C#Corner

ecome a member   Login

Post   Ask Question

Real Life Examples of Object Oriented Programming

Rupesh Kahane   Updated date Jun 06, 2014   105.2k   0   2

Class - Consider a Harmonium
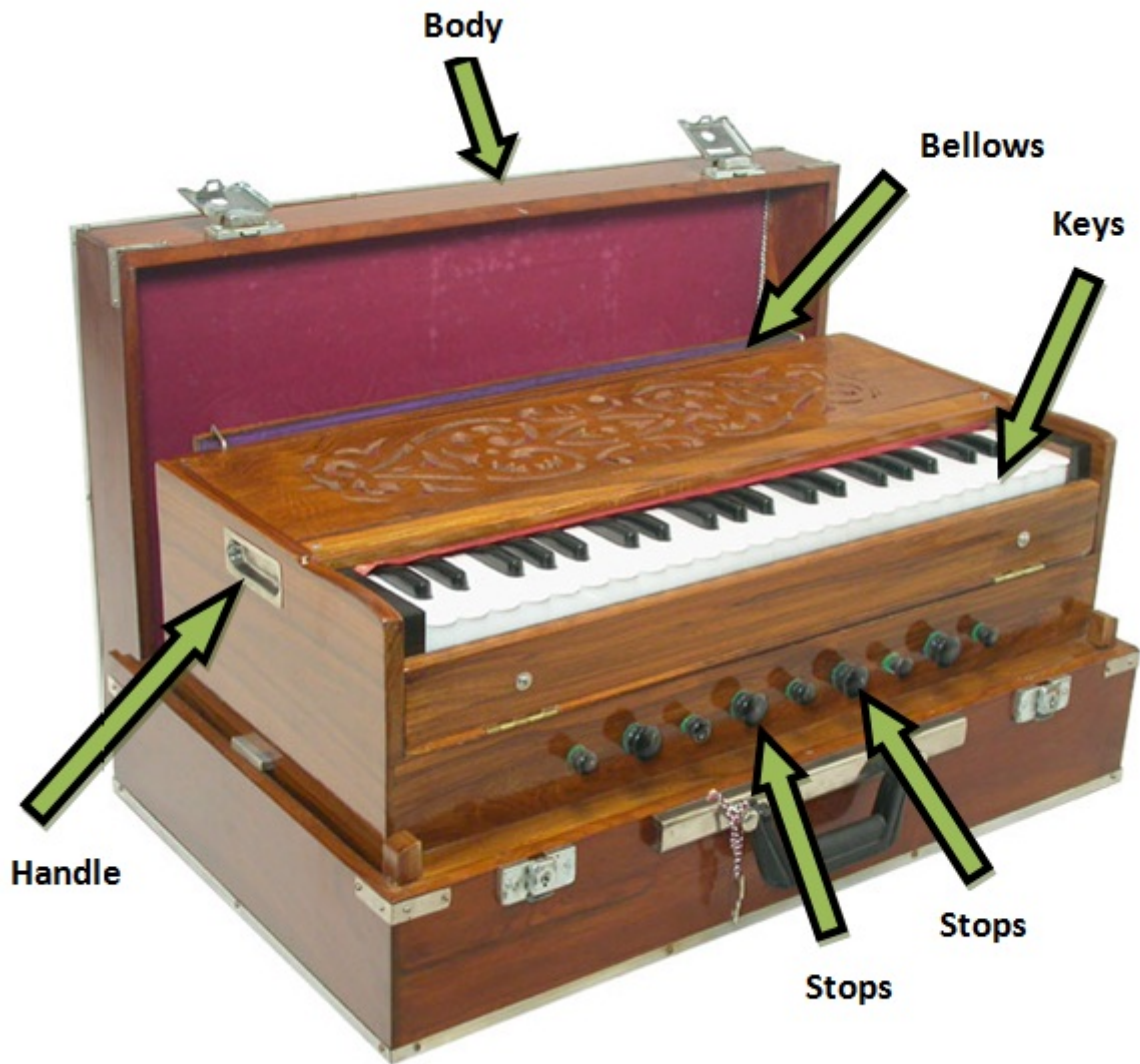
```
Class Harmonium
{
    public string color;
    public int total_keys;
    public string bellows;
    public void Sound();
}
```

It is a blue print of newly created things that contains variables for storing data (characteristics) & functions (behavior) which is applied on that data.

