ASSIGNMENT 1.1

1. Write a program to print “Hello World” on the screen.
2. Write a program that generate the following output
   10, 20, 19
   Use an integer constant for 10, an arithmetic operator to generate the 20, and a decrement operator to generate 19.
3. Write a program that asks the user to enter a radius value and then compute the volume of a sphere with the input radius.
4. Write a program that takes three input of sides of a triangle. The program should indicate whether the triangle would be formed or not. If it can be formed it also indicates the type.
5. Write a program that takes one input as number and it will display whether the number is +ve, -ve or zero. If the number is +ve, then it will display whether the number is odd or even.
6. Write a program which takes username as input and it greets to user with his name.
7. Write a program, which takes two integer numbers as input and it shows their exchanged value. (Don’t use third variable)
8. WAP to check Leap Year.
9. WAP for finding remainder of division of 2 numbers.
10. WAP to calculate Area of Rectangle.
11. WAP to calculate Area of Square.
12. WAP to calculate the area of Triangle.
13. WAP to calculate Area and Circumference of Circle.
14. WAP for two item’s weight (floating points' values) and number of purchase (floating points' values) and calculate the average value of the items.
    Test Data:
    Weight - Item1: 15
    No. of item1: 5
    Weight - Item2: 25
    No. of item2: 4
    Expected Output:
    Average Value = 19.444444
15. WAP to calculate a bike’s average consumption from the given total distance (integer value) travelled (in km) and spent fuel.
    Test Data:
    Input total distance in km: 350
    Input total fuel spent in litres: 5
    Expected Output:
    Average consumption (km/lt) 70.00
16. Write a program that will give the grade of the student based on the percentage he got in the course.
    Use the following criteria for assigning grades:
    Grade = A (when percentage >= 60)
    Grade = B (when percentage >= 50 and percentage < 60)
    Grade = C (when percentage >= 40 and percentage < 50)
    Grade = D (when percentage >= 30 and percentage < 40)
    Grade = E (when percentage >= 20 and percentage < 30)
17. WAP to check whether a number is divisible by 5.
18. WAP to input basic salary of an employee and calculate its Gross salary according to following:
    Basic Salary <= 10000 : HRA = 20%, DA = 80%
    Basic Salary <= 20000 : HRA = 25%, DA = 90%
    Basic Salary > 20000 : HRA = 30%, DA = 95%
19. WAP to input electricity unit charges and calculate total electricity bill according to the given condition:
    For first 50 units Rs. 0.50/unit
    For next 100 units Rs. 0.75/unit
    For next 100 units Rs. 1.20/unit
    For unit above 250 Rs. 1.50/unit
    An additional surcharge of 20% is added to the bill

ASSIGNMENT 1.2

1. WAP for printing all natural numbers till 20.
2. WAP for printing all natural numbers in reverse order starting from 20.
3. WAP for printing all even numbers from 1 to 20.
4. WAP for printing all odd numbers from 1 to 20.
5. WAP for adding all numbers from 1 to 20.
6. WAP for finding sum of all even numbers till 20.
7. WAP for finding sum of all odd numbers till 20.
8. WAP for printing multiplication table of a number. For eg. Display should be " 2 X 1 = 2"
9. WAP to calculate factorial of a number.
10. WAP to check whether a number is prime or not.
11. WAP to print all digits of a number and their sum.
12. WAP to print reverse of a number.
13. WAP to check whether the number is Armstrong or not.
14. WAP to print the Fibonacci series in a given range.
15. WAP to check whether the number entered is palindrome or not.

ASSIGNMENT 1.3

WAP to print following kind of patterns:

1)
2)
```
*                                                   A           A
**
***
****
******
```

3)
```
     *
    **
   ***
  ****
******
```

4)
```
    *
   ***
  *****
 *******
*********
```

5)
```
*********
****  ****
***    ***
**      **
*        *
```

6)
```
ABCD
ABC
AB
A
```

7)
```
1
12
123
1234
12345
123456
```

8)
```
 ABCDEDCBA
ABCD    DCBA
ABC       CBA
AB          BA
```

ASSIGNMENT 2 (FUNCTIONS)

Solve the questions of assignment 1.2 using functions.

