optimus information

Dot Net Induction Plan

Contents

Basics

C Sharp coding standards

Learn about the following:

1. Naming Conventions.
2. Code Complexity (using CCM).

Basic Syntax and constructs

In the following assignments make the methods as static so that they could be called directly from the Main method without using Object instances.

Assignment 1: IF Else Statements

Use console to read the user Preferred choice for language.

1. If language choice chosen is VB then print "VB .NET: OOP, multithreading and more!"
2. If it is C# then print "Good choice, C# is a fine language."
3. In else case print "Well...good luck with that!"

Name the class file as TestIf.

Assignment 2: Switch Statements

Do the assignment 1 with switch statement.

Name the class file as TestSwitch.

Assignment 3:Enumeration

Write an enum WeekDays and use this enum to print WeekDay Messages.

Name the class file as TestEnum.

Assignment4: (String operations)

Declare string and perform all the string operation from string class.

Name the class file as TestStrings.

Assignment 5: (String arrays)

Write a code to return an array of string.

From the main method call a function which will return the string array and then in main print the string arrays values.

Name the method as GetStringArray().

Assignment 6: Encapsulation Concepts

Please take any one real-life problem and create the object model for it e.g. Railway Reservation System. Your model should contain the following:

- The main objects.
- Their attributes.
- List of the key behaviors of the objects.
- List of public methods that would implement the given behavior. The methods and their parameters could be presented in a pseudo-code format. For example, the railway ticket method to store the passenger name may be represented as **storePassengerName(string name)**.
- Abstractions from the first iteration of object identification to derive a second level of objects, just like we discussed for the railway reservation system.

Please ensure that you use a model where there are at least five identified objects. Also please ensure that you choose a model of your own choice. I expect all trainees to come up with a model from a different domain.

OOPS concepts

Assignment 7: Code output

In the below code snippet tell the correct output and explain the concept of static.

```
class SavingsAccount {
public double currBalance;
public static double currInterestRate = 0.04;
public SavingsAccount( double balance){currBalance = balance;}
public static SetInterestRate(double newRate){ currInterestRate = newRate;}
public static double GetInterestRate(){ return currInterestRate;}
public void SetInterestRateObj(double newRate){ currInterestRate = newRate;}
public double GetInterestRateObj() { return currInterestRate;}
static void Main( string [] args)
{
SavingsAccount s1 = new SavingsAccount (50);
SavingsAccount s2 = new SavingsAccount (100);
Console .WriteLine( "Interest Rate is: {0}" ,
s1.GetInterestRateObj());
s2.SetInterestRateObj(0.08);
Console .WriteLine( "Interest Rate is: {0}" ,
s1.GetInterestRateObj());
SavingsAccount s3 = new SavingsAccount (10000.75);
```

```
Console .WriteLine( "Interest Rate is: {0}" , SavingsAccount .

GetInterestRate );

}

}
```

a) 0.04,0.04,0.08

b) 0.04,0.08,0.04

c) 0.04,0.08,0.08

d) None of above

Assignment 8- OOPS Concept

Write a class Student. Think about all the attributes and methods which this class can have.

1. All the member variables should be written as private and exposed through public properties.
2. Write methods as public to be used from another class.
3. Write a parametric constructor which takes in all the details of student and create the student object.
4. Write a random number generator to generate StudentEnrolmentNo (if don't know about Random number generator then check internet)
5. Write a function which prints all the details of student with a welcome message.
6. Write the overloaded print method which should print specific details. For example, print(int choice) should print only the age of the student. and within the method use the enum StudentAttribute with case to show which is the choice of the user.
7. Write an enum for student details/attributes and use this in the print method.

Assignment 9: Inheritance

Write a class Vehicle

1. Write Attributes as make(string),yearOfManufacture(int),model(string) and speed(float).
2. Write constructors and properties to initialize the vehicle objects
3. Write Methods as Accelerate(),Deaccelarate,Stop with return type as void.
4. Write a method isMoving(return type =boolean) to check whether vehicle is

moving or stopped.

5. Derive classes Bicycle, Bike, Car and Truck from Vehicle class.
6. Add some attributes in the derived classes which applies to specific vehicle only.
7. Think about behavior of the vehicles and implement new methods in the derived classes. Print the state of vehicle in the classes.

   Create project Vehicle in folder named as OOPS Concepts.

Assignment 10 - Access Modifiers

You don't have to write any code for this assignment. Just need to write your comments

Problem 1: Mention if the code below will run successfully or not.

```
class Student{

protected int _age;

int _rollNo;}

class CollegeStudent{

public static void Main(){

CollegeStudent stu = new CollegeStudent();

stu._age = 10;}

}
```

Give the reasons for running the code or failing the code.

