

Topic: Introduction

Subtopics:

- 1) *Introduction to Programming Languages*
- 2) *Need to learn Programming Languages*
- 3) *Concept of Assembler, Compiler, Interpreter*
- 4) *C vs C++ vs Java*
- 5) *What is Java*
- 6) *History of Java*
- 7) *JDK,JRE,JVM*
- 8) *How To set Path*
- 9) *Installation of Eclipse*

C/W Assignments:

Tip: Trainors will assist for these questions.

1. Installation of Eclipse.
2. How to Use Notepad, Eclipse Editor
3. How to set path.

H/W Assignments:

Tip: Trainors will not assist for these questions.

1. Installation of Eclipse.
2. How to set path.

interview Questions:

- 1) What are Programming languages?
- 2) What is the difference between compiler and interpreter?
- 3) What are the differences between C, C++ and Java?
- 4) What do you know about Java?
- 5) When and by whom was Java Developed?
- 6) What is the difference between JDK, JRE, and JVM?

Topic: Basics Of Java

Subtopics:

- 1) *Features of Java*
- 2) *What is a program*
- 3) *Comments, Data Types*
- 4) *Variables, Keywords*
- 5) *Literals, Identifiers*
- 6) *Operators*
- 7) *program structure*
- 8) *Hello world program*

C/W Assignment:

Tip: Trainors will assist for these questions.

1. Write a program to enter two numbers and perform sum of two numbers.
2. Write a program to enter P, T, and R and calculate Simple Interest.
3. Write a program to convert days into years, weeks and days. {Hint: Input-373 days Output-1 Year, 1 Week, 1 day}
4. Write a program to enter P, T, and R and calculate Compound Interest.
5. Guess the output (**Feb Monthly**)

```
int j = 2;  
System.out.println(j++ - ++j + j-- - j++ + ++j - j-- + --j + j + j-- - --j + --j + j++);
```

H/W Assignment:

Tip: Trainors will not assist for these questions.

1. Write a program to calculate area of an equilateral triangle.
2. Write a program to enter marks of five subjects and calculate total, average and percentage.

Interview Questions:

- 1) What gives Java its 'write once and run anywhere' nature?
- 2) Why is Java platform independent?
- 3) Explain the structure of a program.
- 4) Why comments are used in a program?
- 5) What are data types?
- 6) What are variables? What are the rules to name a variable?
- 7) What are keywords? Name few.
- 8) What are identifiers and literals?
- 9) What are operators?
- 10) Name different types of operators.
- 11) Explain public static void main (String args[]).
- 12) Can we execute a program without main () method?
- 13) What are the types of relational operators in Java?
- 14) What are the logical operators in Java?
- 15) What are called as Arithmetic operators?
- 16) State the name of / and % operator and explain how will you use this?

- 17) Which are the two relational operators that are called as equality operators? Give an example.
- 18) How will you represent increment and decrement operators?
- 19) What is a ternary operator? Give an example.
- 20) State the use of instance of operator. Give an example.
- 21) Discuss the significance of the Java language.
- 22) Elaborate briefly upon the evolution of Java.
- 23) Compare in brief C/C++, C#, and Java.
- 24) Elaborate on the main features of Java.
- 25) What is the difference between the prefix increment ++x and the postfix increment x++?
- 26) What is a Bytecode?
- 27) What are the various Operators used in Java?
- 28) What is a White space in Java?

Topic: Control Statements

Sub-Topics:

- 1) *Types of Control statements*
- 2) *Introduction to Conditional Statements*
- 3) *Use of Scanner class.*
- 4) *Conditional statements:*
 - i. *If*
 - ii. *If-else*
 - iii. *Nested if-else*
 - iv. *If-else if*
 - v. *Switch*
- 5) *Ternary operator*

C/W Assignment:

Tip: Trainors will assist for these questions.

- 1) How to accept different types of data using Scanner class.
- 2) Write a program to input week number and print week day.
- 3) Write a program to input basic salary of an employee and calculate its Gross salary according to following:
 - a. Basic Salary \leq 10000 : HRA = 20%, DA = 80%
 - b. Basic Salary \leq 20000 : HRA = 25%, DA = 90%
 - c. Basic Salary $>$ 20000 : HRA = 30%, DA = 95%
- 4) Write a program to input electricity unit charges and calculate total electricity bill according to the given condition:
 - a. For first 50 units Rs. 0.50/unit
 - b. For next 100 units Rs. 0.75/unit
 - c. For next 100 units Rs. 1.20/unit
 - d. For unit above 250 Rs. 1.50/unit
 - e. An additional surcharge of 20% is added to the bill

H/W Assignment:

Tip: Trainors will not assist for these questions.

- 1) Write a program to find maximum among three numbers.
- 2) Write a program to check whether a number is even or odd.
- 3) Write a program to check whether a number is negative, positive or zero.
- 4) Display all even and odd numbers from 1 to 100.
- 5) Write a program to find maximum and minimum between two numbers.
- 6) Write a program to input any alphabet and check whether it is vowel or consonant.

- 7) Write a program to check whether a character is uppercase or lowercase alphabet.
- 8) Write a program to check whether the triangle is equilateral, isosceles or scalene triangle.
- 9) Write a program to calculate profit or loss.
- 10) Write a program to find all roots of a quadratic equation.
- 11) Write a program to create simple calculator using switch case.
- 12) Write a program to input any character and check whether it is alphabet, digit or special character.
- 13) Write a program to check whether a year is leap year or not.
- 14) Write a program to input marks of five subjects Physics, Chemistry, Biology, Mathematics and Computer.

Calculate percentage and grade according to following:

- a. Percentage $\geq 90\%$: Grade A
- b. Percentage $\geq 80\%$: Grade B
- c. Percentage $\geq 70\%$: Grade C
- d. Percentage $\geq 60\%$: Grade D
- e. Percentage $\geq 40\%$: Grade E
- f. Percentage $< 40\%$: Grade F

Interview Questions:

- 1) What are control statements?
- 2) What are conditional statements?
- 3) Explain different types of conditional statements.
- 4) What is the difference between if else if and switch?
- 5) Explain the use of break.
- 6) What is ternary operator? Give an example.
- 7) What are called Decision statements in Java?
- 8) How will you legally define "if else" statement. Give an example.
- 9) What is the use of "break" and "continue" statements?
- 10) What do the break and continue statements do?

Topic: Iterations

Sub-Topics:

- 1) *Introduction*
- 2) *While*
- 3) *Do-while*
- 4) *For*
- 5) *Continue*

C/W Assignment:

Tip: Trainors will assist for these questions.

- 1) WAP to generate Fibonacci series in forward and reverse order. Take number of terms to print from console. Series for 8 terms will be as follows : 0, 1, 1, 2, 3, 5, 8, 13 and 13, 8, 5, 3, 2, 1, 1, 0 (**Feb Monthly**).
- 2) Write a Java program to find power of any number x^y .
- 3) WAP to print following Pattern

```
* * * * *
* * * *
* * *
* *
*
```

- 4) WAP to print following Pattern

```
1 2 3 4 5
2 3 4 5
3 4 5
4 5
5
4 5
3 4 5
2 3 4 5
1 2 3 4 5
```

- 5) Write a program to create following number series. Print first n numbers. Accept n from user.
1, 2, 5, 8, 13, 18, 25, 32 (**Feb Monthly**)

- 6) WAP to print following Pattern

```
ABCDEF GFEDCBA
ABCDEF FEDCBA
ABCDE EDCBA
ABCD DCBA
ABC CBA
```

AB BA
A A

- 7) Write a Java program that prints the numbers from 1 to 50. But for multiples of three print "Fast" instead of the number and for the multiples of five print "Slow". For numbers which are multiples of both three and five print "Fast&Slow". After program looping is completed print how many times "Fast" was printed, "slow" was printed and "Fast&Slow" was printed. **(Feb Monthly)**

H/W Assignment:

Tip: Trainors will not assist for these questions.

- 1) Display numbers from 1 to 100 using different loops.
- 2) Write a program to create table of a number.
- 3) Find if given number is prime or not.
- 4) Calculate factorial of a number.
- 5) Count number of digits of any number.
- 6) Find sum of all digits of a number.
- 7) Find out given number is palindrome or not.
- 8) Display all elements between 400 to 500 (both numbers excluded) ending with seven. {Hint: output- 407,417,427....}
- 9) Write a menu driven program to find all prime, even and odd numbers between 1 to 100 using switch case
- 10) WAP to print following Pattern

```
*  
* *  
* * *  
* * * *  
* * * * *
```

- 11) Write a java program to generate a following *'s triangle.

```
1  
1 2  
1 2 3  
1 2 3 4  
1 2 3 4 5
```

- 12) Write a program in Java to make such a pattern like a pyramid with a number which will repeat the number in the same row.

```
1  
2 2  
3 3 3  
4 4 4 4  
5 5 5 5 5
```

13) WAP to print following Pattern

```
A
A B
A B C
A B C D
A B C D E
A B C D E F
```

14) WAP to print following Pattern

```
1
2 1
3 2 1
4 3 2 1
5 4 3 2 1
```

15) WAP to print following Pattern

```
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
```

16) WAP to print following Pattern

```
1 0 1 0 1
0 1 0 1 0
1 0 1 0 1
0 1 0 1 0
1 0 1 0 1
```

17) Write a program in Java to print the Floyd's Triangle.

```
1
01
101
0101
10101
```

18) Write a java program to generate following pattern.

```
*
***
*****
```



```

*****
*****
*****
*****
***
*

```

User should input number of rows to print. It should be odd number. If even number is entered ask user to input odd number.(Feb Monthly)

19) WAP to print following Pattern

```

1
1 0
1 0 1
1 0 1 0
1 0 1 0 1

```

20) Write a Java program to display the number rhombus structure.

```

1
2 1 2
3 2 1 2 3
4 3 2 1 2 3 4

```

21) Accept number and check whether it is kaprekar or not Consider an n-digit number k. Square it and add the right n digits to the left n or n-1 digits. If the resultant sum is k, then k is called a Kaprekar number. For example, 9 is a Kaprekar number since $9^2 = 81$ and $8 + 1 = 9$ and 297 is a Kaprekar number since $297^2 = 88209$ and $88 + 209 = 297$. [5M](April Monthly).

22) WAP to print the following pattern: (Feb Monthly).

```

1
1 1
1 2 1
1 3 3 1
1 4 6 4 1

```

Good To Have:

1. Write a Java program to print the first 15 numbers of the Pell series. In mathematics, the Pell numbers are an infinite sequence of integers. The sequence of Pell numbers starts with 0 and 1, and then each Pell number is the sum of twice the previous Pell number and the Pell number before that. The first few terms of the sequence are : 0, 1, 2, 5, 12, 29, 70, 169, 408, 985, 2378, 5741, 13860,... (Feb Monthly).
- 2) Write a java program to generate following series. Print n terms. Accept n from user. Series is : 1 -3 -5 -7 9 -11 -13 15 -17 -19 21
After generating the series show total count of negative numbers on console.
Hint negative sign is for prime numbers only. (Feb Monthly).

