1. Explain difference between method overloading and method overriding.

2. Implement all string functions in java.

3. Implement all stringbuffer functions in java.

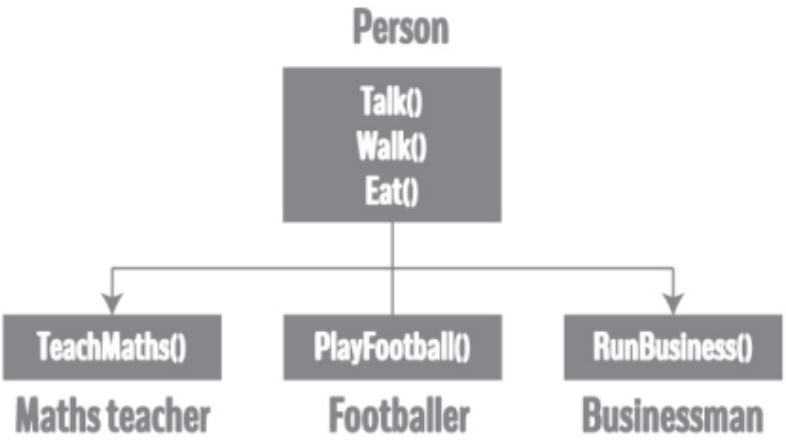
4. Explain with example declaration of string using string literal and new keyword.

5. Create a class named 'Shape' with a method to print "This is This is shape". Then create two other classes named 'Rectangle', 'Circle' inheriting the Shape class, both having a method to print "This is rectangular shape" and "This is circular shape" respectively. Create a subclass 'Square' of 'Rectangle' having a method to print "Square is a rectangle". Now call the method of 'Shape' and 'Rectangle' class by the object of 'Square' class.

6. Create game characters using the concept of inheritance. Suppose, in your game, you want three characters - a maths teacher, a footballer and a businessman. Since, all of the characters are persons, they can walk and talk. However, they also have some special skills. A maths teacher can teach maths, a footballer can play football and a businessman can run a business. You can individually create three classes who can walk, talk and perform their special skill as shown in the figure below.

In each of the classes, you would be copying the same code for walk and talk for each character. If you want to add a new feature - eat, you need to implement the same code for each character. This can easily become error prone (when copying) and duplicate codes. It'd be a lot easier if we had a Person class with basic features like talk, walk, eat, sleep, and add special

skills to those features as per our characters. This is done using inheritance.



Using inheritance, now you don't implement the same code for walk and talk for each class. You just need to inherit them. So, for Maths teacher (derived class), you inherit all features of a Person (base class) and add a new feature TeachMaths. Likewise, for a footballer, you inherit all the features of a Person and add a new feature PlayFootball and so on.

7. WAP to manage the employee allowance from a specific department by creating class structure as follow,
   InheritanceEx2
   |
   |
   InheritanceEx2Main.java
   |
   |- dept | Deparment.java
   |
   |
   |
   |- emp | Employee.java extends Department
   |
   |
   |
   |- allowance | Allowance.java extends Employee
   |
   | [Multilevel Inheritance]

8. Write a Java Program to demonstrate StringBuilder class methods.
9. Write a Java Program to demonstrate Method overriding.(create class Result with method result(). Override method result() in UGResult and PGResult class)

10. Write a java program to create a class called STUDENT with data members PRN, Name and age. Using inheritance, create a classes called UGSTUDENT and PGSTUDENT having fields as semester, fees and stipend. Enter the data for at least 5 students. Find the semester wise average age for all UG and PG students separately.

11. Implement hybrid inheritance using all access specifiers (public, private, protected).

12. Write a program to implement a class Teacher contains two fields Name and Qualification. Extend the class to Department, it contains Dept. No and Dept. Name. An Interface named as College it contains one field Name of the College. Using the above classes and Interface get the appropriate information and display it.