

Accessing the Class Members

In this lesson, we will explore how to access the members of a class through its object.

WE'LL COVER THE FOLLOWING



- Creating an Object of the `VendingMachine` Class
- The Dot Operator

Creating an Object of the `VendingMachine` Class

In the previous lesson, we implemented a basic version of the `VendingMachine` class. In this lesson, we will move a step further and learn how to access the implemented class members in another class through its object. For achieving this, let's look at our implementation of the `VendingMachine` class and create an object of it.

```
class VendingMachine
{
    //Class fields
    public int Count = 30;
    public int Capacity = 100;
    public int MoneyCollected = 300;
    public string Manufacturer = "Vendy Inc.";

    //Class Methods
    public void Display()
    {
        Console.WriteLine("I am the Display method for displaying the menu!");
    }
    public void DispenseProducts()
    {
        Console.WriteLine("I am the DispenseProducts method for dispensing the products!");
    }
    public void Refill()
    {
        Console.WriteLine("I am the Refill method for refilling the products!");
    }
}
```



```
class Demo //Class containing the Main() method
```

```

class Demo //class containing the Main() method
{
    public static void Main(string[] args)
    {
        var vendingMachine = new VendingMachine(); //Object creation
        //Calling the members of the VendingMachine class
        Console.WriteLine("The capacity of the machine is: {0}", vendingMachine.Capacity);
        vendingMachine.Display();
    }
}

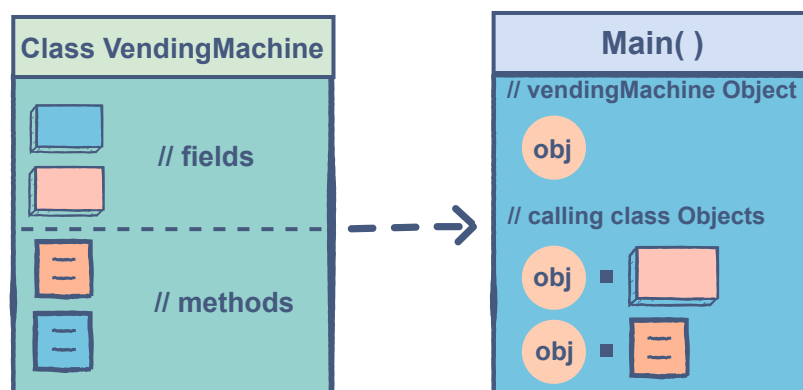
```



In the above code, you can ignore the **public** keyword used with every member of the **VendingMachine** class. We will discuss the accessibility levels in detail in the next lesson. For now, let's just focus on how to access a class' members from another class using an object.

The Dot Operator

In the above code widget, at **lines 32 and 33**, we have accessed the **capacity** field and the **Display()** method of the **VendingMachine** class inside the **Demo** class using the **vendingMachine** object.



Use of dot operator

The members of a class can be accessed using the object name followed by the *dot operator* **'.'** and the name of the respective class member. In general, we can write it down as:

```

classObject.memberVariable; //accessing a field
classObject.MemberMethod(); //accessing a method

```

Now that we have learned how to access the class members using an object of that class, in the next lesson, we will discuss an important aspect of OOP known as access modifiers.