

# **Mercer County Community College**

## **Division of Business and Technology**

### **IST 231**

### **ADVANCED C# PROGRAMMING**

#### **COURSE DESCRIPTION**

A continuation of C# programming that addresses advanced topics including advanced Windows programming, accessing and updating databases from local and remote locations, and web based client-server applications. Laboratory exercises use the .NET programming environment to reinforce the materials that are covered in the lecture and the texts.

Text (s): **Reference Division Booklist**

Prerequisites: **IST 229 and at least 30 college credits.**

Co-requisites: **None**

Credits: **3**

Lecture Hours: **2**

Lab Hours: **2**

<p><b>Food and Drink are strictly prohibited in classrooms as per Health and Safety Laws. Students may not bring in chemicals or cleaning fluids of any kind without the appropriate MSD sheets</b></p>
---

**Course Coordinator: Winston H. Maddox**

**Latest Review: Fall 2003**

## **I. COURSE OUTLINE**

The course consists of nine (9) modules that review and extend the students understanding of object-oriented windows programming, local and remote database access, web programming, and the development of client-server applications.

## **II. OVERALL COURSE OBJECTIVES**

Upon completion of this course, students will be able to:

- Modify and create object oriented programs in C# that access and update databases on local and remote computers.
- Modify and write programs in C# that generate Web pages for delivery via an Intranet or the Internet.
- Modify and develop object oriented windows applications that pass data to web pages and receive data from web pages using a browser as the client software.
- Modify and create object oriented windows applications that function as client applications that access host computer services.

## **III. SPECIFIC MODULE OBJECTIVES**

### **A. Review of Object-Oriented Windows Form Programming (Students will):**

1. Review the material in the first 23 Chapters of Sharp and Jagger and the first 12 chapters of Coombs, and prepare a typed memo describing the three topics that the student finds most confusing.
2. Select a topic or topics from those that are confusing to classmates, research the topic using other texts, the library, and the internet, prepare a brief 20 to 40 minute presentations to help other students understand the concepts, and conduct the presentation.
3. Design and develop an object-oriented windows application. The following are potential projects; however, the student may propose an alternative (**1<sup>st</sup> major project**):
  - Simulates the execution of a 16 instruction computer that uses an 8-bit instruction that consists of 4 instruction bits and 4 relative address bits.
  - Develop an application that accepts data pairs, computes the best linear approximation, and displays the linear approximation and the raw data in the same area on a Windows form.

## **SPECIFIC MODULE OBJECTIVES** (cont'd)

- Develop an advanced “child’s calculator” that in addition to the standard four functions, also supports square-root, sign-change, memory-clear, memory-plus, and memory-minus.
  - Develop a hang-person word game with age appropriate vocabulary sets for young children.
4. At the discretion of the instructor, each student is to complete an in-class preliminary examination, with or without a programming component to demonstrate that he/she is prepared for advanced study.

### **B. Using Databases (connections, reading, writing, and deleting data; and ADO classes; Students will):**

1. Demonstrate the ability to connect to a local Access Database on the student’s machine.
2. Demonstrate the ability to connect to a SQL Server Database on the student’s local machine.
3. Demonstrate the ability to connect to a SQL Server Database that is only available on a single server in the laboratory, rather than each local personal computer.
4. Demonstrate a basic understanding of displaying and updating database information using intrinsic controls by completing the text exercise.
5. Demonstrate a basic understanding of accessing a database using ADO.NET programmatically by completing and thoroughly commenting the text exercise.
6. Document a basic understanding of the two methods of database access, dragging controls from the Server Explorer and writing self-contained code, by writing a brief report to compare and contrast the two methods. The report should use at least five current texts or articles.

### **C. Binding Controls to Databases, and Connecting, Disconnecting, and Reconnecting to Databases (Students will):**

1. Demonstrate the ability to bind a control to a table in a dataset using code and using Visual Designer, by completing and thoroughly commenting two simple exercises in the text.
2. Demonstrate the ability to bind joined datasets, so called complex data binding, using programmatically generated SQL statements and the Query Builder, by completing and thoroughly commenting two exercises in the text.
3. Demonstrate a basic understanding of accessing, disconnecting from, updating and editing data, and reconnecting prior to updating the database.
4. Demonstrate an advanced understanding of using databases by designing and programming an application to manage product categories for the “Northwind database” (2<sup>nd</sup> major project).