- 3) Write a java program or function to check Harshad number (or Niven number). Your program should take one positive integer from the user as input and check whether this integer is Harshad number (Niven number) or not. Harshad number or Niven number is a number which is divisible by the sum of its digits. For example, 21 is a Harshad number because it is divisible by the sum of its digits. $2+1 = 3$ and 21 is divisible by 3 $\rightarrow 21/3 = 7$. **(Jun Monthly)**.
- 4) Write a Java program to display the pattern.
- ```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4 5 6
1 2 3 4 5 6 7
1 2 3 4 5 6
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
```
- 5) Generate Fibonacci series for first n terms.

**Interview Questions :**

- 1) What are different types of loops?
- 2) What is the difference between while and do-while loop?
- 3) What are the looping constructs in Java?
- 4) Can you create a 'for' loop without any conditional expressions?

# **Topic: OOPs**

## ***Sub-Topics:***

- 1) *Introduction*
- 2) *Classes & Objects*
- 3) *Creation of class and conventions for creating a class*
- 4) *Elements of classes*
- 5) *Local, instance and static variables*
- 6) *Creating objects*
- 7) *Introduction to OOPs methodologies*
- 8) *Wrapper classes*

## ***C/W Assignment:***

Tip: Trainors will assist for these questions.

- 1) WAP to display how to create a class and objects and the different methods and other elements of a class using objects.
- 2) Create a class Addition having three instance variables, three methods. Calculate and display addition for two objects.
- 3) A company decided to give bonus of 5% to employees if his/her year of service is more than 5 years. Ask user for their salary and year of service and print the net bonus amount.
- 4) Scenario: A kids laptop manufacturer needs to develop a laptop which will display a message first,
  - a. Enter Option:
  - b. Add-1
  - c. Subtract-2
  - d. Multiply-3
  - e. Quit-4
  - f. The kid should be allowed to enter an option. If the kid enters 1, a message needs to be displayed,
  - g. "Enter two numbers to be added".
  - h. The kid should be allowed to enter two numbers (In two separate lines).
  - i. Based on the numbers entered, the program should add and display the result as below
  - j. "The result is <result>"
  - k. After the result is displayed, the program should loop back and ask for the next menu entry. If the kid enters 4, the program should quit. (The program needs to be executed and do either one of the options until the kid enters the option 4)

## ***H/W Assignment:***

Tip: Trainors will not assist for these questions.

- 1) Write a program to enter temperature in °Celsius and convert it into °Fahrenheit and vice versa.
- 2) Problem Statement:

Create a class ScannerLaptopDemo.java with a main method which performs the required operations as specified above.

Algorithm:

1. Display message as specified to print the various options 1 to 4.  
2. On user enters one of the option based on the input the appropriate arithmetic operation is done as mentioned below.

3. Using switch statement, display the appropriate message.

Example if option is 1 “Enter two numbers to be added”, get the two numbers using Scanner class and print the added result.

If option is 2 “Enter two numbers to be subtracted”, get the two numbers using Scanner class and print the subtracted result.

If option is 3 “Enter two numbers to be multiplied”, get the two numbers using Scanner class and print the multiplied result.

If option is 4 terminate the program.

Perform step 1 to 3 in a do-while loop (while option != 4).

#### ***Interview Questions:***

- 1) What is the difference between object oriented programming language and object based programming language?
- 2) What are the OOPS methodologies?
- 3) What is a Class?
- 4) What is an Object?
- 5) What is Scanner class?
- 6) What is new?
- 7) What is the purpose of wrapper classes?
- 8) What are the wrapper classes available in Java?
- 9) What do you mean by autoboxing and auto-unboxing?

# Topic: Access Modifiers

## *Subtopics:*

- 1) Public
- 2) private
- 3) protected
- 4) default

## *C/w Assignments:*

Create a class Student with 4 variables called as rollNo, admissionNo, age, courseId. Each of the variables should have one of the access modifiers: public, protected, no-access-modifier and private. Add 4 methods in the class: public method doPublic, <default> modifier method doDefault, protected method doProtected, private method doPrivate.

1. In main method outside the class but in same package – create object of type Student.
  - a. Try to access all the variables and all the methods in it. Verify the visibility against the access modifier table.
2. In same package as Student, create a new class TQPPStudent which extends the Student class.
  - a. Create a method called as checkStudentVariableAccess in TQPPStudent and try to assign values to the inherited variables in the method. Verify the visibility against the access modifier table.
  - b. Create a method called as checkStudentMethodAccess in TQPPStudent and try to call the inherited methods in the method. Verify the visibility against the access modifier table.
3. In another package, create a new class OtherPackageStudent which extends the Student class.
  - a. Create a method called as checkStudentVariableAccess in Student and try to assign values to the inherited variables in the method. Verify the visibility against the access modifier table.
  - b. Create a method called as checkStudentMethodAccess in Student and try to call the inherited methods in the method. Verify the visibility against the access modifier table.
4. In main method outside the class but in some other package – create object of type Student.
  - a. Try to access all the variables and all the methods in it. Verify the visibility against the access modifier table.
5. In main method outside the class but in some other package – create object of type HefshineStudent.
  - a. Try to access all the variables and all the methods in it. Verify the visibility against the access modifier table.
6. In main method outside the class but in some other package – create object of type OtherPackageStudent.
  - a. Try to access all the variables and all the methods in it. Verify the visibility against the access modifier table.

# Topic: Methods

## **Sub-Topics:**

- 1) *Introduction*
- 2) *Structure of method*
- 3) *Non-parameterised method with no return type*
- 4) *Non-parameterised method with return type*
- 5) *Parameterised method with no return type*
- 6) *Parameterised method with return type*
- 7) *Conventions in methods*
- 8) *Method overloading*
- 9) *Static methods*

## **C/W Assignment:**

Tip: Trainors will assist for these questions.

- 1) Write a program to show method overloading.
- 2) Write a program to copy values of one object into another by assigning the values of one object into another.
- 3) Create a class to print the area of a square and a rectangle. The class has two methods with the same name but different number of parameters. The method for printing area of rectangle has two parameters which are length and breadth respectively while the other method for printing area of square has one parameter which is side of square.
- 4) Create a class Account containing following methods :  
insert() to insert account data  
display() to display account information  
deposit() to deposit amount  
withdraw() to withdraw amount  
checkbalance() to check balance  
e.g.

```
class Account{
 int acc_no;
 String name;
 float amount;
}
```

## **H/W Assignment:**

Tip: Trainors will not assist for these questions.

- 1) Create Mydateclass which have 3 instance variables (day, month, and year). Create two methods setDate() and displayDate() for Mydate class where write a logic for assigning some values to instance variable in setDate and display logic in displayDate() method.

- 2) Create one class Box having instance variables (height, width, depth). Add one method to calculate volume with void return type.
- 3) Create Calculator class which have 4 methods add(int i, int j), sub(int i, int j), mul(int i, int j), div(int i, int j) with return type double.
- 4) Write a program to find the smallest number among three numbers using method
- 5) Create 4 overloaded methods for “test ()” and invoke all versions of the overloaded methods.
  - a. Create another class Overload.java which has a main method to call the overloaded methods in OverloadDemo.java
- 6) Create one class Box having instance variables (height, width, depth). Add one method to calculate volume with double return type.
- 7) Create a class to print an integer and a character with two methods having the same name but different sequence of the integer and the character parameters. For example, if the parameters of the first method are of the form (int n, char c), then that of the second method will be of the form (char c, int n).
- 8) Perform the following:

|                    |                                                                                             |
|--------------------|---------------------------------------------------------------------------------------------|
| Class Name         | Calculator                                                                                  |
| Method Name        | calculateSum                                                                                |
| Method Description | Calculates the sum of two numbers                                                           |
| Argument           | int number1, int number2                                                                    |
| Return Type        | int - Sum                                                                                   |
| Logic              | Calculate the sum of the two numbers number1 and number2 and return the sum.                |
| Method Name        | calculateDifference                                                                         |
| Method Description | Calculates the difference between two numbers                                               |
| Argument           | int number1 , int number2                                                                   |
| Return Type        | int - difference                                                                            |
| Logic              | Calculate the difference between the numbers number1 and number2 and return the difference. |

1)

- 9) Perform the following:

|                    |                                                     |
|--------------------|-----------------------------------------------------|
| Class Name         | MessagePrinter                                      |
| Method Name        | printMessage                                        |
| Method Description | Prints the message                                  |
| Argument           | String name                                         |
| Return Type        | Void                                                |
| Logic              | Print the message using the console output command. |

- 10) Create a Java class “Square.java” with a method “calculateArea”  
 This method should accept length as an argument, calculate the area and return the area.  
 The main method should invoke the Square objects “calculateArea” method by passing a value for the length, say 20.

The main method should also print the area (result).

***Interview Questions:***

- 1) What are methods?
- 2) What is meant by Method Overloading?
- 3) What are Method Overloading rules?
- 4) Is java Pass by Reference or Pass by Value?
- 5) What are the features we need to know while invoking overloaded



# Topic: Constructors

## **Sub-Topics:**

- 1) *Introduction*
- 2) *Structure and use of constructors*
- 3) *Difference between methods & constructors*
- 4) *Parameterised constructors*
- 5) *Non-Parameterised constructors*
- 6) *Constructor overloading*
- 7) *Use of this*
- 8) *Inner Classes*

## **C/W Assignment:**

Tip: Trainors will assist for these questions.

- 1) Create a class Cycle with member variables: int accountNo, int noOfWheels. Create a default constructor with a SOP in it "I am default constructor". Create another constructor which takes 2 arguments, calls the default constructor using this () and has a SOP in it "I am another constructor". In main method, create an object of type Cycle by using default constructor. Note the output. Create another object of type Cycle by using the parameterized constructor. Note the sequence of SOPs indicating that inner most constructors are called first.
- 2) Declaring and using constructors  
Create a class Circle.java in a package "com.HefShine.shapes", add a float instance variable radius and add a default constructor (Constructor 1) for the class. This constructor should initialize the radius to a default value 1.5f.  
The above constructor should be invoked from a main method from another class, Shape.java (in different package com.HefShine.geometry).
- 3) Create a class Test with 2 constructors and 3 methods with zero, one and two parameters respectively. Methods should perform following programs:
  - a. Even odd
  - b. Factorial of a number using do-while loop.
  - c. Find sum of all digits of a number
- 4) Write a demo program for different types of Inner Classes.

## **H/W Assignment:**

Tip: Trainors will not assist for these questions.

- 1) Create one class Dog, create only one default constructor. Create one object for Dog class in main () method.
- 2) Create one class Cat and create only one but parameterized constructor. And create one object like Cat c = new Cat (); It will give error since you have defined your own parameterized constructor, Java does not provide default one.
- 3) Create class student and write a meaningful program with 2 constructors (default and parameterized), 3 methods (1 with return type (calculating percentage), 1 without return type (displaying student data), 1 with parameter list).

