Object-Oriented Design

In this course, you’ll gain fundamental understanding and refresher with Object-Oriented Programming Principles, including the UML. This course is excellent preparation for pursuit of any object-oriented programming curriculum study of language-specific tools (e.g. programming or managing development projects in C++, .NET technologies such as VB.NET, C#, Java, etc.). This course is ideal for software engineers, developers, designers and project managers who will be working on programming projects in an object-oriented development environment. Topics include describing the differences between procedural and object-oriented programming (OOP); defining classes, abstraction, objects, encapsulation, inheritance, and polymorphism; using the UML to model and develop a class diagram, identifying and working with class design guidelines, building objects using types of composition.

Who should take this course?

This course is intended for software engineers, developers, testers, designers and project managers who are working on programming projects in an object-oriented development environment.

Course Objectives

- Describe the differences between procedural and object-oriented programming (OOP).
- Define Classes, Abstraction, Objects, Encapsulation, Inheritance, and Polymorphism.
- Use the UML to model a class diagram.
- Identify and work with class design guidelines.
- Develop UML diagrams.
- Use Inheritance and Composition.
- Build Objects using types of composition.

Course Details

- Length: 12 hours
- Format: Classroom
- Prerequisites: Windows and PC basics required; C# Programming: 1 recommended

The above prerequisites are considered to be the basic skills and knowledge needed prior to taking this class. Instructors will assume your readiness for the class materials and will NOT use class time to discuss prerequisite materials.
Course Contents

Introduction to Object Oriented Programming

Object Oriented Concepts

Anatomy of a class

Designing with Objects

Frameworks and Reuse: Designing with Interfaces and Abstract Classes

Building Objects

UML Overview