**Introduction to C# Exercise II**

**Q.1 -** What is the difference between abstract class and interface ? write a sample program to explain ?

**Solution:-**

**Abstract class:-**

1. It is the process to hide the internal details and showing only the functionality.
2. The **abstract modifier** indicates the incomplete implementation. The keyword **abstract** is used before the class or method to declare the class or method as abstract.
3. Generally, we use abstract class at the time of inheritance.
4. A user must use the override keyword before the method which is declared as abstract in child class, the abstract class is used to inherit in the child class.
5. It can contains constructors or destructors.
6. It can implement functions with non-Abstract methods.
7. It cannot support multiple inheritance.
8. It can’t be static.
9. It can contains constructors or destructors.

**Interface:-**

1. interfaces will contain only the declaration of the members.
2. It is used to achieve total abstraction.
3. Interfaces can’t have private members.
4. By default all the members of Interface are public and abstract.
5. The interface will always defined with the help of keyword ‘interface‘.
6. Interface cannot contain fields because they represent a particular implementation of data.
7. Multiple inheritance is possible with the help of Interfaces but not with classes.

**Q.2 -** What is method overloading and method overriding ? write a program .

**Solution:-**

**Method Overloading:-**

1. It is one of the way to implement polymorphism.
2. A user can implement function overloading by defining two or more functions in a class sharing the same name.
3. It is the ability to redefine a function in more than one form.
4. Overloaded methods are differentiated based on the number and type of the parameters passed as arguments to the methods.
5. You can not define more than one method with the same name, Order and the type of the arguments. It would be compiler error.

**Method Overriding**

1. Method Overriding is a way that allows the invoking of functions from another class in the derived class.
2. Overriding is a feature that allows a subclass or child class to provide a specific implementation of a method that is already provided by one of its super-classes or parent classes.
3. . Method overriding is one of the ways by which C# achieve Run Time Polymorphism /Dynamic Polymorphism

**Q.3 -** What would happen in case if the inherited interfaces have conflicting method names?

**Solution:- Explained in git repository**

**Q.4-** What are the different ways a method can be overloaded ? write a program .

**Solution:-**

**Different ways of doing method overloading are as follows:-**

1. The number of parameters in two methods.
2. The data types of the parameters of methods.
3. The order of the parameters of method.

These methods are explained in the git repository.

**Note-** When method signature is same and the return type is different then there would be compile time error as the return value alone is not sufficient for the compiler to figure out which function it has to call

**Q.5-** What is virtual method in c# ? Explain with program .

**Solution:-**

1. A method which is redefined in derived class is virtual method
2. A virtual method has its implementation in both base class and derived class.
3. It is used when a method's basic functionality is the same but sometimes more functionality is needed in the derived class
4. A virtual method is created in the base class that can be overriden in the derived class. We create a virtual method in the base class using the virtual keyword and that method is overriden in the derived class using the override keyword.

Program is shown in git repository.

**Q.6-** How to preventing Derived class from overriding virtual members?

**Solution:-** By making classes sealed we can prevent derive class from overriding virtual members.We can also use the sealed modifier on a method that overrides a virtual method in a base class. This enables us to allow classes to derive from our class and prevent other users that are using our classes from overriding specific virtual methods.

Program is shown in git repository.

**Q.7 -**Write a program to for parameterised constructor and call the same from derived call ?

**Solution:- Explained in git repository**

**Q.8-** Write a program to explain encapsulation and abstraction ?

**Solution:- Explained in git repository**