- 4) Write a program to print the names of students by creating a Student class. If no name is passed while creating an object of Student class, then the name should be "Unknown", otherwise the name should be equal to the String value passed while creating object of Student class.
- 5) Create a class named 'Rectangle' with two data members- length and breadth and a method to calculate the area which is 'length\*breadth'. The class has three constructors which are:
  - 1 - having no parameter - values of both length and breadth are assigned zero.
  - 2 - Having two numbers as parameters - the two numbers are assigned as length and breadth respectively.
  - 3 - Having one number as parameter - both length and breadth are assigned that number.Now, create objects of the 'Rectangle' class having none, one and two parameters and print their areas.
- 6) Suppose you have a Piggy Bank with an initial amount of \$50 and you have to add some more amounts to it. Create a class 'AddAmount' with a data member named 'amount' with an initial value of \$50. Now make two constructors of this class as follows:
  - 1 - without any parameter - no amount will be added to the Piggie Bank
  - 2 - having a parameter which is the amount that will be added to Piggie BankCreate object of the 'AddAmount' class and display the final amount in Piggie Bank
- 7) Create a class named 'Programming'. While creating an object of the class, if nothing is passed to it, then the message "I love programming languages" should be printed. If some String is passed to it, then in place of "programming languages" the name of that String variable should be printed. For example, while creating object if we pass "Java", then "I love Java" should be printed.
- 8) Create a class Student with int id and String name as member variables. Generate getter and setter methods for id and name. Observe in the generated setter method how the parameter's value is assigned to the member variable using this keyword. What would happen in the setter method if you remove 'this.' when assigning the parameter value?
  - a. In main create 2 objects of Student. Set id of first student to 11 and 21 to the other student's id. Check and observe if you can use 'this' in main method.
- 9) Create a class Test, create 4 constructors explaining the concept of constructor overloading.
- 10) Overloading constructors and using "this" keyword.

In the Circle.java class created above add an instance float variable pi and create two overloaded constructors.

**Constructor 1-** with a float argument name radius. The constructor should initialize the class variable radius with the method argument radius.

**NOTE:** The instance variable and the method argument should be named same as "radius".

**Constructor 2-** with two float arguments radius and pi. Default the class pi value to 3.5 and set the instance variable with the radius method argument.

The constructor "constructor 2" should be invoked from a main method from class, Area.java (in a packagecom.HefShine.shapes).
- 10) Constructor Chaining

In Circle.java, invoke the Constructor 2 created in the previous step from Constructor 1.

11) Create two methods and calculate area and circumference of a Circle

In the Circle.java class, create two methods as listed below

- a. Method 1 - calculateCircleArea should accept the float radius as parameter and calculate the area ( $\pi * r * r$ ). It should return the result value to the main method where it should be printed in the console.
- b. Method 2 – calculateCircumference should accept float radius as parameter and calculate the circumference ( $2 * \pi * r$ ). It should return the result value to the main method where it should be printed in the console.

Call these two methods from the main method in Circle.java by passing appropriate parameters.

12) WAP to demonstrate different types of Inner Classes .Create class College as a Outer Class and Department as a Inner Class.Use appropriate Variables and Methods.

**Interview Questions:**

- 1) What are constructors?
- 2) What is the difference between methods and constructors?
- 3) Explain default constructors.
- 4) Can a class have multiple constructors?
- 5) What are the rules to define a constructor?
- 6) Explain the use of 'this' keyword.
- 7) What is an Inner class?
- 8) What are the types of classes available in Java?
- 9) Write two lines of code which will instantiate the inner class from the outer class.
- 10) Write two lines of code and explain how will you implement an anonymous inner class.
- 11) Explain about Static Nested classes.
- 12) What are the valid modifiers of an inner class?
- 13) How will you define a constructor? Give an example.
- 14) What is the benefit of reference variable? Give an example.
- 15) What are the possible access modifiers for a constructor?

# Topic: Inheritance

## **Sub-Topics:**

- 1) *Introduction*
- 2) *Single Inheritance*
- 3) *Multilevel Inheritance*
- 4) *Multiple Inheritance*
- 5) *Hierarchical Inheritance*
- 6) *Hybrid Inheritance*
- 7) *Use of super, final*

## **C/W Assignment:**

Tip: Trainors will assist for these questions.

- 1) Write 3 Meaningful programs for single inheritance.
- 2) Create a class named 'Member' having the following members:  
Data members  
1 - Name  
2 - Age  
3 - Phone number  
4 - Address  
5 - Salary  
It also has a method named 'printSalary' which prints the salary of the members.  
Two classes 'Employee' and 'Manager' inherit the 'Member' class. The 'Employee' and 'Manager' classes have data members 'specialization' and 'department' respectively. Now, assign name, age, phone number, address and salary to an employee and a manager by making an object of both of these classes and print the same.
- 3) Create a class named 'Rectangle' with two data members 'length' and 'breadth' and two methods to print the area and perimeter of the rectangle respectively. Its constructor having parameters for length and breadth is used to initialize length and breadth of the rectangle. Let class 'Square' inherit the 'Rectangle' class with its constructor having a parameter for its side (suppose s) calling the constructor of its parent class as 'super(s,s)'. Print the area and perimeter of a rectangle and a square.
- 4) Show use of Multi Level Inheritance & Method Overriding in case of Hotel Mgmt System(**Feb Monthly**).

### ***H/W Assignment:***

Tip: Trainors will not assist for these questions.

1. Create a class with a method that prints "This is parent class" and its subclass with another method that prints "This is child class". Now, create an object for each of the class and call
  - 1 - method of parent class by object of parent class
  - 2 - method of child class by object of child class
  - 3 - method of parent class by object of child class
2. Create 3 class tests and define methods explaining the concept of multilevel inheritance.
3. WAP to have Department class created with id, name. Student class has roll, name and Department object should have id and name. Assign and print individual values in main method
4. Explain the concept of hierarchical inheritance by suitable example.
5. Create Class Laptop which has variables noOfUSBPort, processorSpeed of type int. Create getter, setter methods for the variables. In main method, create Laptop object set values of variables noOfUSBPort, processorSpeed using setter methods. Print variables noOfUSBPort, processorSpeed using getter methods.
6. Create class IPLTeam with method play (). Create child classes of IPLTeam called as CSK, RCB. In main, call play () from child class objects.
7. WAP to create a class Kid with method readBook() and another method readBook () with 2 parameters. The method readBook here is over-loaded (same method name but different parameters). Create a class BigKid which extends Kid created above. Implement readBook() differently in BigKid class. Here the method readBook() has been over-ridden in the child class BigKid()
8. WAP to Show 3 use of final - final class, final variable and final method
9. WAP to check whether you can inherit static variable of Parent class in Child class
10. **Scenario:** In a company there are employees with two designations Manager and Trainee. Both employees share the same set of attributes and basic salary calculation logic is same but the basic salary of trainee and manager are different.

The Manager has a travel allowance equal to 15% of the basic salary, whereas all the other employees the travel allowance is 10% of the basic salary. Write a program to maintain the entities using inheritance.
11. Problem Statement on above scenario
  - a. Create a class Employee with the following instance variables.

| Instance variables | Data type                     |
|--------------------|-------------------------------|
| employeeId         | Long                          |
| employeeName       | String                        |
| employee Address   | String                        |
| employee Phone     | Long                          |
| basicSalary        | Double                        |
| specialAllowance   | double default value- 250.80  |
| Hra                | double,default value- 1000.50 |

- b. Create an overloaded constructor in the employee class, which takes the below constructor parameters and initializes them to their respective instance variables.

| Constructor parameter | Instance Variable |
|-----------------------|-------------------|
| Id                    | employeeId        |
| Name                  | employeeName      |
| Address               | employeeAddress   |
| Phone                 | employeePhone     |

- c. Create a method calculateSalary in which the basic salary needs to be calculated as below.  
 $\text{salary} = \text{basicSalary} + (\text{basicSalary} * \text{specialAllowance} / 100) + (\text{basicSalary} * \text{hra} / 100);$   
 The calculated salary should be displayed in the console.  
 NOTE: salary is a local variable.
- d. Create the sub classes Manager and Trainee with base class Employee. Create overloaded constructors which takes the below parameters and initializes them to their respective variables in the super class

| Constructor parameter | Instance Variable |
|-----------------------|-------------------|
| Id                    | employeeId        |
| Name                  | employeeName      |
| Address               | employeeAddress   |
| Phone                 | employeePhone     |
| Salary                | basicSalary       |

- e. Create a class "InheritanceActivity.java" with a main method which performs the below functions

## 12. Problem Statement on above scenario

- a. Add a method called calculateTransportAllowance in Employee class which should calculate the transport allowance by calculating 10% (default allowance) of the salary. Print the salary after calculating.  
 $\text{transportAllowance} = 10/100 * \text{basicSalary}.$
- b. For a manager, the transportation allowance is 15% of the basic salary. So override the calculateTransportAllowance method in Manager class which should calculate the transport allowance as 15% of the base salary. Print the salary after calculating.  
 $\text{transportAllowance} = 15 * \text{basicSalary} / 100.$
- c. For a trainee, the transport allowance is same as the default allowance; the method calculateTransportAllowance in the base class can be used.
- d. Invoke the calculateTransportAllowance for the manager and trainee class in the main method of InheritanceActivity.java.

### Interview Questions:

- 1) Explain difference between single and multilevel inheritance.
- 2) Why java does not support multiple inheritance?

- 3) What is final?
- 4) What is hierarchal inheritance?
- 5) What do you understand by the keyword super?
- 6) What are the benefits of inheritance?
- 7) Write a code that uses inheritance concept.
- 8) What are the types of inheritance relationships
- 9) Is it possible to extend more than one class? State the reasons.
- 10) How does a subclass call a constructor defined in its superclass?
- 11) In what order are constructors called in a class hierarchy?.

# **Topic: Containment (Object under object)**

## ***Subtopics:***

1. Structure & Need
2. Difference between containment and inheritance
3. Containment using constructor
4. Containment using Getter & Setter

## **C/W Assignment:**

Tip: Trainers will assist for these questions.

1. WAP to have Department class created with id, name. Student class has roll, name and Department object should have id and name. Assign and print individual values in main method
2. WAP to use containment for following hierarchy. Fill all data and print all data: (id is a integer number, name is a String)

Class Institute:

id      name

Branch:

Id      name

Subject:

Id      name

Topic:

Id      name

SubTopic:

Id      name

Question:

Id      name

## **H/W Assignment:**

Tip: Trainers will not assist for these questions.

1. Create class Person which has attributes (name, gender, age, Address). Address is class which has attributes (city, state, country). Display Persons data  
Note. Containment using constructor and getter/setter
2. Create Employee class which has attributes (id, name, salary, dept, mydate). Where dept and mydate is class, dept has (dept\_id, dept\_name) And mydate has (day, month, year).  
Display Employee information.  
Note. Containment using constructor and getter/setter



## **Topic: Polymorphism& Encapsulation**

### **Sub-Topics:**

1. *Introduction*
2. *Compile time polymorphism*
3. *Runtime polymorphism*
4. *Encapsulation*

### **C/W Assignment:**

Tip:Trainers will assist for these questions.

1. Explain the concept of method overloading with an example.
2. Explain the concept of constructor overloading with suitable example.
3. Create one class Employee (emp\_id, name, sal) with private access specifier and create getter and setter. Print the data by creating objects of the class.
4. Create a class named 'Shape' with a method to print "This is shape class". Then create another class named 'Rectangle' inheriting the Shape class, having a method to print "This is rectangular shape" .Create another class 'circle' inheriting 'rectangle' class and create a method to print "This is circular shape" respectively. Now call the method of 'Shape' and 'Rectangle' class by the object of 'circle' class.

### **H/W Assignment:**

Tip:Trainers will not assist for these questions.

