

## Assignment 1

### Introduction to JAVA

**CO1:** Understand and implement fundamental and OOP features through Java and Python Programming

1. Write a program to accept two short integers from user and display the sum.
2. Write a program that accepts number of command line parameters and displays the parameters and count of such parameters.
3. Write a program that accepts radius of a circle and displays area of the circle. Declare a constant pi equals to 3.14 using OOP concept.
5. Write a program that accepts a **String** and assigns it to another. Check the outcome of comparison with == and equals() method. Take two Strings and put same input for them. Repeat the equality checking. Observe the outcome.
6. Write a program where class contains **void show(int)** to display the argument passed. Call the function once with **short** as actual parameter and again **double** as actual parameter. Add another function as **void show(double)** . Repeat the calls. Observe the outcomes in each case.
7. Design and implement **Student** class with roll, name and score as attributes. It will have methods to set attributes (attribute values passed as arguments), display the attributes, copy (that copies the content of invoking object to another object passed as argument). Verify that methods are working properly.
8. Add constructors in the **Student** class of earlier problem so that objects can be created with i) roll only, ii) roll and name only, iii) roll, name and score, iv) no value. Also include a copy constructor. Check whether constructors are working or not. Verify, copy constructor results into deep copy or not.
9. Design a **BankAcct** class with account number, balance and interest rate as attribute. Interest rate is same for all account. Support must be there to initialize, change and display the interest rate. Also supports are to be there to return balance and calculate interest.
10. For a programme (such as, BCSE), each Instructor has name and phone number. Each textbook has a title, author name and publisher. Each course (that is, subject) has a course name, instructor and text book.
  - One can set the data for a textbook and view the same.
  - One can view instructor information and set the information.
  - One can set the course data and view the same.

Design and implement the classes.