```
Delegate : হল একটি মেখড এর pointer। যা একটি মেখড কে refer করে।
এটি জাস্ট মেখড কে কল করে।
Delegate হল একটি POINTER যা হল METHOD বা FUNCTION কে POINT করে।
If i say In Real life example , Delegate is Like a Representative,To Communicate the two countries use a delegate .
```

#### Rules:

- Delegate signature As Like as Method signature.
- Can Create obj as like as class.

## **Delegate Example:**

#### Example:1

```
delegate double AreaDelegate(int radius);
static double getArea(int r)
{
    return Math.PI * r * r;
};
static void Main
{
    AreaDelegate ad = new AreaDelegate(getArea);
    console.writeline(ad(5));
    // Or use Invoke
    console.writeline(ad.Invoke(5));
}
```

### Example: 2

Now we will create new class which call two method through Delegate.

```
Class Rectangle
{
     // Method 1
     Public Static void GetArea( int h, int w )
     {
          Console.writeline ("Area:{0}",h*w)
     }
```

```
// Method 2
        Public Static void GetPerimeter( int h, int w)
         Console.writeline ("Area:{0}",2*(h*w))
       }
}
// Now Create a Delegate
delegate void RectDelegate(int h,int w);
// Now call those two method through delegate
// As this method is static type so we don't need create a obj for delegate.
Static void Main
    RectDelegate rd = new RectDelegate(Rectangle.GetArea);
    // if we want to use other method we have to use compound operators ,like -
    rd+= Rectangle.GetPerimeter;
    rd(5,7);
   // If we want to call only one method, then use - = operator
   // which method we don't not need to use , we can use remove
    rd - = Rectangle.GetArea;
    rd(6,9);
```

// here we will find the GetPerimeter result.

}

### Example 3:

# **Anonymous Delegate**

As before we call method into delegate, now same way we can create a method into delegate. For this, we will create 2 delegate.

```
delegate double Adelegate ( int x );

delegate double Bdelegate ( int x );

Static void Main
{
    Adelegate ad = delegate( int a ) { return a * a };
    Bdelegate area = delegate( int a ) { return Math.PI * a * a };

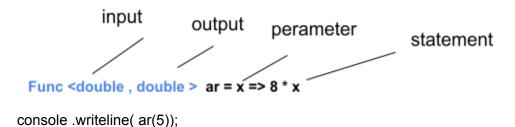
    console.writeline( ad(10));
    console.writeline( area (7));
}
```

### Example 4:

## **Built in Delegate**

That same work will be done by the built delegate function. Some of Built in Delegate Functions are:

- Func
- Action
- Predicate



// In the same way we can create an Area.

```
Func <double , double > area = x => Math.Pl * x * x; console .writeline( area (5));
```

## **Example For Action:**

• Only take input parameters, No Return Type, It's Like a void.

```
Action < string> myAction = x=> console.writeline(y);
myAction ("Hello world");
```

### **Example For Predicate:**

- Take Input parameter and return Boolean type.
- If you want to compare u can use predicate

```
Predicate <int> pred = pd = > pd>=10
Console.writeline ( pd(15));
```