1. Create a class Cycle with member variables: intaccountNo, intnoOfWheels. Create a default constructor with a SOP in it "I am default constructor" "Create another constructor which takes 2 arguments, calls the default constructor using this () and has a SOP in it "I am another constructor". In main method, create an object of type Cycle by using default constructor. Note the output. Create another object of type Cycle by using the parameterized constructor. Note the sequence of SOPs indicating that inner most constructors are called first.
2. Write a program to check whether constructors can override.
3. Create Academy class with appropriate attributes (instance variable) and create getter & setter methods.
4. In our FaceBook profile, we would like to hide the age, marital status and phone number information to the external world. This can be done using encapsulation where the fields are made private and can be accessed only by the accessor and mutator methods.

### **Interview Questions:**

1. What do you understand by polymorphism
2. What is method overloading?
3. Explain method overriding.
4. Can constructors be overridden?
5. Can constructors be overloaded?
6. How will you differentiate "overloading" and "overriding"?
7. Write a Java code that does method overriding.

8. Write a Java code that does method overloading?
9. Give examples for illegal method overrides.
10. Write a code snippet that uses encapsulation concept.

HetShine Software

# **Topic: Abstraction& Interfaces**

## ***Sub-Topics:***

- 1) *Introduction*
- 2) *Abstract methods*
- 3) *Structure & need of abstract classes*
- 4) *Abstract classes*
- 5) *Structure & need of interfaces*
- 6) *Difference between abstract classes & interfaces*
- 7) *Extending Interfaces*
- 8) *Interface Java 8 features*
- 9) *Packages*

## ***C/W Assignment:***

Tip: Trainors will assist for these questions.

- 1) Explain the concept of abstraction with a suitable example.
- 2) Create an abstract class Machine with an implemented / concrete method rotate and an abstract method crush. Create a class Juicer which extends this abstract class Machine and implements method crush. Now, add another method filter in the class Juicer. In main method,
  - a. Create an object of type Juicer and calls its crush, rotate and filter methods.
  - b. Create an object of type Juicer with reference variable of Machine (Machine m = new Juicer). Check the methods available to m.
- 3) Above we created an abstract class Machine with an abstract method crush and a concrete / implemented method rotate. Now create another abstract class Mixer which extends the abstract class Machine, has method crush implemented and has an additional concrete method blend. In main create an object of class of Mixer. Call methods rotate, blend and crush.
- 4) Explain the concept of interfaces with an example.
- 5) WAP to create and use classes from different Packages.

## ***H/W Assignment:***

Tip: Trainors will not assist for these questions.

1. Create 2 abstract classes Abs1 and Abs2 each with different implemented methods doAbs1() and doAbs2() respectively. Check if you can create a class Temp which extends both the abstract classes. (Note- A class cannot extend 2 classes simultaneously)
2. Check following variations:
  - a. Can we create an object of an abstract class?
  - b. Can we declare a class abstract with no methods in it?
  - c. Can we declare a class abstract even if it does not have any abstract method?
  - d. Can we declare an abstract class which has both abstract as well as implemented methods?
  - e. Can a class extend 2 or more abstract classes?
  - f. Can an abstract class extend 2 or more abstract classes?

- g. Can an interface extend 1 or many abstract classes?
  - h. Can an abstract class implements 1 or many interface?
3. WAP to create an abstract class Parent. Add an abstract method cook() in it which has only definition and one method wash() which has implementation. Create class Child which extends Parent and add the missing method implementation. In main, use both the methods by creating instance of the concrete class.
  4. Show use of parameterized constructor of parent class (Child class invokes parent's parameterized constructor) using super
  5. Create interface Cake with a method declared as bake. Create 2 classes Strawberry, BlackForest both implementing Cake interface Create interface IceCream with method eat and Juice with method drink. Create class Mango which implements both interfaces.
  6. A library needs to develop an online application for two types of users/roles, Adults and children. Both of these users should be able to register an account. Any user who is less than 12 years of age will be registered as a child and they can borrow a "Kids" category book for 10 days, whereas an adult can borrow "Fiction" category books which need to be returned within 7 days.

Note: In future, more users/roles might be added to the library where similar rules will be enforced.

Develop Interfaces and classes for the categories mentioned above.

- a. Create an interface LibraryUser with the following methods declared

| Method Name     |
|-----------------|
| registerAccount |
| requestBook     |

- b. Create 2 classes "KidUsers" and "AdultUser" which implements the LibraryUser interface.
- c. Both the classes should have two instance variables as specified below.

| Instance variables | Data type |
|--------------------|-----------|
| Age                | Int       |
| bookType           | String    |

- d. The methods in the KidUser class should perform the following logic.
  - i. registerAccount : if age < 12, a message displaying "You have successfully registered under a Kids Account" should be displayed in the console.  
If (age > 12), a message displaying, "Sorry, Age must be less than 12 to register as a kid" should be displayed in the console.
  - ii. requestBook: if bookType is "Kids", a message displaying "Book Issued successfully, please return the book within 10 days" should be displayed in the console. else, a message displaying, "Oops, you are allowed to take only kids books" should be displayed in the console.
- e. The methods in the AdultUser class should perform the following logic.
  - i. registerAccount : if age > 12, a message displaying "You have successfully registered under an Adult Account" should be displayed in the console.

If age<12, a message displaying, “Sorry, Age must be greater than 12 to register as an adult” should be displayed in the console.

- ii. requestBook: if bookType is “Fiction”, a message displaying “Book Issued successfully, please return the book within 7 days” should be displayed in the console.  
else, a message displaying, “Oops, you are allowed to take only adult Fiction books” should be displayed in the console.
  - f. Create a class “LibraryInterfaceDemo.java” with a main method which performs the below functions
- 7.WAP to create Package arithmetic having classes Addition, Subtraction, Division ,Multiplication with appropriate methods .Use this Classes outside the package.

***Interview Questions:***

1. What do understand by abstraction?
2. What are abstract classes and abstract methods?
3. Explain interfaces?
4. How multiple inheritance is achieved using interfaces?
5. What is the difference between abstract class and an interface?
6. Does an abstract class contain non-abstract methods?
7. Can we use default when we override methods of an interface?
8. Can an abstract method have a body?
9. Can you create an object of an abstract class using new?

# Topic: Arrays

## **Sub-Topics:**

1. *Introduction*
2. *Declaration, initialization of array*
3. *Single dimensional array*
4. *for each*
5. *Operations on array: searching, sorting*
6. *Two-dimensional array*
7. *Array of objects*

## **C/W Assignment:**

Tip: Trainers will assist for these questions.

1. Write a program to enter elements and perform binary search.
2. WAP sort array elements in ascending order using selection sort.
3. Create class Dept(did, dname), class MyDate(day, month, year)  
Class Employee(emp\_id, emp\_name, salary, MyDate (object), dept(object)). Create array of Employee and display the array elements.
4. WAP to print minimum in columns. Means e.g. arr[][] = {{22, 31, 9}, {12, 5, 16}, {34, 42, 2}} output is: 12, 5, 2.
5. Write a Java program to arrange the elements of an given array of integers where all positive integers appear before all the negative integers.
6. WAP to create transpose of a matrix (transpose is converting rows to columns) and print it.
7. Number of unique pairs in an array. Given an array of N elements, the task is to find all the unique pairs that can be formed using the elements of a given array. **(March Monthly).**

Examples: Input: arr[] = {1, 1, 2}

Output: 4

(1, 1), (1, 2), (2, 1), (2, 2) are the only possible pairs.

Input: arr[] = {1, 2, 3}

Output: 9

8. WAP to print outer elements of 2D array of 4X4..
9. WAP to find the average of the inner most elements of an array
10. reate or implement stack in java using array as underlying datastructure. (Jun Monhly)  
Push Pop peek mandatory methods for min. Create stack class having following methods:  
Push method: Push method will be used to insert new element tostack.  
Pop method: Pop method will remove top element of stack.  
Size method: Size method will return current size of stack.  
isEmpty method: isEmpty method will check, whether stackcontains any element.

isFull method: isFull method will check, whether stack has exhausted its capacity.

Peek method: Peek method will return top element of stack but will not remove it.

### ***H/W Assignment:***

Tip: Trainers will not assist for these questions.

1. WAP to create 1D array and accept data in that array. Calculate the average value of array elements.
2. WAP to add elements to single dimensional array and print elements from 1D array
3. WAP to insert an element in a specific position into an array.
4. WAP to merge 2 arrays to 3rd array.
5. WAP to print reverse of an array.
6. WAP sort array elements in ascending order using bubble sort.
7. WAP to print all negative elements in an array and also count total number of negative elements in an array.
8. Write a Java program to find sum of elements of an array.
9. WAP to put even and odd elements of array in two separate arrays.
10. WAP to find the maximum and minimum value in an array.
11. WAP to find the second smallest element in an array.
12. Write a Java program to test the equality of two arrays.
13. WAP to print the details of employees from Employee[] array who has same salary (Create Employee class which has 3 attributes id, name, salary and add employee objects to your array)
14. WAP to reverse the array itself. {Hint- arr[] = [3, 90, 45, 29, 37, 78] so your same array must be [78, 37, 29, 45, 90, 3]} without using temporary array.
15. WAP to replace all the 0's with 1's in your array. Your array is [26, 0, 67, 45, 0, 78, 54, 34, 10, 0, 34].
16. Write a Java program to get the difference between the largest and smallest values in an array of integers. The length of the array must be 1 and above
17. Write a Java program to separate even and odd numbers of an given array of integers. Put all even numbers first, and then odd numbers.
18. WAP to accept data in 2D array and print the data.
19. WAP to subtract two matrices.
20. WAP to sum two matrices.
21. WAP to print maximum in row wise in 2D array. Means e.g. arr[][] = {{22, 31, 9}, {12, 25, 16}} output is: 31 and 25.
22. Create Student class having rollno, name, marks. Create 10 objects. Using Array of Objects display information of student who got highest marks.
23. Create Class Employee (id, name, Salary). Create 5 Employee Objects. Display Employee information in descending order of Salary using Array Of Objects
24. Given an integer array and size of subarray, find the first subarray with least average in single loop. Print first index of subarray and average. (Mindstix)  
Method signature  
FindFirstsub(int arr[], int arr\_len, int sub\_arr\_len)

```
{
//Your code
}
```

Example:

Input:

int arr={3,7,90,20,5,50,40}, k=3

FindFirstsub(arr,7,3)

Output: Index:3 Avg:25

25. Given 2 character arrays s1 and s2 and another empty character array s3. Populate s3 by interleaving characters from both s1 and s1(**Mindstix**)

Method signature

Void interleaved (char[] s1, char[]s2, char[]s1, int s1\_len, int s2\_len)

```
{
// Your Code
}
```

Example:

S1={'a','b','c','d'};

S2={'w','x','y','z'};

Output:

S1={'a','w','b','x','c','y','d','z'}.

26. Write a java program or function to find saddle point of a matrix. Your program should take input matrix from the user, display the matrix and find the saddle point of that matrix. Saddle point of a matrix is an element in the matrix which is smallest in its row and largest in its column. A matrix can have many or no saddle points. For example,

```
6 3 1
9 7 8
2 4 5
```

In this matrix, 7 is the saddle point. Because it is the smallest in its row (2<sup>nd</sup> row) and largest in its column (2<sup>nd</sup> column). **.(April Monthly).**

27. Given an array of N distinct interger and a sum value S.WAP to find out count of triplets with sum smaller than given sum value.