Problem 2: Assume we have another assembly (University) where you refer your Students assembly. _age variable in student class is declared as protected internal.

```
namespace University

{

class univPeople{

Students[] students = new Student[3];
```

```
WorkingStaff[] staffMembers;

//Const logic for this class which fills in both the array of student and staff

public univPeople{

Student Ankit = new Student();


Ankit.fullName="Ankit Mathur";

Ankit._age = 25;

students[0]=Ankit;

}}}
```

Assume there is no property exposed for _age variable in Student class.

Assignment 11 - Inheritance

For the student class as mentioned in the previous assignments derive two classes School Students and College students. Think about their attributes and methods.

Advanced concepts

Assignment 12 – CollectionNLists – Part 1

1. Implement IComparable interface in a Vehicle class to compare two vehicles. Add all the vehicle objects to the array and sort the objects. Print the status of the vehicle.

2. Override Equals Method in your vehicle class to check two vehicles for the equality.

3. Write a class VehicleCollections and Implement IEnumerable interface in it. Your collection's class should maintain an internal list of vehicles. You should be able to enumerate through this collection using foreach.

Assignment 13 - CollectionsNLists – Part 2

Assignment Description: Use your vehicle class for this assignment.

1. Create an array list of Vehicle objects named as VehiclesList.

2. Add your vehicle objects along with derived objects in the list.

3. Enumerate through the list contents and call the tostring method of each object and print the tostring input.

4. Get the list element using indexes and then call the tostring method of each object of different type. Also try to modify the property for a Vehicle object within the loop.
5. Create this assignment under namespace CollectionsNLists2 and name the class as TestCollection. ~~Folder name on winscp server should also be CollectionsNLists under your user account and name the assignment folder as ArrayList.~~

Assignment 14 - - Generic Classes

- This assignment is broken in multiple steps.

Write the same vehicle class as you write in previous assignment, but with Generics. Implement IComparable<Vehicle> , IEquatable<Vehicle> interfaces.

Derive your Car, Truck etc.. classes from the Vehicle class.

Add all the vehicle objects(Vehicle,car,Truck etc..) to List<Vehicle> object and then call the sort method and check if items get sorted according to the functionality given in the CompareTo() method. Change the functionality and check if sorting also changes.

Check different Vehicle objects for the equality.

Web Application and Exception Handling

Assignment 15 - - Web Application

Create a new web application considering the following scenarios:

1. Create a login Page with two textbox for Username and Password and a button submit.On click of submit button save the userName in session Object and use it further in Master Pages to print the Welcome message.

2. For sitemap Create two ASPX Pages, i.e one for Technology News (techNews.aspx) and another for ExtraCurricular Activites(extraCurr.aspx) in the university.

3. Create a sitemap XML file which should have same structure (tree view structure) and show the sitemap of your application.

4. Create Master Page and in the master Page Header you give Optimus Logo, logout link,welcome message and a common element which you think that it should be shown there.

5.Try out the treeview part but don't include it in the master pages.

Assignment 16- - Exception Handling

Write a new exception named as IsCarDeadException in the project CustomExceptions and refer it in your Vehicle Project.

Throw and catch this exception for speed in accelerate and deaccelerate method.

Check the type of vehicle and accordingly apply the condition as bicycle speed cannot go to 100KM/Hr.

Pass the message as "Car has overheated" to the base constructor of your exception class .

In the data object of the exception class you add the time when the car was overheated and the speed at which the car exploded.

Log the exception in the Application event log of machine. Check the way of logging issues from the MSDN.

ADO.Net

Assignment 17- - ADO.Net Part1

Write all the methods in student class and same in StudentDataHandler, XCall should go fro student clas to studentDataHandler class

1. Write method "public bool AddStudent() in student class library which add the studentrecord to the database and based on success or failure of the database method, a message should print on the Add Students screen .

2. Write method "public bool UpdateStudents()"

Call this method from EditStudent screen.

3. Write method "public List<Student> GetAllStudents(int streamID)"

This method should be called while implementing ShowAllStudents screen.

4. Write method "public Dictionary<int,string> GetAllStreams()" in a new class UtilityFunctions.cs(static class) which get all the streams from the Master table 'Stream'.

Use this dictionary object to fill Stream selection list on GUI.

5. Write method "public Dictionary<int,string> GetAllStates()" in UtilityFunctions.cs which get all the states from the Master table 'States'.

Use this dictionary object to fill 'State' selection list on GUI.

6. Write method "public string GetStateName(int stateID)" in UtilityFunctions.cs which will return the name of state. Use this function in ShowAllStudents screen for showing the name of state in gridView.

7. Write method "public string GetStreamName(int streamID)" in UtilityFunctions.cs which will return the name of stream. Use this function in ShowAllStudents screen for

showing the name of stream in gridView.

