CSS CORP INTERVIEW QUESTIONS

CORE JAVA QUESTIONS

1. What is mean by inheritance?

Inheritance is used to reduce the object memory

- We can access one class property into another class using 'extend' keyword is called inheritance

- Reusable purpose
- It has 5 types

1. Single Inheritance
2. Multilevel Inheritance
3. Multiple Inheritance
4. Hybrid Inheritance
5. Hierarchical Inheritance

1. Single Inheritance:

- One parent class is directly support into one child class using extend keyword

2. Multilevel Inheritance:

- More than one parent class support into one child class using extends keyword

3. Multiple Inheritance:

- More than one parent class parallely support into one child class but it won't support in java because

- Priority problem
- Compilation error/syntax error

(i.e.) if both parent class having same method name it gets priority problem so it doesn't work in java

- but multiple inheritance support in java using interface

4. Hybrid inheritance:

- It's a combination of single and multiple inheritance

5. Hierarchical Inheritance:

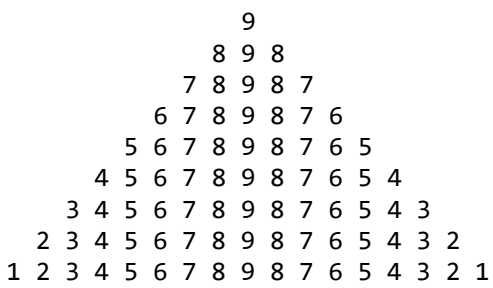
- One parent class directly support into more than one child class

2. How to access one class property into another class?

2 ways we can access

1. by using extends keyword(inheritance)
2. By creating object

3. Construct the triangle



Program:

```
publicclassReverseString {
        publicstaticvoid main(String[] args)
          {
            Scanner sc = newScanner(System.in);

        System.out.println("How Many Rows You Want In Your Pyramid?");

        intnoOfRows = sc.nextInt();
        introwCount = 1;

        System.out.println("Here Is Your Pyramid");

        for (inti = noOfRows; i>= 1; i--)
              {
        //Printing i*2 spaces at the beginning of each row

        for (intj = 1; j<= i*2; j++)
                {
        System.out.print(" ");
                }

        //Printing j where j value will be from i to noOfRows

        for (intj = i; j<= noOfRows; j++)
                {
        System.out.print(j+" ");
                }

        //Printing j where j value will be from noOfRows-1 to i

        for (intj = noOfRows-1; j>= i; j--)
                {
        System.out.print(j+" ");
                }

        System.out.println();

        //Incrementing the rowCount
```

```
            rowCount++;
                }
              }
            }
```

Output:

```
How Many Rows You Want In Your Pyramid?
9
Here Is Your Pyramid
                  9
                8 9 8
              7 8 9 8 7
            6 7 8 9 8 7 6
          5 6 7 8 9 8 7 6 5
        4 5 6 7 8 9 8 7 6 5 4
      3 4 5 6 7 8 9 8 7 6 5 4 3
    2 3 4 5 6 7 8 9 8 7 6 5 4 3 2
  1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1
```

4. **Write a program to find sum of each digit in the given number using recursion?**

Program:

```
publicclassMyNumberSumRec {

intsum = 0;

publicintgetNumberSum(intnumber){

if(number == 0){
returnsum;
        } else {
sum += (number%10);
getNumberSum(number/10);
        }
returnsum;
    }

publicstaticvoid main(String a[]){
MyNumberSumReca = newMyNumberSumRec();
System.out.println("Sum is: "+a.getNumberSum(5678));
    }
      }
```

Output:

```
Sum is: 26
```

5. Longest substring without repeating characters

INPUT		OUTPUT
java2novice	=	a2novice
java_language_is_sweet	=	uage_is
java_java_java_java	=	va_j, _jav
abcabcbb	=	bca, abc, cab

program:

```
publicclassMyLongestSubstr {

private Set<String>subStrList = newHashSet<String>();
privateintfinalSubStrSize = 0;

public Set<String>getLongestSubstr(String input){
//reset instance variables
subStrList.clear();
finalSubStrSize = 0;
// have a boolean flag on each character ascii value
boolean[] flag = newboolean[256];
intj = 0;
char[] inputCharArr = input.toCharArray();
for (inti = 0; i<inputCharArr.length; i++) {
charc = inputCharArr[i];
if (flag[c]) {
extractSubString(inputCharArr,j,i);
for (intk = j; k<i; k++) {
if (inputCharArr[k] == c) {
j = k + 1;
break;
                    }
flag[inputCharArr[k]] = false;
                }
            } else {
flag[c] = true;
            }
        }
extractSubString(inputCharArr,j,inputCharArr.length);
returnsubStrList;
    }

private String extractSubString(char[] inputArr, intstart, intend){

StringBuildersb = newStringBuilder();
for(inti=start;i<end;i++){
sb.append(inputArr[i]);
        }
```

```
        String subStr = sb.toString();
if(subStr.length() >finalSubStrSize){
finalSubStrSize = subStr.length();
subStrList.clear();
subStrList.add(subStr);
        } elseif(subStr.length() == finalSubStrSize){
subStrList.add(subStr);
        }

returnsb.toString();
    }

publicstaticvoid main(String a[]){
MyLongestSubstrmls = newMyLongestSubstr();
System.out.println(mls.getLongestSubstr("java2novice"));
System.out.println(mls.getLongestSubstr("java_language_is_sweet"));
System.out.println(mls.getLongestSubstr("java_java_java_java"));
System.out.println(mls.getLongestSubstr("abcabcbb"));
    }
      }
```