Examples:

Array=[5,1,3,4,7]

S=12.

Output :4

Explanation: Below are triplets with sum less than 12

(1,3,4),(1,3,5),(1,3,7),(1,4,5)

28. WAP to find sum of main diagonal elements of a matrix.  
29. WAP to left rotate an array.  
30. How to pass array as a parameter to method in java?



**Good To Have:**

1. Write a Java program to find all the unique triplets such that sum of all the three elements is equal to a specified number. **(April Monthly).**  
Input-2.  
Output- [[1,5,-4],[-2,5,-1]]  
Reason:  $1+5-4=2$  &  $-2+5-1=2$  2 is Target ...
1. To Find unique Pair Of Integers in Array whose Sum is Given Number **[4M]**  
Given array : [2, 4, 3, 5, 6, -2, 4, 7, 8, 9]  
Given sum : 7  
Integer numbers, whose sum is equal to value : (2, 5) (4, 3) (-2, 9) . **(April Monthly).**
  1. Take 10 integer inputs from user and print the following:  
number of positive numbers  
number of negative numbers  
number of odd numbers  
number of even numbers
  2. WAP to print upper triangular matrix.

**Interview Questions:**

1. What are arrays?
2. What is one dimensional array?
3. How to create and access elements in java?
4. What are two dimensional arrays?
5. What are applications of an array?
6. What are advantages & disadvantages of arrays?

# Topic: Strings

## **Sub-Topics:**

- 1) *Introduction*
- 2) *Immutable Strings*
- 3) *Methods of strings*
- 4) *String buffer class*
- 5) *Methods of String Buffer*
- 6) *String builder*
- 7) *Methods of String Builder*
- 8) *String vs String buffer vs String Builder*

## **C/W Assignment:**

Tip: Trainers will assist for these questions.

- 1) Compare string using new operator when.
- 2) Write a Java program to find length of a string.
- 3) Write a Java program to concatenate two strings.
- 4) Write a Java program to compare two strings.
- 5) Write a Java program to convert lowercase string to uppercase.
- 6) Write a Java program to copy one string to another string.
- 7) WAP to split string into 2 tokens where string is "HELLO\$WORLD"
- 8) Write a Java program to find first occurrence of a character in a given string.
- 9) Write a Java program to trim trailing white space characters in a string
- 10) WAP to find longest word in the given sentence
- 11) How do you swap two string variables without using third or temp variable in java
- 12) Write a Java program to remove all extra blank spaces from a given string.
- 13) Write a Java program to toggle case of each character of a string.
- 14) Write a program which creates a StringBuffer "This is StringBuffer" and performs the following.
  - i. Adds the string "- This is a sample program" to existing string and display it.
  - ii. Inserts the string "Object" into the existing string at 21<sup>st</sup> position and display it.
  - iii. Reverses the entire string and displays it.
  - iv. Replaces the word "Buffer" with "Builder" and display it.
- 15) Exchange Cipher (String & char) This simple cipher exchanges 'A' and 'Z', 'B' and 'Y', 'C' and 'X', and so on. Write a program called ExchangeCipher that prompts user for a plaintext string consisting of mix-case letters only. Your program shall compute the ciphertext; and print the ciphertext in uppercase. For examples, Enter a plaintext string: abcXYZ The ciphertext string is: ZYXCBA (**March Monthly**)

## **H/W Assignment:**

Tip: Trainers will assist for these questions.

- 1) Create String with new operator and without new.
- 2) Compare string using ==.
- 3) Write a Java program to count occurrences of a character in given string.
- 4) Write a Java program to trim leading white space characters in a string.
- 5) Write a Java program to count total number of words in a string.
- 6) Write a Java program to find first occurrence of a word in a given string.
- 7) Write a Java program to search all occurrences of a character in given string.
- 8) Write a Java program to remove all occurrences of a character from string.

- 9) Returns the lowercase of the string and display it.
- 10) Write a Java program to remove all repeated characters from a given string.
- 11) Accept email\_id from user and check valid or not(should contain @,.)
- 12) Accept sentence replace each vowel by next consecutive character.
- 13) Write a Java program to count frequency of each character in a string
- 14) Write a Java program to find lowest frequency character in a string.
- 15) Write a Java program to toggle case of each character of a string
- 16) Write a Java program to repeat every character twice in the original string.
- 17) Write a Java program to get the character at the given index within the String
- 18) Write a Java program to trim both leading and trailing white space characters in a string.
- 19) Add 10 StringBuffer objects in an Array , Count no. Of Palindrome Strings, Display in Ascending order such Strings

20) Problem Statement 1:

Write a program which creates a String “Welcome to Java World” and performs the following

Returns the character at 5<sup>th</sup> position and display it.

Compares the above String with “Welcome” lexicographically ignoring case differences and display the result.

Concatenates “- Let us learn” to the above string and display it.

Returns the position of the first occurrence of character ‘a’ and display it.

Replaces all the occurrences of ‘a’ character with the new ‘e’ and display it.

Returns string between 4<sup>th</sup> position and 10<sup>th</sup> position and display it.

21) Write a Java program to test methods of String Buffer.

22) Write a Java program to test methods of String Builder.

23) Write a java program to find the percentage of uppercase letters, lowercase letters, digits and other special characters(including space) in the given string. For example, In “Tiger Runs @ The Speed Of 100 km/hour.”, (**March Monthly**)

Number of uppercase letters is 5. So percentage is 13.16%

Number of lowercase letters is 20. So percentage is 52.63%

Number of digits is 3. So percentage is 7.89%

Number of other characters is 10, So percentage is 26.32%

24) Write a program to sort numbers from String. Also output should be in String. E.g. if number in String is “2713” output should be “1237”. [3M](**April Monthly**)

25) Consider there is already a Set<String> created with 50 elements. Write a function which will take input as a String. If input string is present in Set then return the String which is present 2 locations before the matching String. Eg "AA", "BB", "CC", "DD" ..present. I/p "CC" then return "AA". If "BB" is passed then return null as 2 level before there is no string [4M] (**April Monthly**)

### **Interview Questions:**

- 1) What are Strings in Java? Is string a data type?
- 2) What are different ways to create string objects?
- 3) What is String pool?

- 4) What do you mean by mutable and immutable strings?
- 5) Why String Buffer is called mutable?
- 6) Can you create mutable string objects?
- 7) What is the difference between String Buffer and String Builder class?
- 8) Why String Buffer and String Builder classes are introduced in java when there already exist string class to represent the set of characters?
- 9) What are the various ways of assigning a string literal to a String variable?
- 10) Which method is used to append a string literal to a String variable?
- 11) What are the most widely used methods of String class?
- 12) When will you use String class and when will you use StringBuffer?
- 13) How will you add string to a StringBuffer? Give an example.
- 14) Which class is preferred :StringBuffer or StringBuilder? Why?
- 15) Is it possible to invoke chained methods in Java? If so, how will you invoke?
- 16) Can you point out the main difference between C/C++ strings and Java strings?
- 17) Can the contents of strings in Java be changed?

# **Topic: Exception Handling**

## ***Sub-Topics:***

- 1) *Introduction*
- 2) *Types of Exceptions*
- 3) *Using try-catch*
- 4) *Multiple catch clauses*
- 5) *Exception Hierarchy*
- 6) *Runtime stack Mechanism*
- 7) *Default Exception handler*
- 8) *Types of exceptions*
- 9) *Checked exceptions*
- 10) *Partially checked exceptions*
- 11) *Fully checked exceptions*
- 12) *Unchecked exceptions*
- 13) *Creating own exceptions*
- 14) *Throws vs Throw*
- 15) *Nested try statements*
- 16) *Use of throw, throws & finally*

## ***C/W Assignment:***

Tip: Trainors will assist for these questions.

- 1) WAP to demonstrate runtime stack mechanism using Arithmetic Exception.
- 2) WAP to demonstrate runtime stack mechanism using Arithmetic Exception and handle it using try catch.
- 3) WAP to show unchecked exception and use multiple catch blocks.
- 4) WAP to check whether Unchecked Exception is propagated in calling stack
- 5) WAP to throw exception to check if finally should get executed
- 6) WAP to show checked exception and use multiple catch block with Exception handler.
- 7) WAP to show the use of finally block.
- 8) WAP to show the usage of throw and throws for checked exceptions.
- 9) WAP to show the usage of throw and throws for unchecked exception.
- 10) .WAP to show the usage of throw by creating a user defined exception and handle it using try catch.

## ***H/W Assignment***

Tip: Trainors will not assist for these questions.

- 1) WAP to show any Exception
- 2) WAP where only try and finally is used.
- 3) WAP to show any Exception and catch that Exception.
- 4) WAP to check can we have an empty catch block?
- 5) WAP to show the scenario in which ArrayIndexOutOfBoundsException is generated and handle this exception.
- 6) WAP to perform following steps and execute the program.
- 7) Step 1: Create a class, *Demo* with a method, *division* with two int parameters

- a. Dividend
  - b. Divisor
- 8) This method should divide the dividend by divisor and return the result.
- 9) This method should also throw an *ArithmeticException* to the calling method.
- 10) Step 2: Create a class, *ThrowsDemo* with a main method
- 11) The main method should invoke the division method in *Demo* class.
- 12) The main method should also *catch* the *ArithmeticException* thrown by the division method and print the Exception "Arithmetic Exception is Thrown"
- 13) The try/catch block should also have a finally block which prints a message "The result is" <Result>
- 14) Step 1: In the Demo class division method perform the following logic.
  - a. If Divisor is zero throw a *ArithmeticException* with message "Divisor cannot be zero"
  - b. This method should throw this *ArithmeticException*.
  - c. Step 2: The exception thrown needs to be handled in *ThrowsDemo*
  - d. The main method should *catch* the *ArithmeticException* thrown by the division method and print the Exception and print the message in the exception Object.
  - e. The try/catch block should also have a finally block which prints a message "The result is" <Result>
- 15) Scenario: A shopping portal provides users to register their profile. During registration the system needs to validate the user age above 18 and should be placed in India. If not the system should throw an appropriate error.
  - a. Create a user defined exception classes named "*InvalidCountryException*" & "*InvalidAgeException*"
  - b. Overload the respective constructors.
  - c. Create a main class "UserRegistration", add the following method,
    - i. *registerProfile* - The parameter are String userName , int age, String country. Add the following logic
  - d. if country is not equal to "India" throw a *InvalidCountryException* with error message "User Outside India cannot be registered"
  - e. If age < 18 throw a *InvalidAgeException* with error message "User is a Minor"
  - f. Invoke the method *registerProfile* from the main method with the data specified and see how the program behave s:
- 16) WAP to check what happen when Exception is thrown by main method
- 17) WAP to check whether checked Exception is propagated in calling stack
- 18) Create a menu driven program 1. Try Catch demo 2.Try MultiCatch 3.Try Finally 4.Try Catch Finally 5. Throw 6.Throws

### Interview Questions:

- 1) What is an Exception?
- 2) What is the difference between error and exception?