8. Implement ExceptionHandling in all the methods for catching SqlExceptions.

9. Write a method LogToEventLog(Exception e) in UtilityFunctions.cs file which read this exception object and log the stackTrace in event log.

10. Write a web page 'AllStudents.aspx' which shows all the students in the university in grid view control. Take a selection list in the screen at the top centre position and fill it

with the streams. Admin should be able to select any stream from the selection list and on selection of the stream the screen should show all the students of that stream. Your gridview should give the feature of paging and sorting. Your grid should automatically be sorted based on Name.

11. Incorporate age validation on client side specifing the validator control and give the range for the age.

Assignment 18- - ADO.Net Part2

Develop a new screen 'DeleteStudents(string rollNOs)' for deleting the students. This rollNo string will be comma saparated list of rollNos of students which have to be deleted.

Screen GUI:

Show stream selection list in first row.

Second row should have 2 selection lists separated by left and right arrows.

Show delete button in third row.

Screen Functionality:

1. On selection of a stream from the stream list, the left selection list should filled up with the names of all the students in that stream and their values should be rollNos.

2.On click of right arrow, move the students from left list to right selection list and vice versa for the left arrow(Do it as client side functionality).

3. On click of Delete button, concatenate the RollNos of all the selected students which have to be deleted and a confirmation message should be prompted to ask admin for

the confirmation about deletion(Do it as client side functionality). When clicked 'Yes' button and submit the form to the server side.

4.In server side event call 'Delete students created earlier, and pass the concatenated string' .

5. Show the status on the GUI about the transaction.

Catch all the SqlExceptions.

Assignment 19- - ADO.Net Part3 (Objects Caching)

Cache the contents of data you get from the GetAllStates and GetAllStreams methods discussed in earlier assignment and use it wherever it is required in the application, instead of making multiple calls to the function.

File Handling and mails

Assignment 20- File Handling

Create a new directory in C drive. Add two files to it (FileRead & FileWrite).

- Using DriveInfo class properties to display attributes of C drive.
- Using DirectoryInfo and File classes, display newly created directory and files attribute.
- Display Directory Path using path class
- Using file security class set the access control of file FileRead to be read only
- Read the contents of FileRead and write it on FileWrite.
- Compress FileWrite using GZip compression an save it with a name FileWriteZip
- Name the project as ‘FileHandling’ and Namespace as ‘DirectoriesAndFiles’

Assignment 21- Reading CSV File

Create a StudentsData.csv file, holding the student data. In this .csv file write all the attributes data in comma separated form.

Ask user to browse the file using file Upload control. Onclick of 'SaveStudentsToDB' button call the Static utilityclass's public boolLoadFromCSV(string fileName) function.

In this function Read the student data from the .csv file line by line and create the student objects and add them to the generic list of students .

Insert all the student objects to the database calling the public bool insertStudents(List<Students> studentsData) static method of student class. This method should call the StudentDataHandler's same method. Do the transaction handling inside this method.

On GUI display the status of the request.

Code would not be accepted without

Exception handling ,Naimg conventions ,Coding conventions & proper Comments

Assignment 22- Binary Reader and Writer

Create a binary file and add some data (int, string, char etc.) to it using Binary Writer.

Read the contents back from the file using the BinaryReader and display on console.

Handles all the exceptions esp. I/O exceptions.

Assignment 23- Sending Mail

Make a web form with a textbox and a button.

In Textbox ask user to enter the path of the specified folder. And on clicking ‘Delete’ button first check if the size of file in folder is greater than 100 bytes delete it else append a line to each file ‘Size less than 100 bytes’.

Send a mail to user containing name of files that are deleted.

Subject line should be ‘Deleted Files’.

Body should contain name of the files deleted.

Folder should have at least 4 files with two greater than 100 bytes in size.

XML and Serialization

Assignment 24- XML -1

Create an XML file using XMLTextwriter class. It should be as follows:

<iCalibrator>

<Training day=”1”>

<Chapter>XML-1</Chapter>

<Chapter>XML-2</Chapter>

</Training>

</iCalibrator>

Create a web page with a hyperlink – that will have the url of xml file and on clicking on it the xml file should get open in IE and following buttons

(1) Add Node: On clicking this button a new node (“assignment”) should be added in XML file. It should have an attribute ‘Submitted=”y” and child nodes as <number> 1</number>

(2) First Child: Will display the first child of root node in a textbox.

(3) Insert Before: Insert an empty node (‘testing’) before Training node.

(4) Remove Node: This should remove the ‘Assignment’ node.

(5) Child node: display all the child nodes of root node in a textbox.

(6) Total Nodes: Displays the total number of child nodes of root node.