Array Assignment

1. Write a program that asks the user to take array of 10 integers. The program must compute and write how many integers are greater than or equal to 10.
2. Write a program that asks the user to take array of 10 integers. The program must output the largest element in the array, and the index at which that element was found.
3. Write a program that asks the user to take array of 10 integers. The program will then sort the array in descending order and display it.
4. Write a program that asks the user to take array of 10 integers. The program will then display either "the array is growing", "the array is decreasing", "the array is constant", or "the array is growing and decreasing."
5. Write a program which takes 2 arrays of 10 integers each, a and b. c is an array with 20 integers. The program should put into c the appending of b to a, the first 10 integers of c from array a, the latter 10 from b. Then the program should display c.
6. Write a program that asks the user to take an array of 10 integer and an integer value V and an index value i between 0 and 9. The program must put the value V at the place i in the array, shifting each element right and dropping off the last element. The program must then write the final array.
7. Write a program to handle the command line arguments entered by the user.
8. Write a program to add 2 matrices.
9. Write a program to multiply 2 matrices.

Q1. Write Java program involving two classes: **OddAndEven** and **TestOddAndEven**.
OddAndEven has the following:

- Instance varaibales countOfOdd and countOfEven both of type int
- A method addNumber that takes a number as parameter and increment countOfOdd, if the number is odd, else increment countOfEven.
- A method toString that returns a string in the form: "Number of Odd: x, Number of Even : y", where x and y are the values of the instance variables.

The **TestOddAndEven** class first creates **OddAndEven** object, then in a loop, read a number and use it to call the **addNumber** method until the user enters **Q**. Finally, it prints the count of odd and even numbers entered.

Q2. Write a program to design a class to represent a bank account. Include the following members.
Date members

- Name of depositor
- Account Number
- Type of account
- Balance account in the account

Methods: -

- To assign initial values
- To deposit an account
- To withdraw an account after checking balance.
- To display the name and balance

Q3. Implement a **Student** class with the following fields, constructors and methods :
Fields:
name;
totalScore;
numberOfQuizzes;
Constructors:
public Student(String name, double score)
public Student(double score, String name)
public Student(String name) {
Methods:

public String getName()
public double getAverage() //this should return zero if no quiz has been taken.
public double getTotalScore()
public void addQuiz(double score)
public void printStudent() //this should print the student's name and average
score
public String toString()
Write an application **TestStudent** that reads a student name and use the **Student** class to create a Student object. Then read the scores of the student in three quizzes and add each to the ***totalScore*** of the student using ***addQuiz()*** method and print the student object.
(**Note:** Make use of **this** key word wherever it can be used)

Q4. WAP to calculate to count total object created for a class.

Q5. WAP to use final variable in a class.

Q6. WAP to use static block in a class.

Q1. A sales person is paid commission based on the sales he makes as shown by the following table:

SALES	**COMMISSION**
Under	2 % of SALES
SR500	
SR500	5 % of SALES
And	
under	
SR5000	**figure (a)**
SR5000	8 % of SALES
and over	

Write a class, **Commission**, which has an instance variable, **sales**; an appropriate **constructor**; and a method, **commission()** that returns the commission.
Now write a demo class to test the Commission class by reading a sale from the user, using it to create a Commission object after validating that the value is not negative. Finally, call the **commission()** method to get and print the commission. If the sales are negative, your demo should print the message "Invalid Input".

Q2. The certain instructor assigns letter grade for his course based on the following table:

Score	Grade	
>= 90	A+	
>= 85	A	
>= 80	B+	
>= 75	B	
>= 65	C+	figure (b)
>= 60	C	
>= 55	D+	
>= 50	D	
< 50	F	

Write a class, **Grader**, which has an instance variable, **score**, an appropriate **constructor** and appropriate **methods**

- a method, letterGrade() that returns the letter grade as a String.

Now write a demo class to test the Grader class by reading a score from the user, using it to create a Grader object after validating that the value is not negative and is not greater then 100. Finally, call the **letterGrade()** method to get and print the grade. See figure (b) for sample run.

Q3. Design a class Circle and implement the following methods:

- Define a circle method to compute its area

- Define a circle method to compute its perimeter
- Define a method that takes a given point represented by a pair of numbers and checks whether or not the point is inside the circle.