Object

- It is a instance of class
- They are inherit from System.Object
- Object is an entity which has a well-defined structure & behavior.

Object has characteristics as :

1. State
2. Behavior

3. Identity
4. Responsibility

Example - a car, a person, hard disk, pen, bank account

1. State of an object

State of an object includes the current values of all its attributes.

An attribute can be static (values will not change) or dynamic (values will be change)

Consider a Employee has following attributes.

Static

1. Empid
2. Name static
3. Gender

Dynamic

1. Age
2. Address
3. Phone / mobile no dynamic
4. Salary
5. Education

2. Behavior of an object

Behavior is how an object acts & reacts, in terms of its state change and message passing.

Employee is an object ===> Calculate_salary, Emp_details

- The change in state of an object reflects in its behavior.
- In above example Calculate_Salary and Emp_Details are behaviors of an employee

3. Identity of an object

- Every object has its own identity.
- Property of an object which distinguishes it from all other objects is an Identity.
- Single or group of attributes can be identity of an object

Employee - Empid, Name, Gender, Mobile_No

4. Responsibility of an object

Responsibility of an object is the role it serves within the system.

Example - responsibility of an employee is to carry out the work given & get the salary

Abstraction

Abstraction

Internal Structure of Harmonium

A Harmonium consists of many things such as Bellows, Keys, Stops, Body and Handle. The bellows are the pumps which force the air through the instrument. Keys are the small wooden controls that the performer fingers to play the music. Body is cover of wooden which covers the workings of the harmonium. Stops are a series of valves which control the way that air flows in the instrument. The handles allow for easy transport of the harmonium.

Working of Harmonium: The sound of a harmonium is produced by the vibration of reeds, normally made from brass, caused by the release of the flow over them of escaping air when the bellow is pumped.

We just need to know how to operate the Harmonium by using these keys, bellows. Internal details are invisible. Here harmonium is an object that is designed to hide the complexity. To produce sound by using internal working are in terms of harmonium application is called abstraction i.e. hiding the implementation details

Encapsulation

- A complete harmonium is an example of Encapsulation i.e. class.
- Binding data & functions together is called Encapsulation. It keeps safe from outside interface.

Inheritance

Consider case of old harmonium, it has a less functionality. Now someone came & he needs a harmonium with new functionality recording, bass, volume control. In this case the by using adding new feature to that harmonium inheritance is achieved.

- Inheritance is the object oriented concept which is used for reusability.

Polymorphism

- Poly means many & morph means forms.
- Ability to take more than one form is called polymorphism.
- Binding is the process of associating a function call to an object.
- According to above point polymorphism is divided into two types.

1. Compile time polymorphism (binding occurs at compile time)
2. Run time polymorphism (binding occurs at run time)

Overloading

Consider a musical instrument like Harmonium, Piano, Drum kit which produces sound. For Harmonium we need to press some keys and buttons, for Drum sound is produced either directly with players hand or by using drum stick.

In above example function name is Sound; it takes parameter list like keys, button or players hand, drum stick.

Overloading is the mechanism having same method name with different parameter list:

- Scope - in same class
- Signature - different for each method overloaded
- Return type & access modifiers does not matter.

Overriding

- Scope - in different class i.e. can be done in derived class.
- Signature -method name is name with same parameter list as well as return type.
- Return type must be same.
- Modifiers new, virtual, static, abstract cannot modify method in derived class.

Summary

I hope this article is useful for beginners. If you have any query then contact me.

Real Life Examples of Object Oriented Programming

Next Recommended Reading
OOPS Concept With Real Time Example

Rupesh Kahane *TOP 500*

Currently working as a Senior Software Engineer in MNC company. Having working knowledge of technologies like ASP.NET, MVC 5.0, C#, Angular 6, TypeScript, Azure, Kendo UI, KnockOut.Js, JavaScript, SQL Server 2012, Enti... Read more

120   **3.1m**   **5**

LATEST BLOGS

How To Copy Image From One List To Another Using Flow