

**Subject Code: IMT-48**

**Subject Name : DOT NET PROGRAMME**

**Notes:**

- Write answers in your own words as far as possible and refrain from copying from the text books/handouts.
- Answers of I<sup>st</sup> Set (Part-A), II<sup>nd</sup> Set (Part-B), III<sup>rd</sup> Set (Part – C) and Set-IV (Case Study) must be sent together.
- Mail the answer sheets alongwith the copy of assignments for evaluation & return.
- Only hand written assignments shall be accepted.

- First Set of Assignments: 5 Questions, each question carries 1.5 marks.
- Second Set of Assignments: 5 Questions, each question carries 1.5 marks.
- Third Set of Assignments: 5 Questions, each question carries 1.5 marks. Confine your answers to 150 to 200 Words.
- Forth Set of Assignments: Two Case Studies : 7.5 Marks. Each case study carries 3.75 marks.

**Objectives**

- Describe how XML and related technologies fit into the .NET Platform
- Create a well-formed XML document.
- Use XML schemas to validate an instance document
- Describe how XML Document Object Model(DOM), XML Path Language (XPath), XSL Transformations (XSLT), and XQuery are used to create real-world solutions.

**Contents**

<b>INTRODUCING C#</b>
Overview of c#; need for c#; evolution of c#; characteristics of c#; applications of c#; difference between c# and c++; difference between c# and java; creating first program in c#; compiling c# program; executing c# program;
<b>OVERVIEW OF C#</b>
Namespaces; Accessing Namespace Members; Nesting Namespaces; Standard Namespaces In .Net; Adding Comments; Main Returning A Value; Using Aliases For Namespaces; Passing String Objects To Writeline() Method; Command Line Arguments; Main() Method With A Class; Providing Interactive Input; Using Mathematical Functions; Multiple Main Methods; Compile Time Errors; Program Structure; Program Coding Styles
<b>VARIABLES AND DATA TYPES STRUCTURE</b>
Tokens; literals; variables; data types; value types; reference types; declaration of variables; initialisation of variables; default values; constant variables; scope of variables; boxing & unboxing;
<b>OPERATORS AND EXPRESSIONS</b>
Operators; Arithmetic Operators; Relational Operators; Logical Operators; Assignment Operators; Increment And Decrement Operators; Conditional Operator; Bitwise Operators; Arithmetic Expressions; Evaluation Of Expressions; Precedence Of Arithmetic Operators; Type Conversions; Implicit Type Conversion; Explicit Conversion; Operator Precedence And Associativity; Mathematical Functions
<b>CONTROL STATEMENTS</b>
Decision making statements; simple if statement; the if-else statement; nesting if-else statements; the switch statement; the operator; the while statement; the do-while statement; the for loop; the foreach statement; jumps in loops; break statement; continue statement; goto statement; return statement

<b>METHODS IN C#</b>
Introduction to methods; declaring methods; main method; method overloading;
<b>HANDLING ARRAYS</b>
Introduction to arrays; declaring and initialising one dimensional array; declaring and initialising multi dimensional array; variable size arrays; the system.array class; searching an array; sorting an array; accessing an array; copying an array; the arraylist class;
<b>STRING HANDLING</b>
Overview Of String; Using Special Characters In Strings; Creating New Strings; Trimming And Removing Characters; Strings Padding; Strings Comparison; Changing Character Casing; Strings Parsing; String Search; Search Strings Using String Methods; Search Strings Using Regular Expressions
<b>STRUCTURES AND ENUMERATIONS</b>
Structures; declaring a structure; structs with methods; advantages of structs; structs and methods; structs and operator overloading; nested structs; enumerations; declaring an enumeration; initialising enumerators; enumerator base types; enum members enumerator type conversion; implicit conversions in c#; boxing conversions; explicit conversions in c#; explicit numerical conversions; explicit enumeration conversions; explicit reference conversions; un-boxing conversions; user-defined conversions;
<b>CLASSES AND OBJECTS</b>
Basic concept of object oriented programming; classes in c#; Access modifier; Enhance features of c#; the this reference; Static members; Indexers; properties; The readonly member; constant members; nesting of classes; Constructor and destructor; Constructors; destructors;
<b>INHERITANCE AND POLYMORPHISM</b>
Inheritance; defining a subclass; visibility control; defining subclass constructors; multilevel inheritance; hierarchical inheritance; overriding methods; hiding methods; abstract methods and classes; sealed classes and methods; polymorphism;
<b>INTERFACES</b>
Defining an interface; extending interfaces; implementing interfaces and inheritance; explicit interface implementation; abstract classes and interfaces;
<b>OPERATOR OVERLOADING</b>
Introduction to operator overloading; defining operator overloading; overloading unary operators; overloading unary plus and unary minus operator; overloading bitwise complement operator; overloading prefix increment and decrement operator; overloading true and false operators; overloading binary operators; overloading comparison operators; overloading conversion operators;
<b>EXCEPTION HANDLING</b>
Introduction To Exception Handling; Need Of Exception Handling; Types Of Errors; Understanding Exceptions; The Exception Hierarchy; Using Try Block; Using Catch Block; Using Finally Block; Using Throw Block; General Exception Handler; Throwing User-Defined Exceptions; Handling Exceptions In Expressions; Using Checked Operator; Using Unchecked Operator

