Systems & Network Training

Software development fundamentals

A 3 day Hands on training course

Save on expenses. This course is also available on your premises.

Description

This three-day MTA Training course helps you prepare for Microsoft Technology Associate Exam 98-361, and build an understanding of these topics: Core programming, Object-Oriented programming, general software development, web applications, desktop applications, and databases. This course leverages the same content as found in the Microsoft Official Academic Course (MOAC) for this exam.

Key outcomes

By the end of the course delegates will be able to:

- ✔ Describe core programming.
- ✔ Explain Object Oriented programming.
- ✔ Describe general software development.
- ✔ Describe Web applications.
- ✔ Describe desktop applications.
- ✔ Explain how databases work.

Training Approach

This structured course uses Instructor Led Training to provide the best possible learning experience. Small class sizes ensure students benefit from our engaging and interactive style of teaching with delegates encouraged to ask questions throughout the course. Quizzes follow each major section allowing checking of learning. Hands on sessions are used throughout to allow delegates to consolidate their new skills.

Details

Who will benefit?
Anyone looking to learn the fundamentals of software.

Prerequisites
None.

Duration: 3 days

Overall rating: ★★★★★

Generic Training	Small Class Sizes	Hands On Training	Our Courseware	Customise Your Course
Generic training compliments product specific courses covering the complete picture of all relevant devices including the protocols “on the wire”.	We limit our maximum class size to 8 delegates; often we have less than this. This ensures optimal interactivity between delegates and instructor.	The majority of our courses use hands on sessions to reinforce the theory.	We write our own courses; courseware does not just consist of slides and our slides are diagrams not bullet point text.	Please contact us if you would like a course to be customised to meet your specific requirements. Have the course your way.
“*Friendly environment with expert teaching that teaches the why before the how.”* G.C. Fasthosts	*“Excellent course. The small class size was a great benefit...”* M.B. IBM	*“Not many courses have practice added to it. Normally just the theoretical stuff is covered.”* J.W. Vodafone	*“Comprehensive materials that made the course easy to follow and will be used as a reference point.”* V.B. Rockwell Collins	*“I was very impressed by the combination of practical and theory. Very informative. Friendly approachable environment, lots of hands on.”* S.R. Qinetiq

Software development fundamentals

Course Content

Core programming
- Computer storage and data types
  - How a computer stores programs and the instructions in computer memory, memory stacks and heaps, memory size requirements for the various data storage types, numeric data and textual data.
- Computer decision structures
  - Various decision structures used in all computer programming languages; If decision structures; multiple decision structures, such as If…Else and switch/ Select Case; reading flowcharts; decision tables; evaluating expressions.
- Handling repetition
  - For loops, While loops, Do...While loops and recursion.
- Understand error handling
  - Structured exception handling.

Object-oriented programming
- Classes
  - Properties, methods, events and constructors; how to create a class; how to use classes in code.
- Inheritance
  - Inheriting the functionality of a base class into a derived class.
- Polymorphism
  - Extending the functionality in a class after inheriting from a base class, overriding methods in the derived class.
- Encapsulation
  - Creating classes that hide their implementation details while still allowing access to the required functionality through the interface, access modifiers.

General software development
- Application life cycle management
  - Phases of application life cycle management, software testing.
- Interpret application specifications
  - Application specifications, translating them into prototypes, code, select appropriate application type and components.
- Algorithms and data structures
  - Arrays, stacks, queues, linked lists and sorting algorithms; performance implications of various data structures; choosing the right data structure.

Web applications
- Web page development
  - HTML, CSS, JavaScript.
- ASP.NET web application development
  - Page life cycle, event model, state management, client-side versus server-side programming.
- Web hosting
  - Creating virtual directories and websites, deploying web applications, understanding the role of Internet Information Services.
- Web services
  - Web services that will be consumed by client applications, accessing web services from a client application, SOAP, WSDL.

Desktop applications
- Windows apps
  - UI design guideline categories, characteristics and capabilities of Store Apps, identify gestures.
- Console-based applications
  - Characteristics and capabilities of console-based applications.
- Windows Services
  - Characteristics and capabilities of Windows Services.

Databases
- Relational database management systems
  - Characteristics and capabilities of database products, database design, ERDs, normalisation concepts.
- Database query methods
  - SQL, creating and accessing stored procedures, updating and selecting data.
- Database connection methods
  - Connecting to various types of data stores, such as flat file; XML file; in-memory object; resource optimisation.