**Delegate demo 1**namespace delegatedemo\_pre

{

delegate void mydelegate(); //delegate declaration

class Accenture

{

public void show()

{

Console.WriteLine("welcome to accenture");

}

}

internal class Program

{

static void Main(string[] args)

{

Accenture obj=new Accenture();

// mydelegate d = new mydelegate(obj.show);

mydelegate d = obj.show;

d(); //show() called

Console.ReadKey();

}

}

}

**Delegate demo 2:**

//unicast delegate

namespace delegatedemo2\_pre

{

delegate int mydelegate(int a,int b); //delegate declaration

class Accenture

{

public int addition(int a,int b)

{

return a + b;

}

public int subtraction(int a, int b)

{

return a - b;

}

}

internal class Program

{

static void Main(string[] args)

{

Accenture obj = new Accenture();

mydelegate d = obj.addition;

int addres=d(70,30); //addition called

Console.WriteLine("Addition:"+addres);

mydelegate d2 =obj.subtraction;

int subres = d2(70, 30); //subtraction called

Console.WriteLine("Subtraction:" + subres);

Console.ReadKey();

}

}

}

**Delegate demo 3:**

//multicast delegate

namespace delegatedemo3\_pre

{

delegate void mydelegate(int a, int b);

class Accenture

{

public void addition(int a, int b)

{

int res= a + b;

Console.WriteLine("Addition:" + res);

}

public void subtraction(int a, int b)

{

int res= a - b;

Console.WriteLine("subtraction:" + res);

}

}

internal class Program

{

static void Main(string[] args)

{

Accenture obj = new Accenture();

mydelegate d = obj.addition;

d += obj.subtraction;

d(70, 30);

d -= obj.addition;

Console.WriteLine("\n----------after remove----------");

d(80, 30);

Console.ReadKey();

}

}

}

**Delegate demo 4**

//using anonymous method

namespace delegatedemo4\_pre

{

delegate void mydelegate();

internal class Program

{

static void Main(string[] args)

{

mydelegate d = delegate

{

Console.WriteLine("this is anonymous method");

};

d();

Console.ReadKey();

}

}

}

**Delegatedemo 5**

//using anonymous method 2

namespace delegatedemo5\_pre

{

delegate void mydelegate(int a,int b);

internal class Program

{

static void Main(string[] args)

{

mydelegate d = delegate(int a,int b)

{

int res = a + b;

Console.WriteLine("Addition:"+res);

};

d(10,20);

Console.ReadKey();

}

}

}

**Delegate demo 6**

//using anonymous method 3

namespace delegatedemo6\_pre

{

delegate int mydelegate(int a, int b);

internal class Program

{

static void Main(string[] args)

{

mydelegate d = delegate (int a, int b)

{

return a + b;

};

int res=d(10, 20);

Console.WriteLine("Addition:"+res);

Console.ReadKey();

}

}

}