The circle class needs to have instance variables to store the radius of the circle, and the x and y coordinates of the center. Add main program to test the class Circle repeatedly, until user enters negative value for the radius of the circle.

Q4. Define a class called fruit with the following attributes:
1. Name of the fruit.
2. Single fruit or bunch fruit.
3. Price.

Define a suitable constructor and displayFruit() method that displays values of all the attributes. Write a program that creates 2 objects of fruit class and display their attributes.

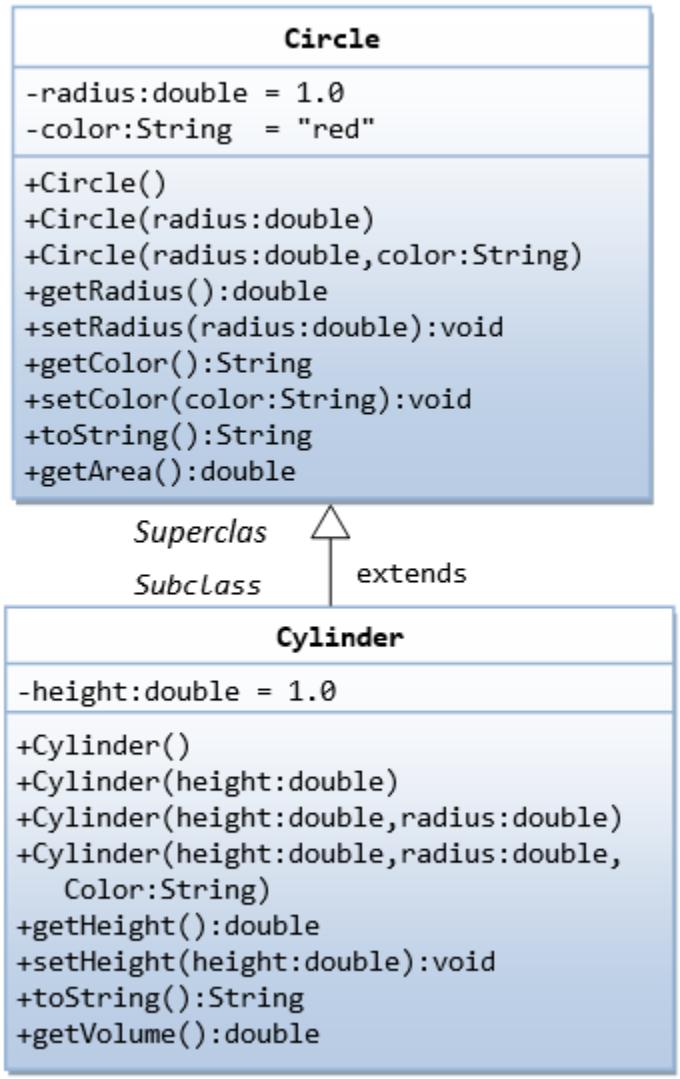
Q5. Implement a superclass **Person**. Make two classes, **Student** and **Instructor**, inherit from Person. A person has a name and a year of birth. A student has a major, and an instructor has a salary. Write the class definitions, the constructors, and the methods **toString** for all classes. Supply a test program that tests these classes and methods.

Q6. Define a class Employee having private members – id, name, department, salary. Define default and parameterized constructors. Create a subclass called "Manager" with private member bonus. Define methods accept and display in both the classes. Create n objects of the Manager class and display the details of the manager having the maximum total salary (salary+bonus)

Q7. Write a Java program to create a super class **Vehicle** having members Company and price. Derive 2 different classes LightMotorVehicle (members – mileage) and HeavyMotorVehicle (members – capacity-in-tons). Accept the information for n vehicles and display the information in appropriate form. While taking data, ask the user about the type of vehicle first.

Q8. A bank maintains two kinds of accounts - Savings Account and Current Account.The savings account provides compound interest, deposit and withdrawal facilities. The current account only provides deposit and withdrawal facilities. Current account holders should also maintain a minimum balance. If balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account number, and type of account. From this derive the classes Curr-acct and Sav-acct. Include the necessary methods in order to achieve the following tasks.
a. Accept deposit from a customer and update the balance.
b. Display the balance.
c. Compute interest and add to balance.
d. Permit withdrawal and update the balance (Check for the minimum balance, impose penalty if necessary).