#### **D. Files (Students will):**

1. Demonstrate a basic understanding of traditional console input, output, and error handling, by writing a version of the classic “Hello World” program that also says hello to three other people and the courses’ instructor
2. Demonstrate a basic ability to create, open, read, write, close, and delete files by completing a laboratory project to accept input and to store the input in a text file. The input to test the program will consist of the first two paragraphs of the U.S. Declaration of Independence.
3. Students will demonstrate a thorough understanding of file I/O by developing a program that reads one of three files and displays letter counts, sorted by letter; and word counts, sorted by word. The three files will consist of the U.S. Declaration of Independence, The U.S. Declaration encrypted using an 8-bit XOR mask, and the U.S. Declaration of Independence translated as literally as possible into German (without compound nouns). The student will be asked to use the program to determine the XOR mask and to build a German to English translation table. (**3<sup>rd</sup> major project**)
4. Students will demonstrate their understanding of databases and files by writing a brief report to explain the differences between files and databases and to recommend when one or the other is the more appropriate approach. This report should use several references.

#### **E. XML (Students will):**

1. Demonstrate a basic understanding of using XML by completing and thoroughly commenting the Timesheet exercises in the text.
2. Demonstrate the ability to work with XML by modifying the Timesheet system to collect the Employee’s telephone number and address as well as the Employee ID and Name, and to collect a work-level code and a customer-code for the activities.
3. Further demonstrate an understanding of XML by designing and programming a new application that displays Timesheet data to prepare customer bills.

#### **F. Active Server Pages - ASP.NET (Students will):**

1. Demonstrate a basic understanding of using ASP.NET by completing the exercise in the text that produces a form that can be displayed in a browser.
2. Demonstrate a basic understanding of form validation by completing the exercises in the text that validates the data that is entered in the preceding form.
3. Demonstrate an understanding of server and client side validation using C# and ASP.NET by writing a brief report to discuss the advantages and disadvantages of both approaches

### **G. Integrating Database Access and Active Server Pages (Students will):**

1. Demonstrate the ability to create a logon form by completing the exercise in the text.
2. Demonstrate the ability to present database information on a form, displayed in a user's browser by completing the customer table exercise in the text.
3. Demonstrate the ability to update, add, or delete entries in a database by completing the customer table exercise in the text.
4. Gain experience in the advanced features of the DataGrid control by completing the product categories application in the text.
5. Demonstrate the ability to integrate database access and active server pages by either designing or programming an application to display and manage employee data using the Northwind Database or by developing an equivalent application of interest to the student. A suitable application will require a logon password, provide sorted access to a simple database to those who logon successfully, and then permit updates, additions, and deletions from the browser.  
**(4<sup>th</sup> major project)**

### **H. Building and Accessing XML Web Services (Students will):**

1. Gain experience building XML Web Services by completing the exercises in the text.
2. Demonstrate an understanding of building XML Web Services by adding two methods to show employees and to show customers by type.
3. Gain experience accessing XML Web Services by completing the exercises in the text.
4. Demonstrate an understanding of accessing XML Web Services by creating form applications to retrieve and display employee and customer information.

### **I. Final Project (Students will):**

1. Demonstrate a thorough understanding of C# and the advanced features of the .NET environment by proposing, designing, programming, testing, and documenting a complex application that uses both databases and remote web access; and by presenting their project to their classmates. The **final project, which is equivalent to two major projects**, must consist of a database, with windows forms to maintain and update the data base, for example a stock portfolio database; a means to access information that relates to the database using the Internet, for example getting stock prices; a means to analyze the retrieved information, for example detect price changes and or detect warning prices; and a means to graphically display information.

Students will be required to enter, compile, and test programs outside of class. There are suitably configured computers in the library. The primary text for this course includes a C# programming environment and a C# compiler. .