- 3) What is the use of the finally block? Is finally block in java guaranteed to be called? When finally block is not called?
- 4) What do you mean by Checked Exceptions?
- 5) Explain Runtime Exceptions?
- 6) Which are the two subclasses under Exception class?
- 7) When throws keyword is used?
- 8) What things should be kept in mind while creating your own exceptions in Java?
- 9) What are the exception handling keywords in java?
- 10) What is an Exception Hierarchy?
- 11) Which is the parent class of Exception?
- 12) In which package throwable interface is present?
- 13) How does Default Exception Handler works.
- 14) What are different types of exceptions?
- 15) What are unchecked exceptions?
- 16) What is the difference between checked and unchecked exceptions?
- 17) Explain fully checked exceptions?
- 18) What are partially unchecked exceptions?
- 19) What is the main use of keyword throw?
- 20) What is the use of the finally block?
- 21) How will you define Exception Handling?
- 22) What are the different types of Exceptions?
- 23) How will you handle Exception in Java?
- 24) Write a compilable code using "try", "catch", and "finally"?
- 25) What is the use of "finally" clause? Give an example.
- 26) What are the exception types that can be thrown using the "throw" keyword?
- 27) How will you write a compilable code block using "throw" keyword?
- 28) Give a one line definition of an exception. What do you mean by checked and unchecked exceptions?
- 29) Which class is at the top of the exception hierarchy?
- 30) Which are the two important subclasses of Throwable?
- 31) Can you name the all important subclass of Exception?
- 32) What does the throws clause do?
- 33) Can you create your own custom exceptions in Java?
- 34) What do you understand by chained exceptions?

# **Topic: Multithreading**

## ***Sub-Topics:***

- 1) *Introduction*
- 2) *Thread life cycle*
- 3) *Creation of threads using class*
- 4) *Creation of thread using interface*
- 5) *Creating multiple threads*
- 6) *Thread scheduler*
- 7) *Synchronization*
- 8) *Thread priorities*
- 9) *Garbage Collection*
- 10) *Daemon Thread*
- 11) *Methods of thread-Join(),Sleep(),yield()*

## ***C/W Assignment:***

Tip: Trainors will assist for these questions.

- 1) Change priorities of thread. Check what values are for MIN\_PRIORITY, NORM\_PRIORITY and MAX\_PRIORITY.
- 2) WAP to show use of sleep method.
- 3) WAP to set & get priorities of a thread.
- 4) WAP to show example of daemon thread.
- 5) WAP to show the implementation of a thread and pausing of a thread till completion of main thread using yield().
- 6) WAP to execute 2 threads main and child thread, displaying that main thread calls join() and wait() for the child thread to get executed first and then gets completed.
- 7) What is deadlock? Show one example of deadlock.
- 8) A consumer thread consumes chocolates from a basket, producer thread produces fixed number of chocolates at a time. Write a program in which consumer thread checks for sufficient chocolates in basket, it waits for producer to produce if sufficient chocolates are not available in the basket and then consumes given number of chocolates. Producer thread will notify consumer thread after it finishes producing chocolates.
- 9) Create a class Item which has sell and purchase method when one thread is updating the item then other thread should not execute on same item. [5M] (April Monthly).

## ***H/W Assignment:***

Tip: Trainors will not assist for these questions.

- 1) Create thread by extending Thread class.
- 2) Create thread by implementing runnable interface.



- 3) Create 3 threads. First thread displays days of a week. Second thread displays table of 5. Third thread displays Fibonacci series.
- 4) Thread t1 prints numbers 1 to 10. Thread t2 prints characters a to h. Ensure that always characters are printed first and then numbers using join method.
- 5) Show use of all 3 join methods of thread class.
- 6) Show example of synchronized method in which two threads are trying to update same thread.
- 7) Show use of synchronization if two threads are sharing same objects.
- 8) If thread t1 is accessing static synchronized method m1, can thread t2 access synchronized method m2 at same time.
- 9) Prove following statement in your code.
  - i. While a thread executing static synchronized method, the remaining threads are not allowed to execute any static synchronized method of that class simultaneously. But remaining threads are allowed to execute the following methods simultaneously 1. Normal static methods 2. Synchronized instance methods 3. Normal instance methods.
- 10) Create Circle class having setRadius(), area(). One thread is calling setRadius(), another thread is calling area(). Using wait() and notify() implements this program.

#### ***Interview Questions:***

- 1) What is multithreading?
- 2) Explain the life cycle of a thread.
- 3) What is the difference between multi-processing and multithreading?
- 4) What are the different ways to create a thread?
- 5) What are the advantages of multithreading?
- 6) What is thread scheduler?
- 7) What is synchronisation?
- 8) What do you understand by Garbage collection?
- 9) How will you define Garbage Collection?
- 10) When does the garbage collection get executed?
- 11) Is there any chance that the Java application can throw "out of
- 12) memory" error? Why?
- 13) Explain with a code sample when an object is ready for garbage collection.
- 14) What are the situations in which JVM triggers garbage collection?
- 15) What are the lines of code you need to write to programmatically
- 16) trigger garbage collection?
- 17) What are the concepts that come to your mind about finalize() method?
- 18) How will you define a Thread?
- 19) How will you create Threads in Java?

- 20) What are the methods of Thread class that are mainly used to manage threads?
- 21) How does a thread get executed in Java?
- 22) What are the thread states?
- 23) Write a compilable Java code that creates a child thread using "Thread" class.
- 24) What are the methods of Objects that are used while managing threads?
- 25) Is it possible to create more than one thread in a Java application? If so, how will the threads communicate with each other?
- 26) How will you define Synchronization?
- 27) How will you make a thread to pause for ten minutes?
- 28) How will you use the "synchronized" keyword? Give code examples.
- 29) What happens when a synchronized method is invoked?
- 30) How will you make the thread to wait and start its execution again so that certain process gets executed?
- 31) Write a Java code and implement "wait" and "notify" methods.
- 32) What are the methods that belong to "Runnable" interface?
- 33) What is the use of join () and yield () methods?

# **Topic: Input-Output**

## ***Sub-Topics:***

- 1) *Introduction*
- 2) *Input stream*
- 3) *Output stream*

## ***C/W Assignment:***

Tip: Trainors will assist for these questions.

- 1) Write a program to read contents from file and store reverse contents in another file.
- 2) Write a program to input data (numbers, characters), store in a.txt file, read it and separate the contents in two different files namely char.txt and number.txt.
- 3) Write a program copy file contents into another file.

## ***H/W Assignment:***

Tip: Trainors will not assist for these questions.

- 1) Write a program to accept input from console in write contents in file.
- 2) Write a program to read data from File.
- 3) Write a program to write data in file.
- 4) Write a program to read string names from file and sort and write those names in another file.
- 5) Write a program to accept input from user. Append the contents to the existing file.
- 6) Write a Java program to find the longest word in a text file. **(April Monthly)**

# Topic: Collections -list

## **Sub-Topics:**

- 1) *Introduction*
- 2) *Lists*
- 3) *ArrayList-Introduction*
- 4) *Constructors & methods of ArrayList*
- 5) *Linked List-Introduction*
- 6) *Constructors & methods of LinkedList*
- 7) *Difference between ArrayList & LinkedList*
- 8) *Vector –Introduction*
- 9) *Constructors & Methods of Vectors*
- 10) *Stack-Introduction*
- 11) *Constructors & Methods of Vectors*
- 12) *Cursors-Introduction*
- 13) *Enumeration*
- 14) *Constructors & Methods of Enumeration*
- 15) *Limitations of Enumerations*
- 16) *Iterator-Introduction*
- 17) *Constructors & Methods of Iterator*
- 18) *Limitations of Iterator*
- 19) *List Iterator*
- 20) *Constructors & Methods of ListIterator*
- 21) *Difference between Enum,Iterator and ListIterator*

## **C/W Assignment:**

Tip: Trainors will assist for these questions.

- 1) WAP to add an element at a particular index using Add().
- 2) WAP to empty ArrayList
- 3) WAP to search an element from ArrayList
- 4) WAP to search the specified collection in this collection
- 5) WAP to retrieve an element (at a specified index) from a given ArrayList
- 6) WAP to print all the elements of an ArrayList using the position of the elements
- 7) WAP to get an element of a particular Index.
- 8) WAP to set or replace an element using set().
- 9) WAP to use add operation of ArrayList
- 10) WAP to print all elements of ArrayList using iterator
- 11) WAP to iterate through all elements in an ArrayList using for loop
- 12) WAP to iterate a linked list in reverse order.
- 13) WAP to insert the specified element at the specified position in the linked list.
- 14) WAP to create Emp (id,name,sal) object and add 2 objects to ArrayList. Sysout and see both variable memory space is printed. This is because toString is not overridden but if you would have done this for Integer then beautiful output would have been printed.
  - a. Now override toString for earlier assignment and now sysout and see values are printed
  - b. WAP to print Emp whose salary is > 10000
  - c. WAP to print Emp who have name "Sachin"

d.WAP to print Emp who have highest number of salary

***H/W Assignment:***

Tip:Trainors will not assist for these questions.

- 1) WAP to Add elements in an ArrayList.
- 2) WAP to Remove an element in an ArrayList
- 3) WAP to check if the ArrayList is empty or not.
- 4) WAP to check if an element is present in ArrayList using Contains().
- 5) WAP to clear all objects from ArrayList.
- 6) WAP to insert an element into the ArrayList at the first position
- 7) WAP to add 1 to 50 numbers in ArrayList and print only even numbers (using iterator)
- 8) WAP to match two collections
- 9) WAP to check if collection is empty
- 10) WAP to convert collection into array
- 11) WAP of swap two elements in an ArrayList
- 12) WAP to append the specified element to the end of a linked list.
- 13) WAP to iterate through all elements in a linked list.
- 14) WAP to iterate through all elements in a linked list starting at the specified position.
- 15) WAP to replace the second element of an ArrayList with the specified element
- 16) WAP to use add all elements to ArrayList
- 17) WAP to sort a given ArrayList
- 18) WAP to copy one ArrayList into another
- 19) WAP to compare two ArrayLists print if equal?
- 20) WAP to empty an ArrayList
- 21) WAP to trim the capacity of an ArrayList the current list size
- 22) WAP to increase the size of an ArrayList
- 23) WAP to update specific array element by given element
- 24) WAP to remove the third element from an ArrayList
- 25) WAP to insert elements into the linked list at the first and last position.
- 26) WAP to insert the specified element at the front of a linked list.
- 27) WAP to insert the specified element at the end of a linked list.
- 28) WAP to insert some elements at the specified position into a linked list.
- 29) WAP to get the first and last occurrence of the specified elements in a linked list.
- 30) WAP to display the elements and their positions in a linked list.
- 31) WAP to remove a specified element from a linked list.
- 32) WAP to remove first and last element from a linked list.
- 33) WAP to remove all the elements from a linked list.
- 34) WAP to shuffle elements in an ArrayList
- 35) WAP to reverse elements in an ArrayList

- 36) WAP to extract a portion of an ArrayList
- 37) WAP to iterate through all elements in an ArrayList using for each
- 38) WAP to create a new ArrayList, add some colors (string) and print the collection.
- 39) WAP to remove element from ArrayList
- 40) WAP to join two linked lists.
- 41) WAP to clone a linked list to another linked list.
- 42) WAP to remove and return the first element of a linked list.
- 43) WAP to retrieve but does not remove, the first element of a linked list.
- 44) WAP to retrieve but does not remove, the last element of a linked list.
- 45) WAP to check if a particular element exists in a linked list.
- 46) WAP to convert a linked list to array list.
- 47) WAP to compare two linked lists.
- 48) WAP to test a linked list is empty or not.
- 49) WAP to replace an element in a linked list
- 50) WAP to remove all elements from ArrayList
- 51) WAP to retain all elements from ArrayList
- 52) WAP to know how many elements in ArrayList
- 53) WAP to join two ArrayLists
- 54) WAP to clone an ArrayList to another ArrayList
- 55) WAP to shuffle the elements in a linked list.
- 56) Write a method that receives an array of int and returns the sum of every element in the array.

