**Week 3 Day 1- Assignment**

**Assignment 1 to 6 in GitHub**

https://github.com/GomathiSiddaiyan/Selenium\_Week3.Day1

============================================================================

**Class Assignment:: 7**

**Problem statement: Polymorphism- Method override**

**Eclipse Program name: Create multiple getStudentInfo() method by passing id argument alone, by id & name, by email & phoneNumber.**

**Pseudocode:**

Class: Students

Methods: getStudentInfo()

**Description:**

Create multiple getStudentInfo() method by passing id argument alone, by id & name, by email & phoneNumber

**Answer: 7**

package week3.day1;

public class StudentsInfo {

public void getStudentInfo() {

// getStudentInfo

System.out.println("Student Information:");

}

public void getStudentInfo(int stuId) {

// getStudentInfo

System.out.println("Student ID: "+stuId);

}

public void getStudentInfo(int stuId, String stuName) {

// getStudentInfo

System.out.println("Student Id: "+stuId);

System.out.println("Student Name: "+stuName);

}

public void getStudentInfo(String stuEmail, long phNo) {

// getStudentInfo

System.out.println("Student Email Id: "+stuEmail);

System.out.println("Student Phone Number: "+phNo);

}

public static void main(String[] args) {

// TODO Auto-generated method stub

System.out.println("Method Override Program");

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

StudentsInfo std=new StudentsInfo();

std.getStudentInfo();

std.getStudentInfo(130);

std.getStudentInfo(130, "Gomathi");

std.getStudentInfo("gomusidyan@gmail.com", 987543210);

}

}

**Output: 7**

Method Override Program

===============================

Student Information:

Student ID: 130

Student Id: 130

Student Name: Gomathi

Student Email Id: gomusidyan@gmail.com

Student Phone Number: 987543210

============================================================================

**Class Assignment:: 8**

**Problem statement: Inheritance Types**

**Eclipse Program name:**

**Pseudocode:**

A flow chart to show the single level, Hierarchical ,Hybrid and multilevel inheritance in Selenium. (That which was discussed in the class)

\*\*Mention Keyword

**Answer: 8**

Keyword used for inheritance is **extends**.

**Types of Inheritance::**

**Single Inheritance::**

- One child class extends one parent class.

**Multi-level Inheritance::**

- More than one level of inheritance (Child extends - parent, parent extends Grandparent.)

**Hierarchical Inheritance::**

- Multiple child class for one parent.

