

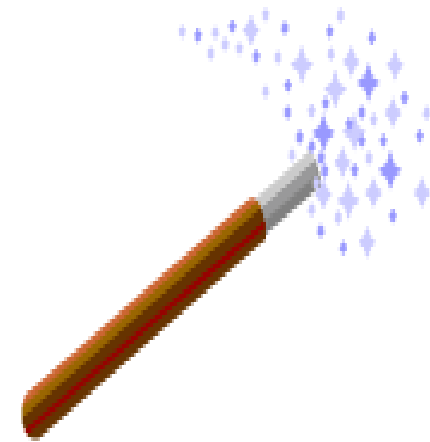
# Welcome

# What Will I Learn?

## Objectives

In this lesson, you will learn how to:

- State the goal of the course
- Explain the course map
- Describe the teaching format used in class
- Become familiar with the computer lab, accounts, and an IDE
- Describe the computer software and folder configuration used in class
- Describe the importance of teamwork and introduce a member of the class





## Why Learn It?

### Purpose

Have you ever wanted to build a video game or make an animated movie? Are you interested in making your own smart phone application? Many of today's video games, movies, and smart phone applications are created using the Java programming language.

This lesson introduces you to the Oracle Academy Java Fundamentals course.





## Course Goal

You will learn the basic elements of computer programming and the Java programming language to start writing your own Java programs.

By the end of this course you should be able to:

- Create simple animations and games.
- Demonstrate knowledge of Java technology and the Java programming language.
- Use the Java programming language to create applications.
- Integrate decision, looping, and other intermediate code to build “smarter” programs.



## Course Map

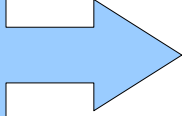
The course map is a lesson planning tool used to plan how you will complete lessons in the amount of time you have available over a unit or semester.

Lessons are designed in a modular way. The course map outlines the duration of each lesson, including minutes per day and total days to review the lesson.

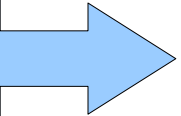
The course map also outlines the corresponding study guides that should be covered each week in tandem with the lessons. These are optional and teach career preparation techniques.

 Course Map (cont.)

**Step 1:  
Oracle Academy  
Java  
Fundamentals**



**Step 2:  
Oracle Academy  
Java  
Programming**



**Advanced Placement  
(US AP College Board  
Computer Science A)**



**Oracle Certification  
(Oracle Certified  
Associate, Java SE7)**

# Teaching and Learning Strategy

The course will be conducted using:

- Lectures
- Hands-on development
- Practice activities
- Quizzes
- Exams
- Projects



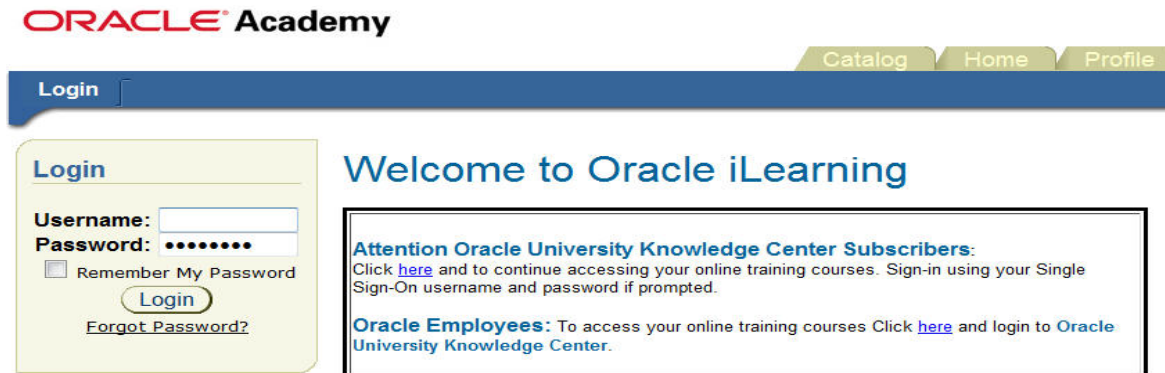


# Oracle iLearning

Oracle iLearning is a learning management system. Your instructor will provide you with an Oracle iLearning username and password.

You will use iLearning to access course:

- Curriculum
- Quizzes
- Exams



The screenshot shows the Oracle Academy login interface. At the top, the Oracle Academy logo is on the left, and navigation tabs for 'Catalog', 'Home', and 'Profile' are on the right. Below this is a blue 'Login' header. The main content area is divided into two sections. On the left is a 'Login' form with fields for 'Username:' and 'Password:'. The password field is masked with dots. There is a 'Remember My Password' checkbox and a 'Login' button. Below the form is a link for 'Forgot Password?'. On the right is a 'Welcome to Oracle iLearning' message. It contains two paragraphs: one for 'Attention Oracle University Knowledge Center Subscribers' with a 'here' link, and another for 'Oracle Employees' with a 'here' link.

