

Object Essentials

An Introduction to Object-Oriented Software Development

Object-orientation is now considered a mainstream approach to software development. Objects underpin many modern software architectures, programming languages and application environments. The principles on which object-orientation is based are drawn from good software engineering practice and theory, balancing the practical aspects of day-to-day development with longer term goals such as management of complexity. As such, object-oriented development supports different project styles and sizes, ranging from very large-scale, long term infrastructure projects to smaller, bespoke, rapid development projects.

Object-orientation comes with its own set of concepts and terminology. The *Object Essentials* course aims to introduce these to software development professionals. The course is based around lectures, discussion and exercises.

Objectives

- Explain the terminology of object-orientation
- Understand the principles and practices on which object-oriented development is based
- Appreciate the technologies used to build and deploy modern object-oriented systems

Audience

The course is suitable for anyone involved in software development who wishes to gain an understanding of object-orientation, including technical managers, experienced developers, graduate developers, etc.

No prior knowledge of object-orientation or object-oriented programming is assumed. However, knowledge of software development concepts is required, and knowledge of programming is advantageous.

Content

Fundamentals Objects · Encapsulation · Methods and messages · Classes · Object instantiation and lifetimes · Polymorphism · Inheritance · Overriding

Structuring Objects Object references and relationships · Delegation · Design by contract · CRC cards · Command versus query methods · Collections · Representing values, entities and services as objects

Relating Classes Inheritance and substitutability · Multiple inheritance · Abstract classes and methods · Dependency management · Pure interfaces

Programming Object-oriented programming · Features · C++ · Java · C# · Visual Basic · Scripting languages · Libraries and frameworks

Technologies Persistence · Components · Distributed object computing · Multi-tier architectures · COM(+) · .NET · CORBA · Object request brokers · J2EE

Development Object-oriented analysis and design · UML · Use cases · Class diagrams · Interaction diagrams · CASE tools · Patterns · Prototyping · Testing · Refactoring · Iterative and incremental development · Agile development methods

Additional Details

Duration 1 day

Setup Projection facilities for a laptop · Whiteboard and/or flip chart

Contact Kevlin Henney · kevin@curbralan.com · Curbralan Limited · +44 117 942 2990