Students will be expected to prepare for each class by reading assignments from the textbooks and by entering the code for the laboratory exercise in an appropriate file.

The course will consist of two hours of lecture and two hours of laboratory each week. The two hours of lecture will review the reading material, help students with any questions that they have, and cover some of the simpler examples that are included in the text. During the two-hour laboratory periods, students will work through the step-by-step laboratory exercises at his/her own pace. The student will document the results of his studies by adding detailed comments to the laboratory exercises. Students will be expected to complete the step-by-step exercises and several other assignments outside of class.

During the course, each student will be expected to complete five significant programming projects, with the last counting twice as much as the other projects.

Textbook: Sharp, John & Jagger, Jon, (2002). *Microsoft Visual C#.NET Deluxe Learning Edition*. Redmond, WA: Microsoft Press. ISBN: 0-7356-1633-7. **Includes C#.NET standard software. REQUIRED**

Price, Jason & Gunderloy, Mike. (2002). *Mastering Visual C#.NET*. San Francisco, CA: Sybex. ISBN 0-7821-2911-0. **REQUIRED**

Coombs, Ted. (2002). *Programming with C#.NET*. Albany, NY: Delmar. ISBN: 0-7668-5008-0. **REQUIRED**

Five reports or memos, two with presentations, are required during this course to demonstrate the students understanding of various concepts. Each will be graded.

When students are given an assignment to describe something in writing, they will be required to prepare a report or a memo to the instructor explaining or answering the given question. Reports are to be typed and should include a list of references. Students name must appear on every page. Multiple page documents should be stapled. Written assignments will be graded for content (65%), format or style (20%), and timely submission (15%).

Students will be expected to complete each of the step-by-step exercises in chapters 24 through 31. In Sharp and Jagger or similar exercises in an alternate text. Any exercise that is not completed in the laboratory will be completed as homework. Each assignment is to be thoroughly commented to demonstrate the student's understanding of the code and turned in to the instructor. Each set of laboratories will be graded and the two lowest, non-zero, grades will be dropped.

Students are expected to demonstrate a thorough understanding of the material presented by completing six homework assignments which consist of either writing short programs or extending the work that was completed in the laboratory. Each homework assignment will be graded.

There are five significant programming projects. All programming assignments, laboratories and homework will be graded based on the following:

Submitted On-time	15%
Performs the Required Functions	25%
Uses All Previously Covered Material, Including the Current Topic	10%
Uses a Consistent and Correct Indentation Style	10%
Uses Sound Program Structuring Methods	10%
Uses appropriate variable names that are meaningful but not too long	10%
Contains Comments that Adequately Describe the Code	10%
Uses Constructs that are Typically Used by C# Programmers	10%

This course requires a final with a significant programming component. The course grade will be based on the following weights:

Final Examination	20%
Two major programming projects	30%
Laboratory Assignments	15%
Programming / Homework Assignments	15%
Writing Assignments	10%
Attendance and Participation	10%

**Because of the amount of homework that the instructor needs to grade and because students must stay caught up to master a subject like C# programming, LATE HOMEWORK, LABORATORY EXERCISES, and PROGRAMMING ASSIGNMENTS WILL NOT BE ACCEPTED.**

**Students are reminded of the school's academic integrity policy. In this class, if a student submits work that has been copied from an outside source, without proper citations, or if two or more students submit work that is essentially identical, the grade for the assignment will be ZERO. Grades of ZERO will not be dropped. Students who receive two or three ZEROS, depending on the assignment, will not be able to pass the course.**

The course percentage will be determined by multiplying the student's average in each category by the specified weight and totaling these products. The letter grade will then be determined as follows:

$90 \leq \text{Percentage} \leq 100$	A
$80 \leq \text{Percentage} < 90$	B
$70 \leq \text{Percentage} < 80$	C
$60 \leq \text{Percentage} < 70$	D
$0 \leq \text{Percentage} < 60$	F

**Students are reminded that they must be aware of the deadline for the drop-add period. The professors who teach this course are not obligated to withdraw students who stop coming to class. WITHDRAWING FROM ANY COURSE IS THE STUDENTS RESPONSIBILITY.**