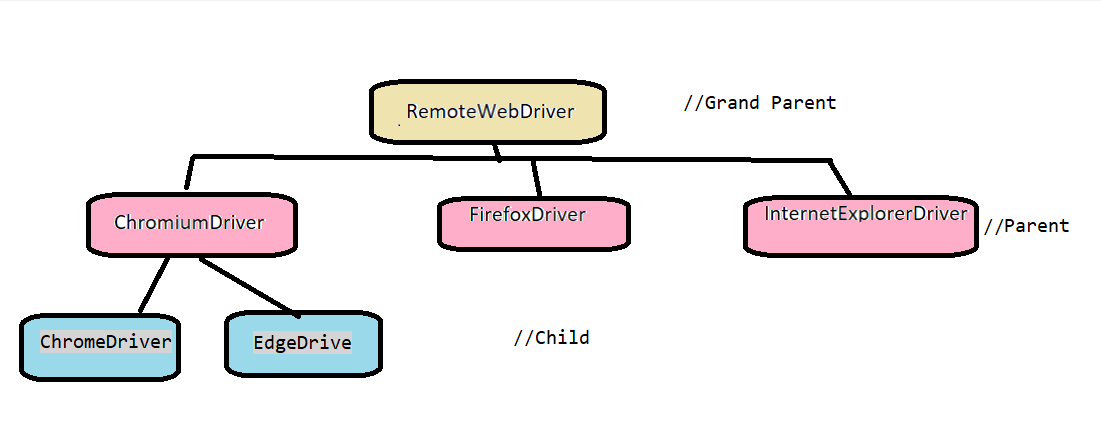
**Hybrid Inheritance::**

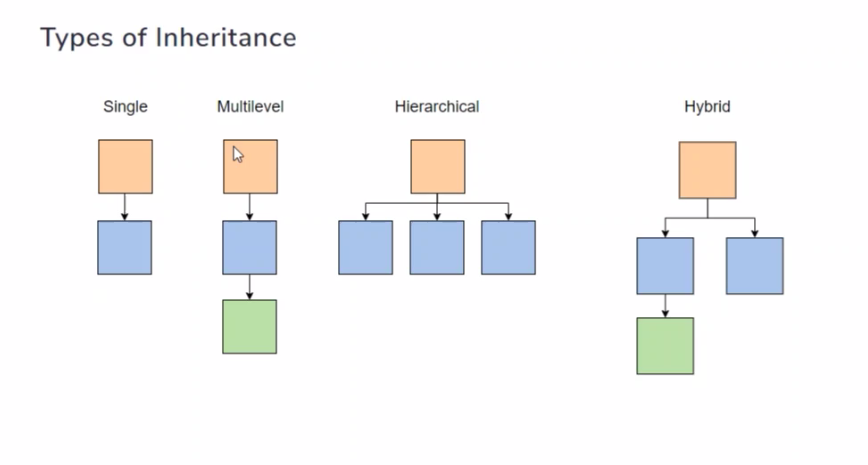
- Combining Multi-level inheritance and Hierarchical Inheritance.

**Multiple Inheritance::**

- One child class extends Multiple parent class( This Inheritance is not allowed in JAVA)

Example::





**Grand Parent:** - RemoteWebDriver

**Parent:** - ChromiumDriver, firefoxDriver, InternetExplorerDriver

**Child:** - ChromeDriver, EdgeDriver

============================================================================

Few Assignments on Java (string)

**Class Assignment:: 9**

**Problem statement:**

**Eclipse Program name:**

**Pseudocode:**

String str3 = new String("Kutty");

String str4 = new String("Kutty");

if(str3==str4) {

System.out.println(" Same text");

}

else

System.out.println("Diff text");

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

👆👆 Justify the above output

**Answer: 9**

If Initializing **String to a variable**,

String str3 = "Kutty";

String str4 = "Kutty";

Str3 -- memory address (1002) -- Value (Kutty)

Str4 -- memory address (1002) -- Value (Kutty)

Because the condition ==, No duplicates values will store instead share same memory reference.

So, the output is **Same text**

In this case we are **passing value of string as arguments in a constructor**,

String str3 = new String("Kutty");

String str4 = new String("Kutty");

Because here we are passing value in a constructor, by default it will print and stores in different memory.

So, the output is **Diff text**

============================================================================

**Class Assignment:: 10**

**Problem statement:**

1. Write a Java program to get the character at the given index within the String.

String text = Java Exercise

Index for character E and s

**Answer: 10**

public class PracticeQue {

public static void main(String[] args) {

String text = "Java Exercise";

int ch=text.indexOf('E');

int ch2=text.indexOf('s');

System.out.println("Index of E: " +ch);

System.out.println("Index of s: "+ch2);

}}

**Output::10**

Index of E: 5

Index of s: 11

============================================================================

**Class Assignment:: 11**

**Problem statement:**

2. Write a Java program to compare a given string to another string, ignoring case considerations.

String 1="I am Learning Java?"

String 2="I am learning java?

\*\* Explore with == operator

\*\* equals

\*\* equalsignorecase

**Answer: 11**

System.out.println("Using equalsignorecase");

String st1 = "I am Learning Java?";

String st2 = "I am learning java?";

if (st1.equalsIgnoreCase(st2)) {

System.out.println(" Strings are equal");

} else {

System.out.println(" Strings are different");

}

System.out.println("Using equals");

if (st1.equals(st2)) {

System.out.println(" Strings are equal");

} else {

System.out.println(" Strings are different");

}

**Output::11**

Using equalsignorecase

Strings are equal

Using equals

Strings are different

=============================================================================

**Class Assignment:: 12**

**Problem statement:**

3.Write a Java program to replace a specified character with another character and add # to the given string.

String sentence = "I am working with Java8"

replace 8 to 11

Print the characters from 5 to 14

**Answer: 12**

//replace - replace 8 to 1, Print the characters from 5 to 14

String sentence = "I am working with Java8";

int length=sentence.length();

System.out.println("The length of the given string is : "+length);

String replace=sentence.replace('8', '1');

String replaceSt=replace;

String str="#";

String ConcatSt=replaceSt.concat(str);

String subSt=sentence.substring(5, 14);

System.out.println("Replaced sentence is: "+replaceSt);

System.out.println("Concatinated sentence is: "+ConcatSt);

System.out.println("After replace and sub string from the given sentece is: "+subSt);

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

**Output::12**

 The length of the given string is : 23

Replaced sentence is: I am working with Java1

Concatinated sentence is: I am working with Java1#

After replace and sub string from the given sentece is: working w

\*\*\*

============================================================================