Output :

```
[a2novice]
[uage_is]
[va_j, _jav]
[bca, abc, cab]
```

6. Kth largest or smallest element in an array

Example : if given array is [1,3,12,19,13,2,15] and you are asked for the 3rd largest element i.e., k=3 then your program should print 13

Program:

```
publicclassThirdLarge {
            publicstaticvoid main(String[] args) {
                  inta[]={1,3,12,19,13,2,15};
                  for(inti=0;i<a.length;i++){
                        for(intj=i+1;j<a.length;j++){
                              inttemp=0;
                              if(a[i]<a[j]){
                                    temp=a[j];
                                    a[j]=a[i];
                                    a[i]=temp;

                              }
                        }
                  }
                  for(intk=0;k<a.length;k++){
                        System.out.println(a[k]);
```

```
		}
		System.out.println("The Third maximum number is :" + a[a.length-5] );
		}
		}
```

Output:

```
19
15
13
12
3
2
1
The Third maximum number is :13
```

7. Armstrong number:

Program:

```
publicclassArmstrongNumberCheck {
	publicstaticvoid main(String[] args) {

		intn, a, i = 0, j = 0;
		Scanner an = newScanner(System.in);
		System.out.println("Enter a number");
		n = an.nextInt();
		a = n;

		while (a> 0) {
			i = a % 10;
			j = j + (i * i * i);
			a = a / 10;

		}
		if (n == j) {
			System.out.println("Armstrong number");

		} else {
			System.out.println("Not armstrong Number");
		}

	}
	}

	Output :


		Enter a number
		371
Armstrong number
```

8. Write a program to remove duplicates from sorted array

Input : 2,3,6,6,9,10,10,10,12,12
Output : 2,3,6,9,10,12

Program:

```
publicclassMyDuplicateElements {

publicstaticint[] removeDuplicates(int[] input){

intj = 0;
inti = 1;
//return if the array length is less than 2
if(input.length< 2){
returninput;
        }
while(i<input.length){
if(input[i] == input[j]){
i++;
}else{
input[++j] = input[i++];
            }
        }
int[] output = newint[j+1];
for(intk=0; k<output.length; k++){
output[k] = input[k];
        }

returnoutput;
    }

publicstaticvoid main(String a[]){
int[] input1 = {2,3,6,6,8,9,10,10,10,12,12};
int[] output = removeDuplicates(input1);
for(inti:output){
System.out.print(i+" ");
        }
    }
      }
```

Output:

```
2 3 6 8 9 10 12
```

9. Binary search

Program:

```
publicclassMyBinarySearch {

publicintbinarySearch(int[] inputArr, intkey) {

intstart = 0;
intend = inputArr.length - 1;
while (start<= end) {
intmid = (start + end) / 2;
if (key == inputArr[mid]) {
returnmid;
            }
if (key<inputArr[mid]) {
end = mid - 1;
            } else {
start = mid + 1;
            }
        }
return -1;
    }

publicstaticvoid main(String[] args) {

MyBinarySearchmbs = newMyBinarySearch();
int[] arr = {2, 4, 6, 8, 10, 12, 14, 16};
System.out.println("Key 14's position: "+mbs.binarySearch(arr, 14));
int[] arr1 = {6,34,78,123,432,900};
System.out.println("Key 432's position: "+mbs.binarySearch(arr1, 432));
    }
      }
```

Output:

```
Key 14's position: 6
Key 432's position: 4
```

10. Butterfly shuffle:
Program:

```
importjava.util.ArrayList;
importjava.util.Scanner;

public class SampleTest {
                public static void main(String[] args) {
                 int a[] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 0 };
                 intlen = a.length / 2;
                 for (int i = len - 1; i >= 0; i--) {
                          System.out.println(a[i]);
                 }
                 for (int i = a.length - 1; i >= len; i--) {
                                  System.out.println(a[i]);
                 }

                }
}
```