9. Write a program to demonstrate given fig.



10. Write a program for user defined Exception that checks the external and internal marks if the internal marks is greater than 40 it raise the exception internal mark is exceed, if the external mark is greater than 60 exception is raised and display the message the external marks is exceed, create the above exception and use it in your program.

11. Write a base class called shape and derive the 2 classes as Circle & Rectangle and implement the appropriate functions in both classes.

13. Write a program for exception handling. Which will take two command line arguments from the user if the user does not provide that arguments it should throw exception.

14. Create a class Student with attributes roll no, name, age and course. Initialize values through parameterized constructor. If age of student is not in between 15 and 21 then generate user-defined exception "AgeNotWithinRangeException". If name contains numbers or special symbols raise exception" NameNotValidException ".

15. Write a program to demonstrate the finally block.

16. Write a program in java. A class Teacher contains two fields Name and Qualification. Extends 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.

Q1 Write a menu based program to store record of student in your database.
create four functions
insert()
update ()
delete()
read()
call them as per user requirement.

Q2 create a class employee and store record of 20 employees using
treemap where empid is the key and employee object is the value for map.

1. Reading input from a file and displaying it in the console

2. Reading input from two files and storing it in a third file

3. Write a program to define a class employee having fields like empno ,name ,age,dept ,salary .

   a) Define constructors and destructors.
   b) Define get method to invoke values from user
   c) Store the complete record of employee into file
   d) Read all records of employee from file and display them on screen

4. Write a java program to throw a exception (checked) for a student details

   a. If name is a number, an exception must be thrown.

   b. If an roll number is greater than 50, an age exception must be thrown.

   c. Or else an object must be created for the entered employee details

5. Write a java program to perform the following operations based upon the choice entered by the user.1. Reading input from a file(f1) and displaying it in the console2.Read the input from the program itself and write to the file(f2)3.Concatenation of files(f1 and f2)

6. TO f3 FILE

7. Appending information to file f3

8. Perform word count on f3

9. Write a program to ask name of person and display name character by character in each line.

10. Write a program to ask two password from user (in string) and compare them .If both are equal print “access allowed “ otherwise ask string maximum of three times .

11. write a program to create a class for employee having fields like name , age , salary. Ask 5 employees records from user in an array and display them in ascending order .

12. Write a program to arrange a set of integer numbers in a ascending order where input will be taken through command line argument

13. . Write a program to calculate the cube & Square using pameterised constructor using BufferedReader class object.

14. Write a java class which consists of 5 integer data array .Overload constructor (Default & normal) to initialize those integer data members.Provide a method which sorts those integer data members using bubble sort.

15. Write a program to maintain the office database using single inheritance. Superclassis Employee that contain the information as follows- Emp_code, Emp_name,Address, Ph_no, Da-10%, Hra-20%.

    Create three subclass of Manager, Typist, officer each class having their own basic pay & da,hra remain same.

    Create a menu driven application in main and depending on choice ask details and printy them on screen .

16. Define an Exception NoMatchFoundException that is thrown when Kolkata isnot found from the following set of strings.city name ={ Kolkata, Chennai, Mumbai, Delhi, Bangalore, Amedabad}

17. Write a class Student, store it in package stud. Write a class Batch with information about subject, faculty, and timing. Store it in package bat. Use the class Batch to set informationin the Student class

18. Write a program that defines an array of 5 elements. This array is later initialized tocontain 10 elements. The program should throw a customized exception when the arrayoverflows

19. Write a program to display two strings moving in the opposite direction.
20. List all '.java' files present in a directory.
21. Create three classes message1 ,message2 and message3 and call their methods one after another and display message .In above case set priority level of all object and run them according to priority. Create another class message4 and make sure it should not be called before previously define class. Hint use join
22. Write a program that contains method even and another method odd , call both of the methods in main and print their values after 2000 millisecond and make sure that another method should not be called before the task of one is complete .
23. write a program for thread synchronization. create an account class with required attributes and method as common resource. create first thread to deposit money, second thread to withdraw money and third thread to check balance.