Lab 7 Answers

Qn 1 public class Book

{

static string title;

static string author;

static DateTime someDate;

static void Display() / Method to be called to display in main

{

title = "Computer Programming";

author = "Isaac Mpoya";

someDate = new DateTime(2020, 05, 07);

Console.Write("Title: ");

Console.WriteLine(Book.title);

Console.Write("Author: ");

Console.WriteLine(Book. Author);

Console.Write("Date: ");

Console.WriteLine(someDate);

Console.ReadLine();

}

public static int Main()

{

Display();

return 0;

}

Qn 2 public class Book

{

//Book Properties

public static string title;

static public string author;

public DateTime someDate;

}

public class Preface // Book Object

{

static void Display()

{

var myBook = new Book();

Book.title = "Computer Programming";

Book.author = "Isaac Mpoya";

myBook.someDate = new DateTime(2020, 05, 07);

Console.Write("Title: ");

Console.WriteLine(Book.title);

Console.Write("Author: ");

Console.WriteLine(Book.author);

Console.Write("Date: ");

Console.WriteLine(myBook.someDate);

Console.ReadLine();

}

public static int Main()

{

Display();

return 0;

}

Qn 3. A constructor is a special method which is invoked automatically at the time of object creation. It is used to initialize the data members of new object generally. Constructors save time by reducing the amount of code.

Qn 4. Constructor Overloading is a technique to create multiple constructors with a different set of parameters and the different number of parameters. It allows us to use a class in a different manner.

Qn 5. The use of **exceptions** allows code error management. It is better than the traditional error management technique because it groups the error type and maintains differentiation, propagates the errors up the caller stack and separates the error to be handled from the regular code. .

The main importance is that it helps in continuing the program execution even after the execution has been caught. It does not lead to terminating the code when the exception has happened.

Qn 6. **Private**: The members declared in a private class can only be accessed by the functions inside that class, or other functions that can access the private class. All members in a **Public** class are available to everyone and can be accessed by other classes too. Public members can be accessed from anywhere in the program using direct member access operator (.).

The main purpose of modifiers is to define data, information Abstraction and Hiding. Anything declared as public will be accessible in and outside the class, ie class objects can be accessed anywhere in the program. This is not applicable for Private members of a class. However, Private members can be used outside the class by integrating it with any other member.

Qn 7. Composition describes a class that references one or more objects of other classes in instance variables. This allows you to model a has-a association between objects. In our Book Class, composition can be used by describing the added Book object “Preface”. In instances where one has more than one Object, Composition also allows reuse of existing codes, designing of clean API, and change the implementation of a class used in a composition without adapting any external clients.