

# Mitte Mai Milla Denge

Page No.	8
Date	

## Assignment 1-1.

Q.1. Write C++ source code to display the given number; positive or negative.

```
→ #include <iostream>
using namespace std;
int main () {
    int num = 12;
    if (num >= 0) {
        if (num == 0)
            cout << "Zero";
        else
            cout << "The num is positive";
    } else
        cout << "The numbers is negative";
    return 0;
}
```

Q.2. Write C++ code to check given year is leap year or not

```
→ #include <iostream>
using namespace std;
int main () {
    int year;
    cout << "Enter a year";
    cin >> year;
    if (year % 4 == 0) {
        cout << year << " is a leap year";
    } else
        cout << year << " is not a leap year";
    return 0;
}
```

# Mitte Mai Milla Denge

Page No.	
Date	

## Assignment - 2

Q.1. Write a Java class for example.

```
import java.lang.*;  
public class Employee  
{  
    String name, address;  
    int empid, age;  
    long contractno;  
  
    int withdraw(int salary);  
  
    return salary;  
  
    public static void main (String a [] )  
    {  
        Employee emp = new Employee ();  
        System.out.println ("Employee Salary Temp.  
            withdraw (2000));  
    }  
}
```

# Mitte Mai Milla Denge

## Assignment - 3

Page No.	
Date	

Q.1. Write a difference between C & C++

C

C++

- (1) C was developed by Dennis Ritchie between the year 1969 & 1973 at AT&T Bell Lab. (1) C++ was developed by Bjarne Stroustrup in 1979.
- (2) C does not support polymorphism, encapsulation & inheritance which means that C does not support OOPS. (2) C++ supports polymorphism, encapsulation inheritance because it supports OOPS.
- (3) C is mostly subset of C++.
- (4) Data is hidden by information hiding.
- (5) Namespace Features are not present inside C language. (5) Namespace is used by C++.
- (6) C structures don't have access modifiers.
- (6) C++ structures have access modifiers.

# Mitte Mai Milla Denge

Page No.	
Date	)

Q. 2. Advantages of Object Oriented Programming language are as follows:-

- (1) class :- Class are used to create the blueprint or the template for defining an object. It contains the properties & characteristic regarding the object.

Ex. Car

1

travelling

speed, average,

transportation

Color, etc.

- (2) Inheritance :- It is the ability of an object to acquire the abilities of other objects. It enhances code reusability.

- (3) Polymorphism :- It is one of the feature of object oriented programming language which means one task can be performed in many different forms.

- (4) Abstraction :- It is hiding of internal running part of the data and only presenting the essential information.

- (5) Encapsulation :- It binds the data together & keep it secured from outer vulnerabilities. Access modifiers / Specifiers like public, private, protected, default are used.

# Mitte Mai Milla Denge

Page No. \_\_\_\_\_  
Date \_\_\_\_\_

(e) Association can represent the relation between various entities like 1-1, 1-Many, Many-1, Many-Many are types of Relation.

Associations hold many types of constraints which are called cardinality constraints. These constraints define the maximum and minimum number of occurrences of one entity in relation to another entity.

Cardinality constraint can be defined as follows:

- 1-1: One entity from one table can be associated with one entity from another table.
- 1-M: One entity from one table can be associated with many entities from another table.
- Many-1: Many entities from one table can be associated with one entity from another table.
- Many-M: Many entities from one table can be associated with many entities from another table.

Relationships can be classified into two types:  
1. Functional relationship: A relationship where one value of one attribute can be determined by the value of another attribute. Functional dependency is represented by the symbol  $\rightarrow$ .

2. Non-functional relationship: A relationship where one value of one attribute cannot be determined by the value of another attribute. Non-functional dependency is represented by the symbol  $\rightarrow\!\!\!\rightarrow$ .

Relationships can also be classified into two types:  
1. Deterministic relationship: A relationship where one value of one attribute can be determined by the value of another attribute. Deterministic dependency is represented by the symbol  $\rightarrow$ .

2. Non-deterministic relationship: A relationship where one value of one attribute cannot be determined by the value of another attribute. Non-deterministic dependency is represented by the symbol  $\rightarrow\!\!\!\rightarrow$ .

Page No.	
Date	

## Assignment - 4

Q.2. List and explain features of JAVA.

→ (1) Simple :- It means Java language is very easy to learn.

(2) Secure :- It means that Java has advanced authentication and access control functionalities which makes the web application secure.

(3) Portable :- It means that Java code can execute on all major platforms. You have to compile your Java source code file into a bytecode file.

(4) Robust :- It means that Java has better memory management and exception handling features which can handle the errors during execution and manage the incorrect input of data.