**ORACLE Academy**

Catalog Home Profile

Login

**Login**

Username:

Password:

Remember My Password

Login

[Forgot Password?](#)

**Welcome to Oracle iLearning**

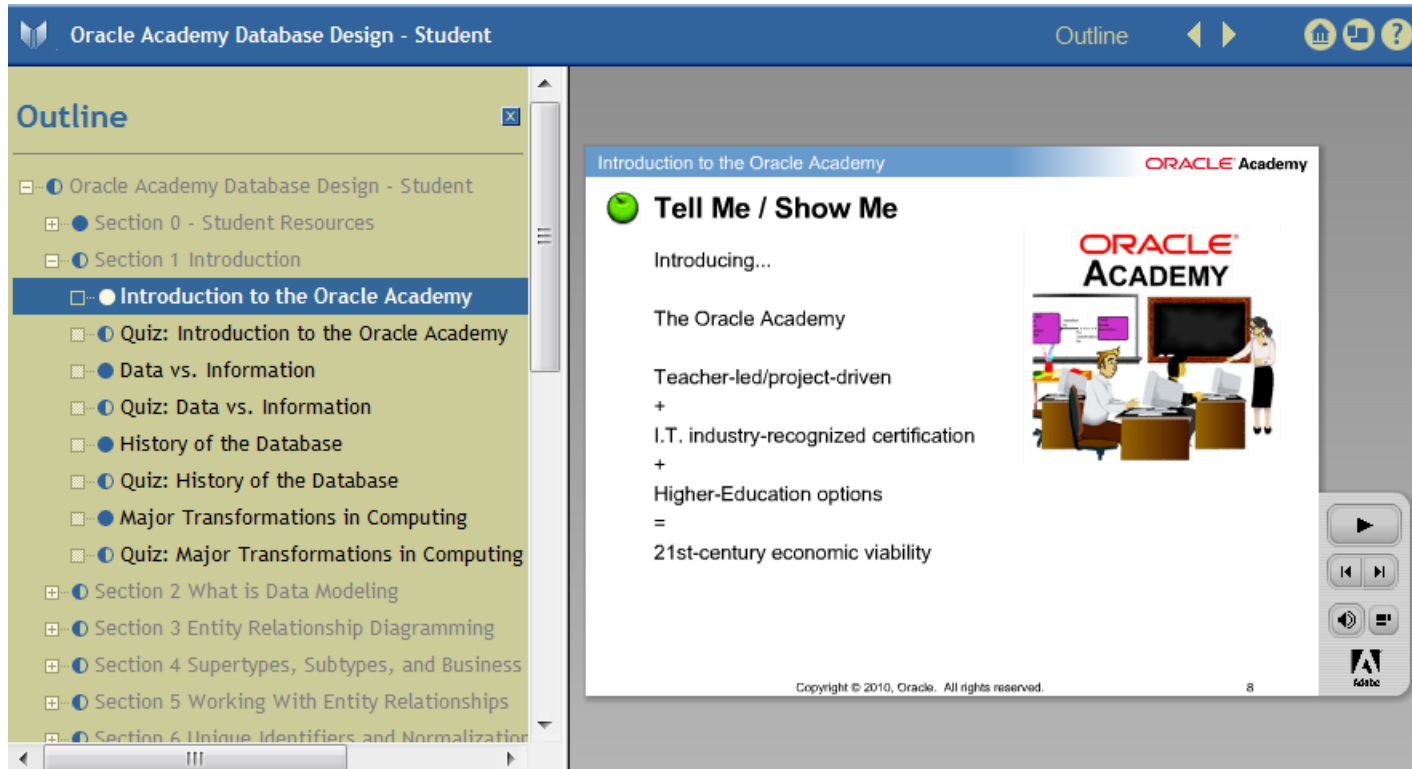
**Attention Oracle University Knowledge Center Subscribers:**  
Click [here](#) and to continue accessing your online training courses. Sign-in using your Single Sign-On username and password if prompted.

**Oracle Employees:** To access your online training courses Click [here](#) and login to Oracle University Knowledge Center.



# Oracle iLearning (cont.)

You can access the slides, quizzes, and exams for the course in the course Outline frame.



The screenshot displays the Oracle Academy Database Design - Student interface. The top navigation bar includes the title "Oracle Academy Database Design - Student", the word "Outline", and navigation icons. The left sidebar, titled "Outline", lists the course structure:

- Oracle Academy Database Design - Student
  - Section 0 - Student Resources
  - Section 1 Introduction
    - Introduction to the Oracle Academy**
      - Quiz: Introduction to the Oracle Academy
      - Data vs. Information
      - Quiz: Data vs. Information
      - History of the Database
      - Quiz: History of the Database
      - Major Transformations in Computing
      - Quiz: Major Transformations in Computing
    - Section 2 What is Data Modeling
    - Section 3 Entity Relationship Diagramming
    - Section 4 Supertypes, Subtypes, and Business
    - Section 5 Working With Entity Relationships
    - Section 6 Unique Identifiers and Normalization

The main content area shows a slide titled "Introduction to the Oracle Academy" with the ORACLE Academy logo. The slide content includes:

- Tell Me / Show Me**
- Introducing...
- The Oracle Academy
- Teacher-led/project-driven
- + I.T. industry-recognized certification
- + Higher-Education options
- = 21st-century economic viability

The slide also features an illustration of a classroom setting with a teacher and students. At the bottom of the slide, it reads "Copyright © 2010, Oracle. All rights reserved." and the number "8". A control panel on the right side of the slide includes a play button, navigation arrows, a volume icon, and a logo.