***Interview Questions:***

- 1) Explain Collections hierarchy?
- 2) Explain Difference between Vector and Array List?
- 3) What is the difference between array and array list?
- 4) How to sort array list?
- 5) How to create and initialize array list?
- 6) What is a linked list?
- 7) How to reverse a linked list
- 8) What is singly linked list?
- 9) What is the difference between Array and linked list?
- 10) Which collection would you choose if you want no duplicates and if objects are not stored in an order?
- 11) How will you use the Comparator interface in your class file?
- 12) What are the activities that can be performed in collection API?
- 13) In collection, which method is used to remove the head of the queue?
- 14) Which collection class method is not synchronized but allows
- 15) growing or shrinking its size and provides indexed access to its elements?
- 16) How will you extract elements from a collection without knowing

- 17) how the collection is implemented?
- 18) What are the different Collection Interfaces in Java collection framework? Distinguish them.
- 19) What is Hashtable? What is its significance in Java?

## **Topic: Collections Set**

### ***Sub-Topics:***

- 1) *Introduction*
- 2) *HashSet-declaration, constructors, methods*
- 3) *SortedSet*
- 4) *TreeSet*
- 5) *LinkedHashSet*

### ***C/W Assignment:***

Tip: Trainors will assist for these questions.

- 1) WAP to retrieve and remove the lowest element of a TreeSet using a single method call. Repeat the same using 2 different method calls.
- 2) WAP to convert a HashSet to an array.
- 3) WAP to remove all of the elements from a HashSet.
- 4) WAP to add user defined objects of type Employee in a HashSet using duplicate Employee object.

### ***H/W Assignment :***

Tip: Trainors will not assist for these questions.

- 1) WAP to iterate through all elements in a TreeSet.
- 2) WAP to create a HashSet with Integer objects without using generics
- 3) WAP to create a HashSet with some colors (String)
- 4) WAP to create a HashSet from an ArrayList
- 5) WAP to iterate through all elements in a HashSet and print the elements. Observe the order of elements.
- 6) WAP to get the number of elements in a HashSet.
- 7) WAP to get the first and last elements in a TreeSet.
- 8) WAP to get the number of elements in a TreeSet.
- 9) WAP to create a reverse order view of the elements contained in a given TreeSet.
- 10) WAP to remove a given element from a TreeSet.
- 11) WAP to empty a HashSet.
- 12) WAP to test if a HashSet is empty or not.
- 13) WAP to create a TreeSet from a HashSet.
- 14) WAP to create a new TreeSet, add Strings and print the TreeSet.

- 15) WAP to get the first and last elements in a LinkedHashSet
- 16) WAP to iterate through all elements in a LinkedHashSet and print the elements. Observe the order of elements.
- 17) WAP to add user defined objects of type Employee in a HashSet. Print the contents in the Set.

## **Topic: Queue**

### ***Sub-Topics:***

- 1) *Introduction*
- 2) *Queue-declarations,Constructors,methods*
- 3) *PriorityQueue- declarations,Constructors,methods*

### ***C/W Assignment:***

Tip:Trainors will assist for these questions.

- 1) Use 2 different method calls to add elements to a queue.
- 2) WAP to check if a queue has values
- 3) WAP to create a Queue with Integer objects
- 4) WAP to attempt to remove non-existing elements from a queue
- 5) WAP to implement your own implementation of Queue using an array internally

### ***H/W Assignment:***

Tip:Trainors will not assist for these questions.

- 1) WAP to create a Queue with Integer objects
- 2) WAP to check the top element in a queue
- 3) WAP to create a Queue with user defined class objects &
- 4) WAP to remove an element from a queue
- 5) Use 2 different method calls to remove elements from a queue

### ***Interview Questions:***

- 1) Explain Difference between HashMap and HashTable?
- 2) Explain Difference between Iterator and ListIterator?
- 3) Why Map interface does not extend Collection interface?
- 4) How HashSet store elements?
- 5) Difference between iterator and ListIterator?
- 6) Can a null element added to a TreeSet or HashSet?
- 7) What are IdentityHashMap and WeakHashMap?



- 8) When to use HashMap or TreeMap?
- 9) How to make a collection read only?
- 10) How to make a collection thread safe?
- 11) What is difference between fail-fast and fail-safe?
- 12) What is Comparable and Comparator interface?
- 13) What are Collections and Arrays class?
- 14) What is Queue and Stack? List their differences?

# **Topic: Collections Sorting**

## ***Sub-Topics:***

- 1) *Introduction*
- 2) *Sorting in collection*
- 3) *String objects*
- 4) *User-defined class objects*
- 5) *Comparable interface*
- 6) *Comparator interface*
- 7) *Comparable vs Comparator*

## ***C/W Assignment:***

Tip: Trainors will assist for these questions.

- 1) WAP to create a class Student with (rollNo, name and age). Create 3 Comparator implementations for each Student attribute (i.e. rollNo, name and age)
- 2) Sort arraylist of employees in ascending order of their salaries. If salary is same, list should be in descending order of name.

## ***H/W Assignment:***

Tip: Trainors will not assist for these questions.

- 1) What will happen if compare method returns only +1. Show example.
- 2) What will happen if compare method always returns -1. Show example
- 3) What will happen if compare method always returns 0; Show example.
- 4) Reverse an arraylist of 10 integers.
- 5) WAP to sort the elements of List that contains String objects. Print ArrayList. Sort using Collections.sort(list) method
- 6) Sort arraylist of integers without using sort method.
- 7) Create class Employee having name, age, salary as variables. Add data of 5 Employees in any Collection Class. Print the values of the object by sorting on the basis of name and salary. **(March Monthly).**

# Topic: Collections Map

## **Sub-Topics:**

- 1) *Introduction*
- 2) *Methods-put,putAll,get etc.*
- 3) *HashMap*
- 4) *SortedMap*
- 5) *TreeMap*
- 6) *HashTable vs HashMap*

## **C/W Assignment:**

Tip:Trainors will assist for these questions.

- 1) WAP to add elements to a HashMap without using generics (ie do not use <>) and print content of it. Use Integer as Key and String as Value. In second HashMap add elements of String type as Key and Integer as Value.
- 2) WAP add elements to HashMap without using generics, 0th location use String as key and Integer as value, on 1st location use String as key String and Integer as value.
- 3) WAP to get a key-value mapping associated with the greatest key less than or equal to the given key
- 4) WAP to get a reverse order view of the keys contained in a given map TreeMap

## **H/W Assignment:**

Tip:Trainors will not assist for these questions.

- 1) WAP to create a Treemap which contains Strings
- 2) WAP to create a Treemap which contains Integers
- 3) WAP to search a key in a TreeMap
- 4) WAP to search a value in a TreeMap
- 5) WAP to get all keys from the given a TreeMap
- 6) WAP to get only the Keys from a HashMap
- 7) WAP to get only the Values from a HashMap
- 8) WAP to delete all elements from a given TreeMap
- 9) WAP to copy a TreeMap content to another TreeMap
- 10) WAP to get all the entries from a HashMap. Iterate the entries and print the Key & Value values
- 11) WAP to create a TreeMap with Integer as key and get a key-value mapping associated with the greatest key and the least key in a map
- 12) WAP to get the first (lowest) key and the last (highest) key currently in a TreeMap
- 13) WAP to sort keys in TreeMap by using comparator
- 14) There is a hash map which has student object as key and marks as integer.Create two array lists from this hash map. In one array list called 'pass Students'Insert all students who has marks > 35 and in another array list 'failed Students'add all students who has less than 35 marks.(Jun Monthly).

## **Interview Questions:**

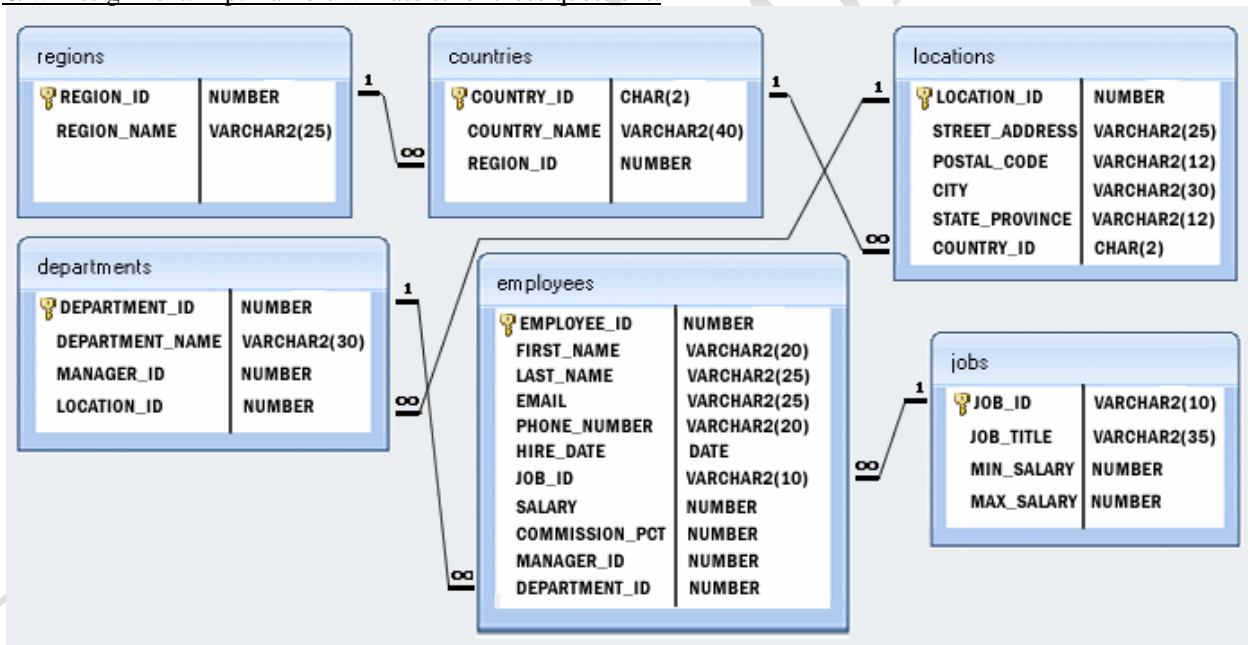
- 1) Why we use map interface?
- 2) What are the main classes implementing Map interface?
- 3) How Hashmap works?
- 4) When to use HashMap or TreeMap?

