr[i] == del)**

**{**

**for(int j=i; j<(size-1); j++)**

**{**

**arr[j] = arr[j+1];**

**}**

**count++;**

**break;**

**}**

**}**

**if(count==0)**

**{**

**System.out.print("Element Not Found..!!");**

**}**

**else**

**{**

**System.out.print("Element Deleted Successfully..!!");**

**System.out.print("\nNow the New Array is :\n");**

**for(i=0; i<(size-1); i++)**

**{**

**System.out.print(arr[i]+ " ");**

**}**

**}**

**}**

**}**

**23)Write a Java program to reverse an array of integer values.**

**Ans:-**

**Class ReverseArray{**

**static void rvereseArray(int arr[], int start, int end)  {**

**int temp;**

**if (start >= end)**

**return;**

**temp = arr[start];**

**arr[start] = arr[end];**

**arr[end] = temp;**

**rvereseArray(arr, start+1, end-1);**

**}**

**static void printArray(int arr[], int size)**

**{**

**int i;**

**for (i=0; i < size; i++)**

**System.out.print(arr[i] + " ");**

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

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

**int arr[] = {1, 2, 3, 4, 5, 6};**

**printArray(arr, 6);**

**rvereseArray(arr, 0, 5);**

**System.out.println("Reversed array is ");**

**printArray(arr, 6);**

**}**

**}**

**24)Write a Java program to find the duplicate values of an array of integer values.**

**import java.util.Arrays;**

**public class DemoArray {**

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

**{**

**int[] my\_array = {11,11,34,89,90,33,33,10};**

**for (int i = 0; i < my\_array.length-1; i++)**

**{**

**for (int j = i+1; j < my\_array.length; j++)**

**{**

**if ((my\_array[i] == my\_array[j]) && (i != j))**

**{**

**System.out.println("Duplicate Element : "+my\_array[j]);**

**}**

**}**

**}**

**}**

**}**