Output:

```
5
4
3
2
1
0
9
8
7
6
```

11. Write code for JDBC connection using mysql database (retrieve the values)

```
package org.test;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.List;

public class Test {
	public void getEmployeeInfo() {
		Connection con = null;
		try {
			// load the driver
			Class.forName("com.mysql.jdbc.Driver");
			// connect to db
			con = DriverManager.getConnection(
					"jdbc:mysql://localhost:3306/selenium_schema",
"root", "");
			// prepare the st
			PreparedStatement ps = con
					.prepareStatement("select * from employee_table
where empid='2'");

			// Execute query
			ResultSet rs = ps.executeQuery();
			while (rs.next()) {
				int id = rs.getInt("empid");
				String name = rs.getString("name");
				String pass = rs.getString("password");
				long ph = rs.getLong("phone");
				System.out.println(id);
				System.out.println(name);
				System.out.println(pass);
				System.out.println(ph);
			}
		} catch (ClassNotFoundException e) {
			// TODO Auto-generated catch block
			e.printStackTrace();
		} catch (SQLException e) {
			// TODO Auto-generated catch block
			e.printStackTrace();
		} finally {
			try {
				con.close();
			} catch (SQLException e) {
				// TODO Auto-generated catch block
				e.printStackTrace();
			}}}}
```

SELENIUM QUESTIONS

1. Explain about POM frame work?

POM:

- Page Object Model
- POM is an object repository design pattern in selenium webdriver
- POM creates our testing code maintainable and reusable
- Page factory is an optimized way to create object repository

POM Steps & Rules:

- Create a maven project/java project
- We have to create 3 source folders(packages)

1. Src/main/java
- It contains POM information (i.e) locators of every page in separate class

2. Src/test/java (Junit,TestNG)
- It contains login, asserts & registration

3. src/resources/java
- It contains reusable methods/codes. Ex, browser coding, radio button, scroll down and etc.

2. INput[Id="123"]>/

L1='541'

L2='541'

L3='541'

L4='541' find the xpath of last li?

//input[@id='123']//following::l4

3. What framework u worked in your project ?

POM
JUnit
Data Driven
Cucumber

4. **Where you keep your test data and how to read values from excel sheet?**

```
package Readexceldata;

import java.io.File;
import java.io.FileInputStream;
import org.apache.poi.xssf.usermodel.XSSFSheet;
import org.apache.poi.xssf.usermodel.XSSFWorkbook;

publicclass ReadExcel {

	publicstaticvoid main(String[] args) throws Exception {
	File src= new File("D:\\Exceldetails\\testdata.xlsx");

	FileInputStream fls= new FileInputStream (src);
	 System.out.println(fls);

	XSSFWorkbook wb= new XSSFWorkbook(fls);
	XSSFSheet  sheet1 =     wb.getSheetAt(0);
	String  data0 = sheet1.getRow(0).getCell(0).getStringCellValue();
	System.out.println("Data from Excel is "+data0);

	}

}
```

5. **Windows handling scenario: Like if I am currently in main window , first I have to close 2nd window then move to 3rd window and need to print "we are in 3rd window" again move to 4th window and print "4th window" then switch back to main window ?**

```
String parentWindowId = driver.getWindowHandle();
		Set<String> allWindows = driver.getWindowHandles();
		int windowCount = 0;
		for (String x : allWindows) {
			windowCount++;
			if (!parentWindowId.equals(x)) {
				if (windowCount == 2) {
					driver.switchTo().window(x);
					driver.close();
					continue;
				}
				driver.switchTo().window(x);
					System.out.println("I am in " + windowCount +
"window");

			}
			driver.switchTo().defaultContent();}
```

SQL QUESTIONS

1. **Write inner join query print the employee and their salary(Assume employee and salary Details table) ?**

SELECT E.NAME,S.SALARY FROM EMPLOYEE E, SALARYDETAILS S
WHERE E.ID=S.ID;

2. **Write inner join query print the employee and their salary and those who are all getting more than 50k(Assume employee and salary Details table) ?**

SELECT E.NAME,S.SALARY FROM EMPLOYEE E, SALARYDETAILS S
WHERE E.ID=S.ID and salary>50000;

3. **Find the sum of salary in salary details table?**

SELECT SUM(SALARY) FROM SALARYDETAILS

4. **Find the 2nd minimum salary in employee table?**

with e as
(select first_name,salary,dense_rank()over(order by salary asc) as RK
from employees)
*select * from E where Rk<=2;*

5. **Find the 2nd minimum salary in employee table?**

with e as
(select first_name,salary,dense_rank()over(order by salary asc) as RK
from employees)
*select * from E where Rk<=3;*