**METHOD OVERLOADING**

public static void Main(string[] args)

{

Program ob = new Program();

Double sum3 = ob.Add(5.9, 6.8, 7.9);

Int sum2 = ob.Add(4,6,3);

Console.WriteLine("Sum of three integer : " + sum2);

Console.WriteLine("Sum of three double : " + sum3);

Console.Read( );

}

public int Add ( int a , int b , int c)

{

int sum = a + b + c ;

return sum ;

}

public double Add (double a , double b , double c)

{

double sum = a + b + c ;

return sum ;

}

**MULTI THREADING**

static void Main(string [] args)

{

Thread t = new Thread(Worker);

t.Start();

for (int i = 0; i < 10; i++)

{

Console.WriteLine("Main thread doing some work");

Thread.Sleep(100);

}

t.Join();

Console.WriteLine("Done");

}

static void Worker()

{

for (int i = 0; i < 10; i++)

{

Console.WriteLine("Worker thread doing no work");

Thread.Sleep(100);

}

}

**LISTDEMO**

static void Main(string[ ] args)

{ List<int> mylist = new List<int>();

for (int j = 5; j < 10; j++)

{

myList.Add( j \* 3 );

}

foreach ( int item in mylist)

{

Console.WriteLine(item);

}

Console.ReadLine();

}

**METHOD OVERIDING**

{

class baseClass

public virtual void Greetings()

{

Console.WriteLine("baseClass Saying Hello!");

}

}

class subClass : baseClass

{

public override void Greetings()

{

base.Greetings();

Console.WriteLine("subClass Saying Hello!");

}

}

class Program

{

static void Main(string[] args)

{

baseClass obj1 = new subClass();

obj1.Greetings();

Console.ReadLine();

}

}

}