**Reference:**

1. *Beginning Visual basic .net anne prince*
2. *c# and the .net frameword, robert powell*
3. *Microsoft Developer Network Reference (MSDN)*

## **ASSIGNMENTS**

**FIRST SET OF ASSIGNMENTS**  
**Marks**

**Assignment-I = 5**

### **PART– A**

1. (a) What is common language Runtime and how it is different from common type system.  
(b) Explain the difference between CTS Class Types and CTS Interface Types.
2. (a) What is ILDasm.exe?  
(b) Mention any five members of System Array class.
3. (a) Explain the difference between CTS Class Types and CTS Interface Types.  
(b) Explain the rule of System Array Base Class in Business application.
4. (a) What is Server Explorer?  
(b) What is web service ?
5. (a) What is ASPX pages?  
(b) What is ADO.NET?

**SECOND SET OF ASSIGNMENTS**

**Assignment-II = 5 Marks**

### **PART– B**

1. (a) What are literals? Explain various types of literals in C#.  
(b) Write a program to explain boxing and unboxing.
2. (a) Explain conditional operators.  
(b) How data type conversion is done in c#.
3. (a) write a C# program to show the following output.  
1  
1 2 1  
1 2 3 2 1  
1 2 3 4 3 2 1  
(b) generate the code to reverse the following number  
34567=76543
4. (a) Explain the main method.  
(b) Write a program for method overloading.
5. (a) Define the advantage of general exceptional handler.  
(b) Explain the use of checking operator in C# application.

---

### **PART – C**

1. (a) Describe the function and use of BindingManagerBase object.  
(b) Define complex data binding and explain how you use it with a data grid.
2. (a) How can you allocate memory to arrays?  
(b) How can you copy an array?

- (c) Write a program in C# to sort the element of a linear array
3. (a) Distinguish between ordinary class properties and methods and shared class properties and methods.  
(b) Explain how you can use the debug object to display information in the output window.
  4. (a) Distinguish between radio button and check box controls.  
(b) Describe how you work with the items in a combo box list or a list box.
  5. Explain "is-a" and "has a" relationship with the help of examples.
- 

### **CASE STUDY-1**

1. Write a C# program to generate a triangle with fibonacci numbers (series given below) as elements of every row. For example, for  $m = 5$ , the output should be

```

1
1 1
1 1 2
1 1 2 3
1 1 2 3 5.
```

Given,  $\text{fibonacci}(n) = \text{fibonacci}(n-1) + \text{fibonacci}(n-2)$ , for  $n \geq 3$   
 $= 1$  for  $n = 1, 2$

### **CASE STUDY-2**

2. Write a program to calculate the telephone bill. Each time when the program is run, it should first ask for the program first ask for the number of calls made and then print the amount payable as per the following rules:
  - First 200 calls are free
  - Next 100 calls @Rs. 1.80 per call
  - And the next calls @ Rs. 2.30 per call