(5) Object-Oriented :- It is model of programming which aims to work with the real-world entities in programming. Such as inheritance, data hiding, polymorphism etc.

(6) Multithreading :- Java supports multithreaded programming. Due to this, multithreaded programs written in Java can perform several tasks at the same time.

Page No.	
Date	

Neutral :-

(7) Architecture - It means that when you compile a Java code it produces a bytecode which can run on many platforms with the help of Java Runtime Environment (JRE) and Java Virtual Machine (JVM).

(8) Interpreted :- Java is a ~~not~~ cross-platform program because it creates an intermediate representation named Java bytecodes.

(9) High-Performance :- It means that java uses the Just-in-Time (JIT) compiler to enable high performance. JIT is used to change the instructions in bytecodes.

(10) Distributed :- By using Java, one can create distributed applications that can access files by calling the program from any machine on the internet using RMI and EJB.

(11) Dynamic :- Java can compile and load its class dynamically on demand. It can compile and load a specifically required portion of code instantly rather than compiling the whole code.

(12) Platform Independent :- Java code can be executed on multiple platforms like windows, Linux, mac OS etc. Java code is compiled by the compiler and converted into byte code.

Ans  
Date

Page No.	
Date	

## Assignment - 5

Float

Double

- |  |   |
|--|---|
| (1) The float data type is a 32 bit single precision datatype. | The double data type is a 64 bit double precision datatype. |
| (2) It takes up 4 bytes  | It takes up 8 bytes   |
| (3) It has low accuracy  | It has high accuracy  |
| (4) It is an integer data type but with decimal                | It is also integer data type but with decimals              |
| (5) For eg. float a = 3.14f                                    | For eg. double a = 3.1489                                   |

Int

Long

- |   |  |
|---|--|
| (1) It is a datatype they has 32-bits       | It is a datatype they has 64-bits                      |
| (2) It takes up 4 bytes                     | It takes up 8 bytes                                    |
| (3) It is also used as a keyword to declare | It can also be used as a keyword to declare a variable |
| (4) It has a wrapper class Integer & Long   |  |

# Mitte Mai Milla Denge

# Mitte Mai Milla Denge

Page No.	
Date	

## Assignment :- 6

Q.1. Write a java program only even number  
from 1 to 100

Public class test

{

    public static void main (String args [] )

{

        int number = 100;

        System.out.print ("List of even number : " )

        for (int i=1; i <= num. number);

            for (int i=1; i <= number; i++)

            {

                if ((i % 2 == 0))

            {

                System.out.println (i + " " );

            }

            }

    Public class Test {

    public static void main (String args [] ) {

        int sum = 0;

        for (int i=1 ; i <= 10 ; i++) {

            sum = sum + i;

        }

        System.out.println (sum);

    }

    }

# Mitte Mai Milla Denge

Page No.	
Date	

## Assignment 9-7

(Q1). Write Multiplication table in java program

Public class Test {

    Public static void main (String args[])  
    {

        int num = 5;  
        for (int i=1, i<=10 ; ++i)  
    {

            System.out.println (num\*i);

    }

Output

5

10

15

20

25

30

35

40

45

50

Ans  
BSP

# Mitte Mai Milla Denge

Page No.	
Date	

## Assignment :- 8

Ques. Write a program to print the following pattern.

Class Main {

{ public static void main (String args[]) {

int n = 5; for (int i = 1; i <= n; i++) {

for (int j = 1; j <= i; j++) {

System.out.print(j + " ");

} System.out.println();

Output

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

Ans  
1 2 3 4 5

# Mitte Mai Milla Denge

Page No.	
Date	

## Q1 - Assignment 8-19

Q1. WAP. To print even number between 2D Array

```
public class Even_numbers {
```

```
    public static void main (String args) {  
        int [][] numbers = {{1, 2, 4, 3},  
                            {6, 5, 3, 2}, {7, 8, 14, 5}, {}};
```

```
        for (int i=0; i<numbers.length; i++) {  
            for (int j=0; j<numbers[i].length; j++) {  
                if ((numbers [i]) [j] % 2 == 0) {  
                    System.out.println (numbers [i] [j]);  
                }  
            }  
        }  
    }  
}
```