(7) Replace Child: Replace ‘Testing’ node with ‘Testing Over’ node.

Note: Use Create Navigator method for the functions that are read only. For ex:

(2) & (5)

Assignment 25- XML -2

Using the students.xml file as in the previous assignment, do the following:

1. Find all those students from the students.xml file who are from the MCA Branch.

2. Find all the students with grade as 'D'.

Make a proper choice of classes for getting the output.

Assignment 26- XML -3

Create a Students.xml file. The structure of XML file should be same as discussed in the class.

Your web interface should provide a screen with browsing files upload control. From the .aspx page call the method public bool LoadStudentsFromXMLFile(string fileName) in the static class UtilityClass. Within your code you have to parse the xml file and get the students nodes. Create the student object from each of these nodes and add them to generic list of students and save the student information in the database by calling public bool InsertStudents(List<Students> students) method of student class..

Assignment 27- Serialization - 1

Create a class student having three public members variables:

(1) RollNo(int) (2) Name(string) (3) TotalMarks(int)

and a private member variable (4) grade(char)

and a readonly property retrieving the student grade using marks

```
Public char grade

{ get{

if(totalMarks>60)

return ('D')

else if(totalMarks>=60 && totalmarks <80)


return ('C')

else if(totalMarks>=80 && totalmarks <90)

return ('B')

else if(totalMarks>=90 && totalmarks <100)

return ('A')

else

return('F')

}

}
```

Serialize student object using Binary,XML and Soap serialization.

Don't serialize the grade of student.

Print the state of student object before serialization and after deserializing the serialized data back to the object.

File name should be iCalibrator(.dat,.xml,.soap) and store it in a folder SerializationData(create it manually) in your application.

In the assignment apply all coding standards and break the code in multiple functions wherever required to achieve the modularity. Give the comments to describe the functioning.

You can take help from this link

Assignment 28- Serialization - 2

Use the same student class from the previous assignment and create the collection of student objects.

Serialize the above collection of student objects in Binary,XML and SOAP format. Print the object state before serialization and after deserialization as did in previous assignment.

Maintain the naming convention, coding conventions and write the proper comments.

Assignment 29- Learning XPath and XSLT

Question - Write a xml for students as root node having child elements: Student- name and age as its attributes. Create child nodes as Stream and Address (Address should have further child nodes City and Country).

(1) Load this XML document in C#.

(2) Learn about XSLT and apply it using C# and XPath to display name, age and Address.

(3) For all those students whose stream is PCM, append a new child element- College (value: Engineering).

HTML, CSS and Javascript

Assignment 30- Learning HTML5, CSS 3 and JavaScript

Write a Html code with css (no in text styling allowed in this assignment) to create a Multiple Choice Question Page displaying Question Name at the top (in bold), then instruction text (in italic and give 5 pixels text indentation). Use elements to display answer choices available to the user.

Please refer the below screenshot

:

Few answers are of exclusive type i.e. if that option is selected, no other option can be selected with it.

Clicking on ** ** elements should also select the checkbox associated with that ** ** element.

Answer which is selected should have a different background than the ones that are not selected.

The code should support Internet Explorer 8, Mozilla Firefox and Google Chrome.

(Note: Answer 1, 2 and 4 are ordinary, Answer 3 is of exclusive type).

Other DotNet Concepts

Assignment 31- Web Service

Create a web service that takes student roll no as parameter and returns the student details(Roll number, name, grade)

Name the web service as GetStudentDetail.

Assignment 32 – Entity Framework

Create a sample Website using Entity Framework considering the following scenario:

A user can **Add** User, **Update** User, **Delete** User and **View details** of a user.

A user can upload a document.

There can be two types of Users:

1. Admin
2. Normal User

Difference between Admin and Normal user is that the Normal user does not have privileges for adding, updating or deleting user. He has the right to upload the document.

Unit testing

Assignment 33 - Unit Testing

Learn MSTest and how to attach it with a unit test created to debug the

application. Create unit test cases and execute using MSTest for the following

scenario:

Consider a scenario where a user needs to enter his 10th percentage in a text box of a Webpage.

You need to write test cases for all possible scenarios like:

Boundary values, Exception thrown, Positive Conditional testing, Negative Conditional testing, etc..

Create a sample Web Application for the above scenario with the namespace **SampleWebApp** and write your test cases in a separate project with namespace **SampleWebApp.Test.**

Use your testCases to identify the Code Coverage of your Unit test cases(Use

Visual Studio Code coverage tool)

Code Coverage should be at least **90%.**

Design patterns

Assignment 34 – Design Patterns

Learn about following Design patterns:

- Singleton
- Factory
- Adapter
- MVC
- MVP

Create a sample application for each pattern.