



📞 978.256.9077

✉️ [admissions@brightstarinstitute.com](mailto:admissions@brightstarinstitute.com)

## Learning to Program with VB.NET

**Duration:** 35 hours

**Price:** \$900

**Prerequisites:** Familiarity with computers.

**Description:** This hands on VB.NET programming course provides an introduction to programming using the VB.NET language. Students are introduced to the application development cycle, structure of programs, and specific language syntax. The course also contains "Thinking Like a Programmer" sections that provide students insight on how to develop common algorithms. The course covers console and file I/O, string and character manipulation, managing data using collections and fundamental object-oriented programming concepts. Error handling techniques are also emphasized. The course also introduces how to access databases using ADO.NET and illustrates how to build user interfaces using Windows Forms. Comprehensive hands on exercises are integrated throughout to reinforce learning and develop real competency.

**This class is intended for non-programmers.**

### Course Overview

#### Application Development Fundamentals

- Overview of Programming Concepts
- Understanding the Structure of a VB.NET Program
- Understanding Data Types
- Working with Variables
- Reading From and Writing to the Console
- Overview of the .NET Framework
- Using Visual Studio
- Thinking Like a Programmer

#### Managing the Flow of an Application

- Conditional Constructs
  - Working with If/Else Constructs
  - Working with Select Case
- Looping Constructs
  - Working with Do/Loop Constructs
  - Working with For and For Each
- Thinking Like a Programmer
  - Making Decisions

- Designing Algorithms
- Finding Patterns in Code

- Designing Loops

## Working with Data

- Using Integer Data Types
- Using Floating Point Data Types
- Using Characters and Strings
- Using Dates
- Using Booleans
- Working with Constants and Literals
- Understanding Option Strict

## Object-Oriented Programming

- Understanding Object-Oriented Concepts
- Designing Classes
- Coding Properties and Methods
- Initializing Objects with Constructors
- Overloading Constructors
- Declaring and Instantiating Objects
- Calling Properties and Methods
- Understanding Value Types vs. Reference Types
- Working with Shared Data Members and Methods
- Thinking Like a Programmer
  - Finding your Classes

## Working with Data Collections

- Understanding Arrays
- Declaring and Instantiating Arrays
- Iterating through Arrays
- Working with `System.Array` Methods
  - Copying, Sorting, Searching and Resizing
- Passing Arrays to Methods

## Using Procedures to Modularize Code

- Defining and Calling Subroutines
- Defining and Calling Functions
- Understanding Variable Scope
- Overloading Procedures
- Passing Parameters
- Understanding `ByVal` vs `ByRef` Parameters
- Understanding the Call Stack
- Thinking Like a Programmer
  - Finding the Procedures
  - Refactoring
  - Improving Productivity with Snippets

## Understanding Namespaces

- Understanding the Role of Namespaces
- Understanding .NET Namespaces
- Defining Custom Namespaces
- Referencing Members in a Namespace
- Using the `Imports` Statement

## Building Inheritance Hierarchies

- Understanding Inheritance
- Building Derived Classes
- Understanding Constructors in Derived Classes
- Defining and Using Protected Class Members
- Understanding Polymorphism
- Defining Overridable Methods
- Overriding Methods
- Understanding Abstract Classes

Working with `ParamArray` Parameters

- Working with Command-line Arguments
- Understanding .NET Collections
- Managing Data Using `List (Of T)`
- Managing Data Using `Dictionary (Of TKey, TValue)`
- Working with LINQ
- Making LINQ Queries
- Enumerating LINQ Query Results
- Working with Anonymous Types
- Using Extension Methods with LINQ

## Building GUIs with Windows Forms

- Overview of Windows Forms
- Designing Forms
- Working with Controls
  - Using Labels and Textboxes
  - Using Buttons
  - Using Checkboxes and Radio Buttons
  - Using Menus
  - Using list Controls
- Handling Events

## Exception Handling

- Understanding Exception Handling
- Using `Try/Catch` to Handle Exceptions
- Working with the Exception Class
- Understanding Exception Propagation
- Using `Finally` to Manage Cleanup Processing
- Throwing Exceptions

## Working with Strings

- Working with the String Class
- Working with String Literals and Escape Sequences
- Understanding String Manipulation Performance Issues

- Defining Abstract Classes
- Inheriting from Abstract Classes

## Accessing Databases Using ADO.NET

- Understanding the ADO.NET Object Model
- Opening Connections
- Executing Queries Using Commands
- Iterating Through Results Using `DataReaders`
- Calling Stored Procedures
- Passing Parameters to Stored Procedures
- Working with `DataSets`
- Binding Data to Controls

## Working with Files, Directories and Streams

- Using the `System.IO` Namespace
- Discovering Drives
- Working with Directories
- Working with Files
- Parsing a File Path
- Understanding Streams
- Working with `FileStream`
- Reading and Writing Text Files
- Understanding other Types of Streams

## Building N-Tier Applications

- Building Large Scale Applications
- Designing N-Tier Applications
- Building .NET Assemblies
- Referencing Assemblies

- Working with the `StringBuilder` Class
  - Formatting Output with `String.Format`
- 

978.256.9077

[admissions@brightstarinstitute.com](mailto:admissions@brightstarinstitute.com)

Copyright© Bright Star Institute