

## Object Oriented Analysis & Design

Cursusduur: 3 Dagen    Cursuscode: OOAD

---

### Beschrijving:

#### Object Oriented Analysis & Design Course Overview

**English - Please note this course is only available in English.**

Nederlands - Let op: deze training is alleen in het Engels beschikbaar.

Français - Veuillez noter que ce cours est uniquement disponible en anglais.

The Object Oriented Analysis & Design course focuses on the fundamental concepts of Object Orientation and UML as part of the

---

### Doelgroep:

#### Who will the Course Benefit?

The Object Oriented Analysis & Design course is aimed at staff and consultants working as part of a development team using OO techniques to develop quality software including Business and System Analysts, Solution Architects, Programmers, Designers, Subject Matter Experts, Project Managers and anyone who needs a good understanding of the use of Object Oriented Analysis and Design within software development.

This course is particularly beneficial for those using Object Oriented development languages such as Java, Python, C#, Visual Basic and Ruby.

---

### Doelstelling:

- Course Objectives
  - By the end of the course delegates should be able to:
    - Explain OO Analysis and Design
    - Describe the main processes and artifacts of OO Analysis and Design
    - Identify and Analyse Requirements
    - Model System Functionality with Use Cases
    - Find and Organise Analysis Classes
    - Create System Domain Models
    - Validate Use Case Models with Stakeholders
    - Create Activity Diagrams to describe System Behaviour
    - Model Object Relationships
    - Understand and apply Generalisation and Inheritance
    - Design Object Behaviour
    - Realise System Design
    - Create Deployment Models
- 

### Vereiste kennis en vaardigheden:

- Delegates attending this course should have a basic understanding of programming in an object-oriented language such as Java, Python, Ruby, JavaScript, etc. Delegates should understand the concepts of classes, attributes and operations. This knowledge can be obtained by attendance on the pre-requisite Introduction to Programming course.
-

## Vervolgcurssussen:

### Further Learning

- Java Programming 1 / Java Developer
  - Python Programming 1
  - Ruby Programming
  - PHP Developer
-

## Cursusinhoud:

Object Oriented Analysis ; Design Training Course Course Contents - DAY 1

Course Introduction

- Administration and Course Materials
- Course Structure and Agenda
- Delegate and Trainer Introductions

Session 1: INTRODUCTION TO OBJECT-ORIENTED ANALYSIS AND DESIGN

- Introduction
- Software Architecture
- Object-Orientation
- Requirements
- Conclusion

Session 2: USE CASE DIAGRAMS

- Use Case Modelling
- Finding Actors and Use Cases
- Use Case Diagrams
- Primary and Secondary Actors

Session 3: USE CASE DESCRIPTIONS

- Developing Use Case Descriptions
- Level of Detail
- Pre-Condition and Post-Conditions
- Main Flow
- Branching within a Flow
- Duplicate Steps
- Future Requirements

Session 4: ACTIVITY DIAGRAMS

- Activity Diagrams for Use Case Flows
- Activity Diagram Notation
- Action States
- Subactivity States
- Transitions
- Decisions
- Parallel Processing
- Loops
- Scenarios
- When to Use Activity Diagrams for Use Case Modelling Object Oriented Analysis ; Design Training Course Course Contents - DAY 2

Session 5: CLASS AND OBJECT ANALYSIS

- Steps in Domain Modelling
- Identifying Candidate Classes
- Noun Extraction Approach
- Common Categories Approach
- Evaluate Candidate Classes
- Elimination/Retention Review
- CRC Cards
- Identify Candidate Attributes and Operations
- Creation of the Static Domain Model

Session 6: CLASS MODELS

- Encapsulation
- Messaging
- UML object notation
- Object attribute values
- Classes
- UML class notation
- Name compartment
- Attribute compartment
- Visibility
- Multiplicity
- Operation compartment

Session 7: MODELLING RELATIONSHIPS

- Association
- Types of Associations
- Aggregation
- Composition
- Reflexive Association
- Navigability
- Associations and attributes
- Association Classes
- Guidelines for Identifying and Modeling Associations

Session 8: GENERALISATION AND INHERITANCE

- Generalisation
- Class generalisation
- Class inheritance
- Overriding

Session 9: MODELLING BEHAVIOUR

- Activity Diagrams
- Defining the Behaviour
- Identifying Steps
- Pre and Post Conditions
- Special Action Types
- Identifying Control and Data Flows
- Data Flows
- Identifying Expansion Regions
- Grouping Actions into Partitions
- Identifying Exceptions and Exception Regions

Session 10: STATE MACHINE DIAGRAMS

- State machines and classes
- Basic state Machine syntax
- States
- State syntax
- Transitions
- Events
- Call events
- Signal events
- Change events
- Time events
- Super States

Session 11: REALISATION AND DESIGN

- Sequence Diagrams
- Detailing Class Operations from Sequence Diagrams
- Creating Sequence Diagrams
- Iteration
- Branching and self-delegation
- Concurrency - Active Objects
- Object State and Constraints

Session 12: IMPLEMENTATION AND DEPLOYMENT

- Solution Modelling
- Solution Architecture
- Infrastructure Architecture 5
- Modelling the Presentation and Data Layers
- Design Modelling
- OO Design Principles
- Modelling Object Behaviour
- Effective Package Design

- Generalisation in OO Analysis and Design
- Identifying Generalisations
- Generalised Classes or Interfaces?
- Mitigating Repeated Inheritance
- Dependency Object Oriented Analysis ; Design Training Course Course Contents - DAY 3

---

### Nadere informatie:

Neem voor nadere informatie of boekingen contact op met onze Customer Service Desk 030 - 60 89 444

[info@globalknowledge.nl](mailto:info@globalknowledge.nl)

[www.globalknowledge.com/nl-nl/](http://www.globalknowledge.com/nl-nl/)

Iepenhoeve 5, 3438 MR Nieuwegein