JavaScript: Objects Constructors

page 1

Meet the expert: Don Kiely is a featured instructor on many of our SQL Server and Visual Studio courses. He is a nationally recognized author, instructor, and consultant specializing in Microsoft technologies. Don has many years of teaching experience, is the author or co-author of several programming books, and has spoken at many industry conferences and user groups. In addition, Don is a consultant for a variety of companies that develop distributed applications for public and private organizations.

Prerequisites: This course assumes that you have at least some programming experience in one or more modern programming languages. JavaScript will be particularly easy for you to learn if you have experience with any C-style languages—including Java, C#, C++, or Visual Basic .NET would be helpful—but this is not required. The focus here is not on programming concepts but on the structure, syntax, and use of the JavaScript language.

Runtime: 53:15

Course description: JavaScript is the programming language of the World Wide Web, the code that makes a Web page come alive. This course is focused on working with Constructors, Literals and Prototypes of Objects.

Course outline:

Using the Object Constructor
- Introduction
- Intro to Object-Oriented JS
- Creating JS Objects
- Using the Object Constructor
- Demo: Object Constructor
- Demo: Objects Defined Method
- Demo: None Standard API
- Demo: Read Only Properties
- Demo: Define Methods
- Demo: Exercise Object
- Summary

Using an Object Literal
- Introduction
- Using an Object Literal
- Demo: Object Literal
- Hierarchy with Prototypes
- Demo: Prototypes
- Demo: Customer Object
- Demo: Corporate Object
- Object Identity
- Demo: IsPrototypeOf Method
- Summary

Custom Object Constructors
- Introduction
- Custom Object Constructors
- Demo: Custom Constructors
- Demo: Customer Prototype
- Demo: Adding Methods
- Demo: Using Constructor
- Demo: Customer Object
- New Constructor Prototype
- Demo: Constructor Prototype
- Demo: Object Literal
- Changing the Prototype
- Demo: Changing the Prototype
- Identity with Constructors
- Demo: Identity
- Summary