# Declaration and Implementation

In this lesson, you will learn about the declaration and implementation details of a class.

WE'LL COVER THE FOLLOWING


- Declaration
- Creating a Class Object
- Implementation of the VendingMachine Class
- Naming Conventions


## Declaration

In C#, we declare classes in the following way:

```
class ClassName // Class name
{

    /* All member variables
    and methods */

}
```

The keyword `class` tells the compiler that we are creating our custom class. All the members of the class will be defined within the class scope, i.e., inside the curly brackets, `{...}`.

## Creating a Class Object

The name of the class, `ClassName`, will be used to create an instance/object of the class in our `Main()` method. We can create an object of a class by using the `new` keyword:

```
class ClassName  //Class name
{
  ...
```

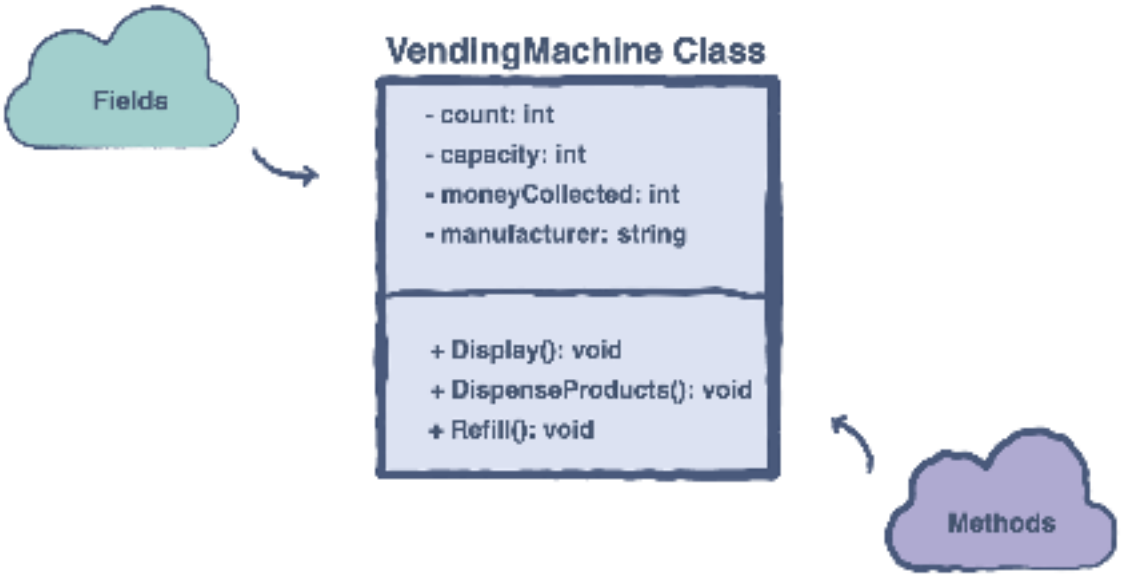
```
}

class Program //Class having the Main() method

{
    public static void Main(string[] args) //Main method
    {
        ClassName objectOneName = new ClassName(); //ClassName first object
        var objectTwoName = new ClassName(); //ClassName second object
    }
}
```

In the above snippet, we can see that there are two ways we can create an object of a class. We can use the name of the class on both sides of the assignment operator or we can use the `var` keyword on the left side and the name of the class on the other. When using `var`, the compiler automatically infers the type of the object being created from the right side of the assignment operator.

> Whenever the compiler comes across the keyword `new`, it realizes that an object is being created and allocates a separate memory location for the object being created.

## Implementation of the VendingMachine Class #

Let's implement the `VendingMachine` class illustrated below:




Class Diagram

```
class VendingMachine //Class name
{
```

```
{
    //Class fields
    int count;
    int capacity;
    int moneyCollected;
    string manufacturer;

    //Class Methods
    void Display()
    {
        //Method definition goes here
    }
    void DispenseProducts()
    {
        //Method definition goes here
    }
    void Refill()
    {
        //Method definition goes here
    }

}
```

Now that we have learned how to declare and implement a class in C#, let's briefly discuss the naming conventions we should follow when working with classes.

## Naming Conventions #

- The name of a class should preferably be a *noun* so that it represents the actual purpose of creating that class, e.g., `VendingMachine`, `Calculator`, etc.
- While naming a *class*, we should use the `PascalCase`.
- While naming the *methods* in a class we should use the `PascalCase`.

The naming conventions of fields will be discussed in the upcoming access modifiers lesson.

---

We've seen the structure of a class and the basic skeleton of the `VendingMachine` class. In the next lesson, we will build upon this by creating an object of the vending machine class and accessing the class members through it.