# Topic: SQL

## **Sub-Topics:**

- 1) Introduction
- 2) Syntax, Data Types
- 3) Tables, fields
- 4) Keys
- 5) Operators
- 6) Create, select, drop, Insert
- 7) Joins, Where, update etc.
- 8) Conditions
- 9) Functions
- 10) View
- 11) Store Procedure
- 12) triggers

C/W Assignment: Tip: Trainors will assist for these questions.



1. Write a query to get the names (first\_name, last\_name), salary, PF of all the employees (PF is calculated as 12% of salary).
2. Write a query to get the employee ID, names (first\_name, last\_name), salary in ascending order of salary.
3. Write a query to display the department name, manager name, and city.
4. Write a query to get the maximum and minimum salary from employees table.
5. Write a query to display the names (first\_name, last\_name) and department ID of all employees in departments 30 or 100 in ascending alphabetical order by department ID.

- 6.
  7. Write a query to display department name, name (first\_name, last\_name), hire date, salary of the manager for all managers whose experience is more than 15 years.
  8. Write a query to display the job history that were done by any employee who is currently drawing more than 10000 of salary
  9. Write a query to display the names (first\_name, last\_name) and salary for all employees whose salary is not in the range \$10,000 through \$15,000.
  10. Write a query to display the names (first\_name, last\_name) and salary for all employees whose salary is not in the range \$10,000 through \$15,000 and are in department 30 or 100.
  11. Write a query to get nth min salaries of employees.
2. Create a database structure as follows (**April Monthly**)
- Book table– Book id , book name , author id , price
- Author table – author id , author name , ph no , email , address, city
- Awards table – award id , award type id, author id , book id , year
- Awards master table – award type id , award name , award price
- i. Create above table. Create appropriate foreign and primary keys. Insert relevant data in all tables ( insert at least 5 records in each table)[1M]
  - ii. Write a query which displays list of author name, book name , price with costliest book at the top.[1M]
  - iii. Write a query which displays city in which maximum authors have won awards.[1M]
  - iv. Write a query to list bookname, authorname, award name of the author who has won maximum awards.[1M]
  - v. Write a JDBC Program which displays list of author name, book name , price with costliest book at the top.[3M]
  - vi. Write a JDBC Program to list authorname& award name of the author who has won maximum awards.[3M]

***H/W Assignment:***

Tip: Trainors will not assist for these questions.

1. Write a query to get the average salary and number of employees in the employees table
2. Write a query to display the names (first\_name, last\_name) using alias name "First Name", "Last Name"
3. Write a query to get unique department ID from employee table.
4. Write a query to get the total salaries payable to employees.
5. Write a query to get all employee details from the employee table order by first name, descending.
6. Write a query to get the number of employees working with the company
7. Write a query to get the number of jobs available in the employees table
8. Write a query get all first name from employees table in upper case
9. Write a query to get the first 3 characters of first name from employee table
10. Write a query to get the names (for example Ellen Abel, SundarAnde etc.) of all the employees from employees table
11. Write a query to get first name from employees table after removing white spaces from both side

12. Write a query to display the last names of employees whose names have exactly 6 characters.
13. Write a query to find the addresses (location\_id, street\_address, city, state\_province, country\_name) of all the departments.
14. Write a query to find the names (first\_name, last\_name), department ID and name of all the employees.
15. Find the names (first\_name, last\_name), job, department number, and department name of the employees who work in London.
16. Write a query to find the employee id, name (last\_name) along with their manager\_id, manager name (last\_name).
17. Find the names (first\_name, last\_name) and hire date of the employees who were hired after 'Jones'.
18. Write a query to get the length of the employee names (first\_name, last\_name) from employees table
19. Write a query to check if the first\_name fields of the employees table contains numbers.
20. Write a query to display the names (first\_name, last\_name) and hire date for all employees who were hired in 1987.
21. Write a query to display the first\_name of all employees who have both "b" and "c" in their first name.
22. Write a query to display the last names of employees having 'e' as the third character.
23. Write a query to display the jobs/designations available in the employees table.
24. Write a query to select all record from employees where last name in 'BLAKE', 'SCOTT', 'KING' and 'FORD'.
25. Write a query to list the number of jobs available in the employees table.
26. Write a query to get the total salaries payable to employees.
27. Write a query to get 3 minimum salaries.
28. Write a query to get the minimum salary from employees table.
29. Write a query to get the average salary and number of employees working the department 90. Write a query to get the highest, lowest, sum, and average salary of all employees.
30. Write a query to get the number of employees with the same job.
31. Write a query to get the difference between the highest and lowest salaries.
32. Write a query to find the manager ID and the salary of the lowest-paid employee for that manager.
33. Write a query to get the department ID and the total salary payable in each department.
34. Write a query to get the average salary for each job ID excluding programmer.
35. Write a query to get the total salary, maximum, minimum, average salary of employees (job ID wise), for department ID 90 only.
36. Write a query to get the average salary for all departments employing more than 10 employees.
37. Write a query to find the names (first\_name, last\_name) of the employees who are managers.
38. Write a query to find the names (first\_name, last\_name) and the salaries of the employees who have a higher salary than the employee whose last\_name='Bull'.
39. Write a query to find the 5th maximum salary in the employees table.
40. Write a query to find the names (first\_name, last\_name) of the employees who have a manager and work for a department based in the United States. Hint : Write single-row and multiple-row subqueries
41. Write a query to fetch even numbered records from employees table.
42. Write a query to find the 4th minimum salary in the employees table.

43. Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is greater than the average salary.
44. Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.
45. Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.
46. Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than Mr. Bull.
47. Write a query to get the department name and number of employees in the department.
48. Find the employee ID, job title, number of days between ending date and starting date for all jobs in department 90 from job history.
49. Write a query to display the department ID, department name and manager first name.
50. Write a query to display the job title and average salary of employees.
51. Display job title, employee name, and the difference between salary of the employee and minimum salary for the job.
52. Write a query to display the last name, job, and salary for all employees whose job is that of a Programmer or a Shipping Clerk, and whose salary is not equal to \$4,500, \$10,000, or \$15,000.
53. Write a query to display the names (first\_name, last\_name), salary and PF (15% of salary) of all employees.

***Interview Questions:***

1. What is DBMS
2. What is SQL
3. Explain Database
4. What are tables and fields?
5. What are keys? Explain different types of keys.
6. What is the difference between primary key and foreign key?
7. What is a join? Explain different types of joins.
8. What is a query?
9. What are functions in SQL?

***Good To Have:***

- 1) **Consider following tables. Write Correct query for following statements.(5\*3=15 marks)(March Monhly)**

1. Write a SQL statement to find the total purchase amount of all orders
2. Write a SQL statement to make a list with order no, purchase amount, customer name and their cities for those orders which order amount between 500 and 2000.

3. Write a SQL statement to find the details of a order i.e. order number, order date, amount of order, which customer gives the order and which salesman works for that customer and how much commission he gets for an order.
4. Write a SQL statement that sorts out the customer and their grade who made an order. Each of the customers must have a grade and served by at least a salesman, who belongs to a city.
5. Write a query to display all the orders from the orders table issued by the salesman 'Paul Adam'.

**Sample table: orders**

ord\_no purch\_amt ord\_date customer\_id salesman\_id

|       |         |            |      |      |
|-------|---------|------------|------|------|
| 70001 | 150.5   | 2012-10-05 | 3005 | 5002 |
| 70009 | 270.65  | 2012-09-10 | 3001 | 5005 |
| 70002 | 65.26   | 2012-10-05 | 3002 | 5001 |
| 70004 | 110.5   | 2012-08-17 | 3009 | 5003 |
| 70007 | 948.5   | 2012-09-10 | 3005 | 5002 |
| 70005 | 2400.6  | 2012-07-27 | 3007 | 5001 |
| 70008 | 5760    | 2012-09-10 | 3002 | 5001 |
| 70010 | 1983.43 | 2012-10-10 | 3004 | 5006 |
| 70003 | 2480.4  | 2012-10-10 | 3009 | 5003 |
| 70012 | 250.45  | 2012-06-27 | 3008 | 5002 |
| 70011 | 75.29   | 2012-08-17 | 3003 | 5007 |
| 70013 | 3045.6  | 2012-04-25 | 3002 | 5001 |

**Sample table: customer**

| customer_id | cust_name    | city       | grade | salesman_id |
|-------------|--------------|------------|-------|-------------|
| 3002        | Nick Rimando | New York   | 100   | 5001        |
| 3005        | Graham Zusi  | California | 200   | 5002        |
| 3001        | Brad Guzan   | London     |       | 5005        |
| 3004        | Fabian Johns | Paris      | 300   | 5006        |
| 3007        | Brad Davis   | New York   | 200   | 5001        |
| 3008        | Julian Green | London     | 300   | 5002        |
| 3003        | Jozy Altidor | Moscow     | 200   | 5007        |

**Sample table: salesman**

| salesman_id | name       | city     | commission |
|-------------|------------|----------|------------|
| 5001        | James Hoog | New York | 0.15       |
| 5002        | Nail Knite | Paris    | 0.13       |
| 5005        | Pit Alex   | London   | 0.11       |
| 5006        | Mc Lyon    | Paris    | 0.14       |
| 5003        | Lauson Hen | San Jose | 0.12       |
| 5007        | Paul Adam  | Rome     | 0.13       |

1. Create the database as per below details and enter atleast 7 records in each table and perform the following:  
Consider following database exist in the „Facebook“ database  
FBUsers – user id , user name , email , phone no , total posts



Post – post id , user id , postContent.

Friendship – user id , friend id, friendship status code

Friendship status - Status code, Status ( 1 family , 2 Friend , 3 colleague , 4classmate)

a) Write a query to list all colleagues (username) userId, postContent.

b) Show no of family, no of friend, no of colleague for each userId sorted by userid.

c) WAQ to delete all posts of user name = "Ankita".

d) JDBC program: Insert 1 record in FBUsers table after that insert another record in Friendship status

E.g. new record "5, Enemy". Now assume second record is going to fail in such a scenario throw an user defined exception and handle it with user defined exception.

# **Topic: JDBC**

## ***Sub-Topics:***

1. *Introduction*
2. *JDBC Drivers*
3. *Connectivity Steps*
4. *Connection with SQL*
5. *Driver Manager*
6. *Connection*
7. *Statement*
8. *Resultset*
9. *Prepared Statement*

## ***C/W Assignment:***

Tip: Trainors will assist for these questions.

1. How to create JDBC connection.
2. WAP using JDBC to select and print employee names along with their manager names.
3. WAP using JDBC to transfer employees from manager with id 10 to manager with id 1. Delete employee with id 10. Do all using commit and rollback.
4. WAP for batch update using Prepared Statement.

## ***H/W Assignment:***

Tip: Trainors will not assist for these questions.

1. WAP to execute and read select queries using JDBC?
2. WAP using JDBC to select and print all country names.
3. WAP to update a record in the database using JDBC?
4. WAP using JDBC to delete employees if their salaries are equal to 0.
5. WAP to enter and display data of students using database connections.
6. Write a program for batch update using Statement.

## ***Interview Questions:***

1. What is JDBC?
2. What are drivers? Explain
3. What are the different drivers in JDBC?
4. What is Statement in JDBC.?
5. What are the types of JDBC statements available?
6. What is database metadata?
7. What do you understand by result set?