(1) 1, 2, 4, 3, 6, 5, 3, 2, 7, 8, 14, 5

(2) 2, 4, 6, 8, 14

(3) 2, 4, 6, 8, 14

(4) 2, 4, 6, 8, 14

(5) 2, 4, 6, 8, 14

(6) 2, 4, 6, 8, 14

(7) 2, 4, 6, 8, 14

(8) 2, 4, 6, 8, 14

(9) 2, 4, 6, 8, 14

(10) 2, 4, 6, 8, 14

(11) 2, 4, 6, 8, 14

(12) 2, 4, 6, 8, 14

(13) 2, 4, 6, 8, 14

(14) 2, 4, 6, 8, 14

(15) 2, 4, 6, 8, 14

(16) 2, 4, 6, 8, 14

(17) 2, 4, 6, 8, 14

(18) 2, 4, 6, 8, 14

(19) 2, 4, 6, 8, 14

(20) 2, 4, 6, 8, 14

Mitte Mai Milla Denge

# Mitte Mai Milla Denge

Page No.	
Date	

## Assignment - 10

Q.2: Write a java code to display square of given matrix.

```
→ public class MatrixSquare {
    public static void main (String [] args) {
        int [][] matrix = {{2,3,4},{5,6,7},{8,9,10}};
        for (int i=0; i<matrix.length; i++) {
            for (int j=0; j<matrix.length; j++) {
                System.out.print (matrix [i] [j] +
                    matrix [i] [j] + " ");
            }
            System.out.println ();
        }
    }
}
```

## OUTPUT

4 9 16

25 16 36

1 4 9

# Mitte Mai Milla Denge

Page No. \_\_\_\_\_  
Date \_\_\_\_\_

## Assignment :- 11

Q.1. WPA) To print hotel menu card.

→ Class Hotel

String orderResult = "";

Void paneerTikka (int quantity)

System.out.println ("Your paneer tikka  
is ready with " + quantity);

String vegKolhapuri (boolean spicy)

orderResult = "Your veg kolhapuri  
with spicy " + spicy;  
return orderResult;

String vegBiryani (boolean Spicy)

orderResult = " your veg biryani with  
spicy " + Spicy;  
return orderResult;

String nonVegBiryani (String type,  
boolean Spicy, int quantity)

orderResult = " your nonveg biryani  
with Spicy " + Spicy + " Type : " + type + " quantity  
" + quantity;  
return orderResult;

# Mitte Mai Milla Denge

Page No.	
Date	

public void main (String [] args) {

    Hotel hotel = new Hotel ();  
    hotel.pannerTikka (2);

System.out.println (hotel.vegkolhapuri  
(true));

System.out.println (hotel.vegBiryani  
(true));

System.out.println (hotel.nonVegBiryani  
( "chicken" , true ));

## OUTPUT

Your panner tikka is ready with 2

Your veg kolhapuri with spicy true

Your veg biryani with spicy true

Your non veg biryani with spicy true

Type:- chicken quantity: 2

int print menu () = this method

found + found + print + type + quantity + true + false

new + menu + print + quantity + true + false

new + menu + print + quantity + true + false

new + menu + print + quantity + true + false

new + menu + print + quantity + true + false

new + menu + print + quantity + true + false

new + menu + print + quantity + true + false

new + menu + print + quantity + true + false

new + menu + print + quantity + true + false

new + menu + print + quantity + true + false

# Mitte Mai Milla Denge

Page No.	
Date	

## Assignment - 12

Q.1 Write a program to make class hotel

class Hotel

{

String OrderResult = " /";

Void PannerTikka (int quality)

{

System.out.println (" your panner tikka is

already with it " + quantity);

String VegKothapuri (boolean Spicy)

{

OrderResult = " your veg kothapuri with

Spicy " + Spicy;

return OrderResult;

{

String VegBiryani ( boolean Spicy)

{

OrderResult = " your veg biryani with

Spicy " + Spicy;

return OrderResult;

{

String nonVegBiryani ( String type, boolean

Spicy, int quantity)

{

OrderResult = " your non-veg biryani is

ready with Spicy " +

Spicy + " Type : " + type + " Quantity : "

+ quantity;

return OrderResult;

# Mitte Mai Milla Denge

Page No.			
Date			

```
public static void main (String [] args)
```

```
Hotel hotel = new Hotel();  
hotel paneerTikka(2);
```

```
System.out.println(hotel.VegBiryani(true));
```

```
System.out.println("Hotel NonVegBiryani(true)");
```

~~2010~~ 2012; ~~2010~~ 2012

Final results of the experiment will be

*Alouatta* *leucophaea* *leucophaea* *leucophaea*

1970-1972

Digitized by srujanika@gmail.com

1. *Agave attenuata* Benth. ex Engelm. var. *attenuata* Benth.

*of importance to many students and others.*

卷之三

WANT TO GET SOME PRACTICE ON THE NEW MATERIAL

10. *Leucosia* (Leucosia) *antennata* (Fabricius)

19. *Leucania* *luteola* (Hufnagel) *luteola* Hufnagel

4.300.000 1.000.000

10. The following table shows the number of hours worked by each employee in a company.

Page 1 of 1

Mitte Mai Milla Denge