# Software

In this course you will use the following software:

- Alice3 – A friendly 3D programming environment to create animations for story telling.
- Greenfoot – An interactive 2D application for creating games.
- Eclipse – One of the most popular environments for Java development.
- Internet Browser – Internet Explorer, Mozilla Firefox





# Integrated Development Environment

As a developer, you will use Eclipse, an Integrated Development Environment (IDE) to create your Java applications.

**An Integrated Development Environment, referred to as simply “IDE”, is a software tool used by computer programmers to develop software applications. An IDE includes tools for writing, editing, compiling, deploying and debugging programs.**

# Teamwork

A team is a group of individuals with unique talents focused on a joint goal or product, such as:

- A presentation
- Completing in-class exercises
- Taking notes
- Discussing a topic
- Writing a report
- Creating a new design or prototype



## Importance of Teamwork

In today's business world, one of the keys to organizational success is teamwork. Individuals must work together effectively for companies to remain competitive.

Teamwork is important because:

- Goals can be reached faster and more efficiently.
- It combines skills and talents from multiple people allowing achievements to be made that could not have been made alone.
- It enhances creativity and can result in new and innovative ideas.



# Teamwork

Ask yourself the following questions:

- Have you ever worked on a team?
- What are the benefits of working on a team?
- What are the challenges that you have encountered in the past working on a team?
- How can you address challenges associated with working on a team?
- What makes a successful team?



## Challenges of Working on a Team

- Some group members want to get an A+, others may be happy with a passing grade.
- Some people may manage their time (school, job, activities) better than others.
- Attendance at group meetings may vary.
- Some people may not finish their tasks on time, or even complete them at all.



## Challenges of Working on a Team (cont.)

- Some people do not return calls or emails.
- A group member may spend more time telling others how to do their parts than they do working on their own.
- Group members may overestimate their technical abilities and lead the group into trouble.





# Characteristics of Effective Teams

Effective teams consist of individuals who work together to achieve a common goal and who hold themselves accountable for team output.

Team members on effective teams have:

- A common purpose and clear goals
- The necessary skills and resources
- A common approach to work
- The willingness to share information
- Trust and support in each other
- The ability to work through conflict
- The willingness to take responsibility for team actions.



## Step 1: Forming and Introducing Teams

Teams can be created in different ways:

- Instructor assigns individuals to teams
- Individuals are randomly assigned to teams
- Individuals self-select teams

After joining a team, the next step is to introduce yourself.

- “Hello, my name is George. I love movies. What is your favorite movie?”
- “Hello, my name is Sarah. I really enjoy my smart phone applications. Do you have any good ideas for a new application?”
- “Hello, my name is Caron. I'd like to create a team with you. Would you like to be in my team?”





## Step 2: Establish Team Roles

Once a team is formed, team roles are defined. The table below defines typical job roles and responsibilities of an application development team.

Role	Responsibility
Project Manager	Ensures the vision of the project is realized and the schedule maintained.
Technologist/ Architect	Has a firm understanding of the different technology choices and determines the best way to implement the project.
Programmer	Codes the application.
Graphic Designer/ Interface Designer	Designs the graphical user interface and look and feel of the application.
Technical Writer	Writes the supporting documentation including help system and training materials.



## Step 3: Define Tasks

Once organized, a team will define all tasks that must be completed.

**A task is a very specific work related responsibility that requires completion, usually within a specified period of time.  
A task is always assigned to a single owner.**

### Examples

- Define application name
- Determine project due date
- Brainstorm project ideas



## Step 4: Assign Tasks

Once all tasks are identified, every task is assigned to a single team member.

### Examples

Role	Task
Leader	1. Organize project brainstorm meeting time and place.
Technologist	1. Propose the solution's look and feel.
Strategist	1. Confirm and report back all due dates. 2. Suggest what we can do to complete project early.
Programmer	1. Confirm coding software options.



# Building an Effective Team

- Meet regularly for assignment updates.
- When problems arise, look for solutions, not blame.
- Ask for help if you need it.
- Respect your team members.
- Be flexible and listen to everyone's ideas.



## Try It

Try the following activities:

1. Examine the computer software and setup for the course.
2. Log on to Oracle iLearning and identify all curriculum components.
3. Introduce yourself to one person in the room.
4. Present the person you have met.
5. Form a team and define roles and tasks.



# Terminology

Key terms used in this lesson included:

IDE

Oracle iLearning

Teamwork

Task





# Summary

In this lesson, you learned how to:

- Describe the course goals and objectives
- Explain the course map
- Describe the course learning strategy
- Describe software configuration and curriculum tools
- Form a project team



## Practice

The exercises for this lesson cover the following topics:

- Stating the course goals and objectives.
- Describing the course map.
- Describing the learning strategy.
- Describing the software configuration and curriculum tools.
- Recognizing teamwork and project learning goals.