

Price: R6,900.00 excl. VAT
Duration: 3 days
Code: OOANL

Object-Oriented Analysis using UML

Description

The Object-Oriented Analysis course focuses on the analysis and documentation of object-oriented systems using established OO methodologies and the UML. Delegates learn to write use cases, develop activity and class diagrams, and participate in CRC sessions. Object-oriented concepts are explained and illustrated using simple examples.

Objectives

Delegates who complete the Object-Oriented Analysis course will be able to:

- Understand object-oriented concepts and the object-oriented project lifecycle.
- Write effective use cases.
- Analyse a system in terms of the objects involved.
- Identify high-level classes and the relationships between them.
- Participate in CRC sessions.

Intended Audience

The Object-Oriented Analysis course is ideal for business analysts, project managers and system architects who need to understand the object-oriented process, analyse object-oriented systems, and produce UML diagrams.

Prerequisites

No prior object-oriented or programming knowledge is required.

Course Contents

The lecturer reserves the right to modify the contents of the course to suit the needs of the delegates.

Introduction • The evolution of the object-oriented paradigm. • OOP compared to other programming paradigms. • Advantages and disadvantages of OOP.

Object-Oriented Concepts and Terminology • Classes and objects. • Attributes and behaviours. • Data abstraction and encapsulation. • Polymorphism. • Inheritance and code reuse. • Associations and relationships between classes.

Unified Modelling Language • History and evolution of the UML. • Overview of UML diagrams: use case, class, object, sequence, collaboration, state, activity, component, package, timing, subsystem, model, deployment diagrams. • Common extension mechanisms. • UML modelling tools.

Object-Oriented Methodologies • Traditional Software Development Lifecycle. • Iterative and incremental development. • The need for an OOAD process. • The Rational Unified Approach (RUP). • The Iconix method. • Extreme Programming.

Object-Oriented Analysis • Behaviour analysis and use cases. • Activity diagrams. • Domain modelling. • Class identification and domain classes. • CRC cards and CRC sessions. • Which UML diagrams to use during analysis.