1

Introduction to the Course

Course Objectives

After completing this course, you should be able to:

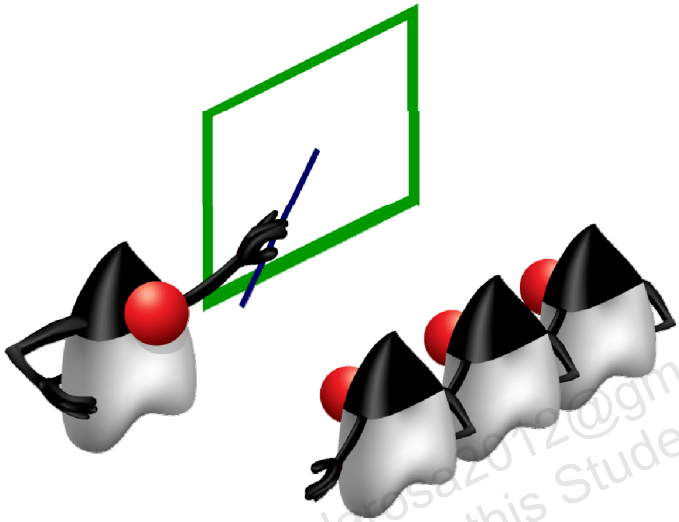
- Apply the Model View Controller (MVC) design pattern to create reusable classes
- Implement a complete program that can be used in an intranet application
- Leverage the Java Persistence API (JPA) in a Java SE environment
- Organize and set up GUI generation and event handling using JavaFX
- Implement the Logging API to generate log messages in the GUI
- Create two-tier and three-tier Java technology applications

Course Objectives

- Connect your application to a REST web service
- Package and deploy a Java SE application
- Secure a Java SE application

Audience

This course is intended for Java developers and JavaFX developers who have at least one year of development experience.

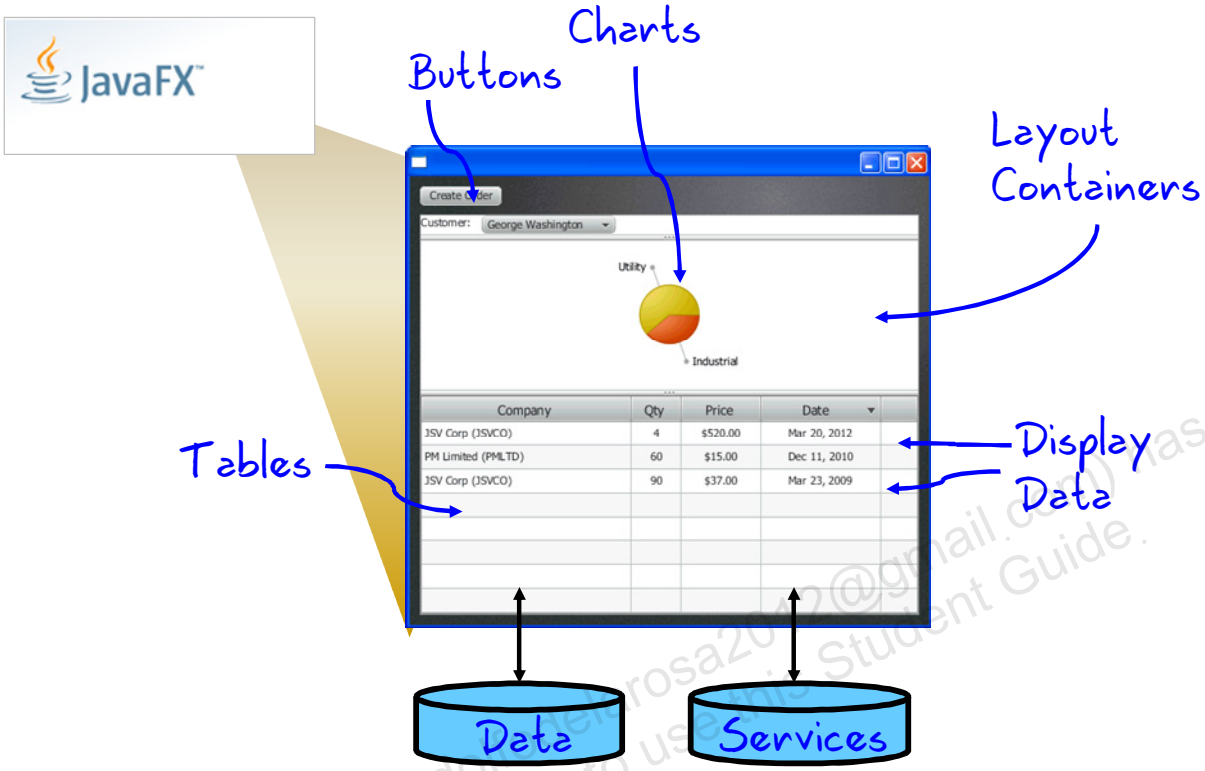


Prerequisite Skills

You should already know how to:

- Use object-oriented programming techniques
- Develop applications by using the Java programming language
- Understand how to implement interfaces and handle Java programming exceptions

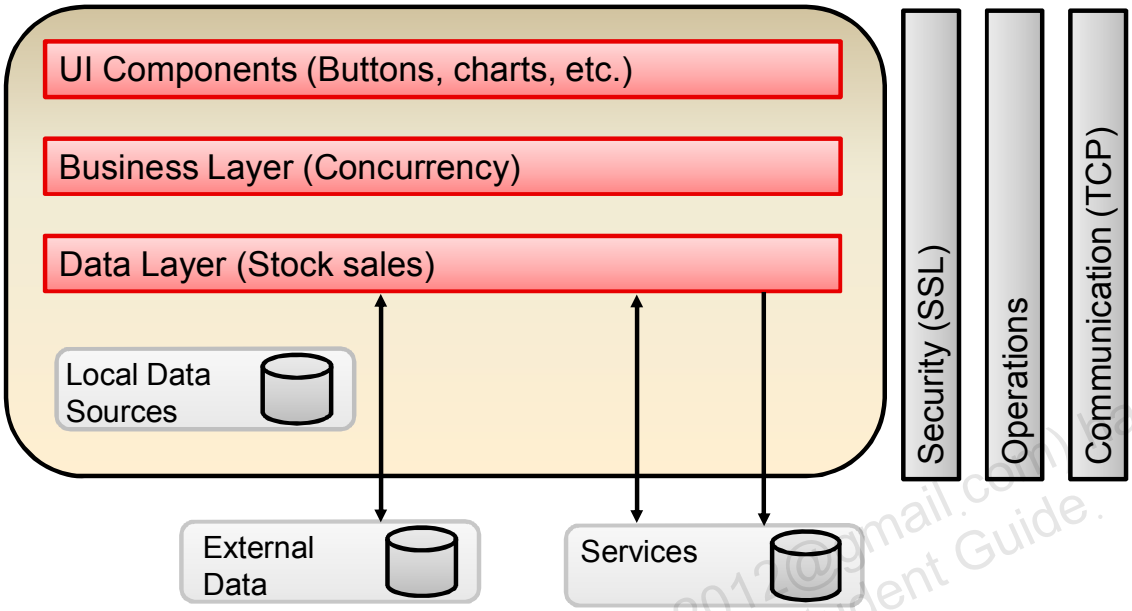
Use JavaFX to Create Rich-Client Applications



A rich-client application is an application that has an interface that relates to the back end without cluttering the user interface. JavaFX has a complete set of buttons, charts, tables, and layout containers that you can use to create a rich user interface. In addition, you can style the client by using CSS. All of these components connect to and display the data back end in an easy-to-read manner.

JavaFX is used to create the rich-client application that we develop throughout this course. Subsequent lessons provide more detail about JavaFX. All of the GUI development lessons use JavaFX.

Rich-Client Application



A rich-client application typically has the following attributes:

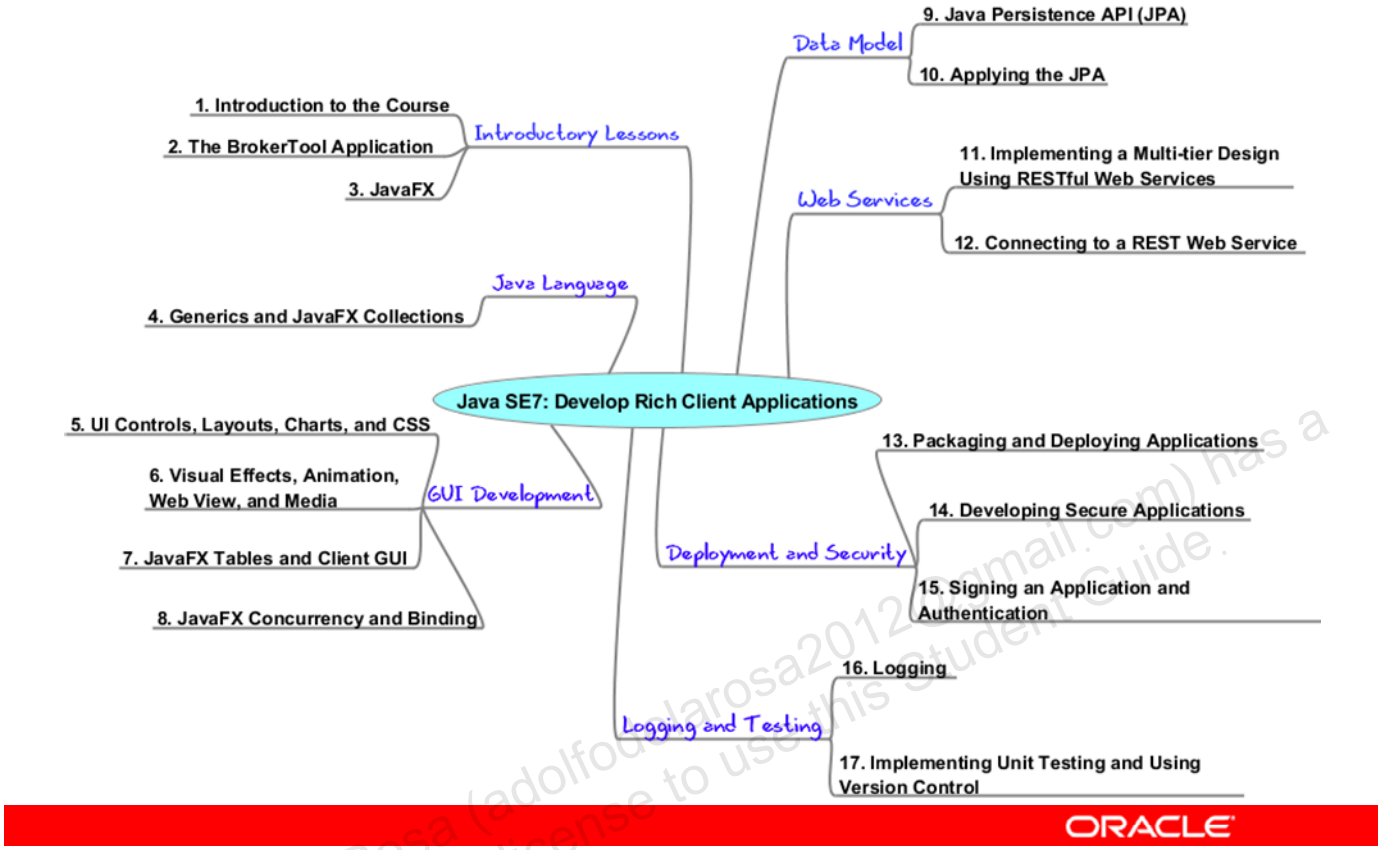
- A stand-alone executable application
- Contains a user interface that has controls and/or forms
- Is deployable from the desktop or web
- Connects to a database and server back end
- Is typically independent of operating system

Things to consider as you develop a rich client application are:

- Application and architecture requirements
- User interface design
- Data accessibility
- Available development tools
- Security
- Localization and accessibility
- Deployment options

Throughout this course, you develop a rich-client interface that connects to the database and server back end that we have created for you.

Course Roadmap



This course is divided into the following units:

- **Introductory Lessons:** These lessons introduce the course, applications, and JavaFX.
- **Java Language:** This lesson is an overview of generics and JavaFX collections.
- **GUI Development:** These lessons cover the rich client development.
- **Data Model:** These lessons introduce and apply the Java Persistence API (JPA).
- **Web Services:** These lessons provide an overview of RESTful web services.
- **Deployment and Security:** These lessons explain how to deploy and secure your application.
- **Logging and Testing:** These lessons cover logging and unit testing.

Schedule

- Day One
  - Lesson 1: Introduction to the Course
  - Lesson 2: The BrokerTool Application
  - Lesson 3: JavaFX
  - Lesson 4: Generics and JavaFX Collections
- Day Two
  - Lesson 5: UI Controls, Layouts, Charts, and CSS
  - Lesson 6: Visual Effects, Animation, Web View, and Media

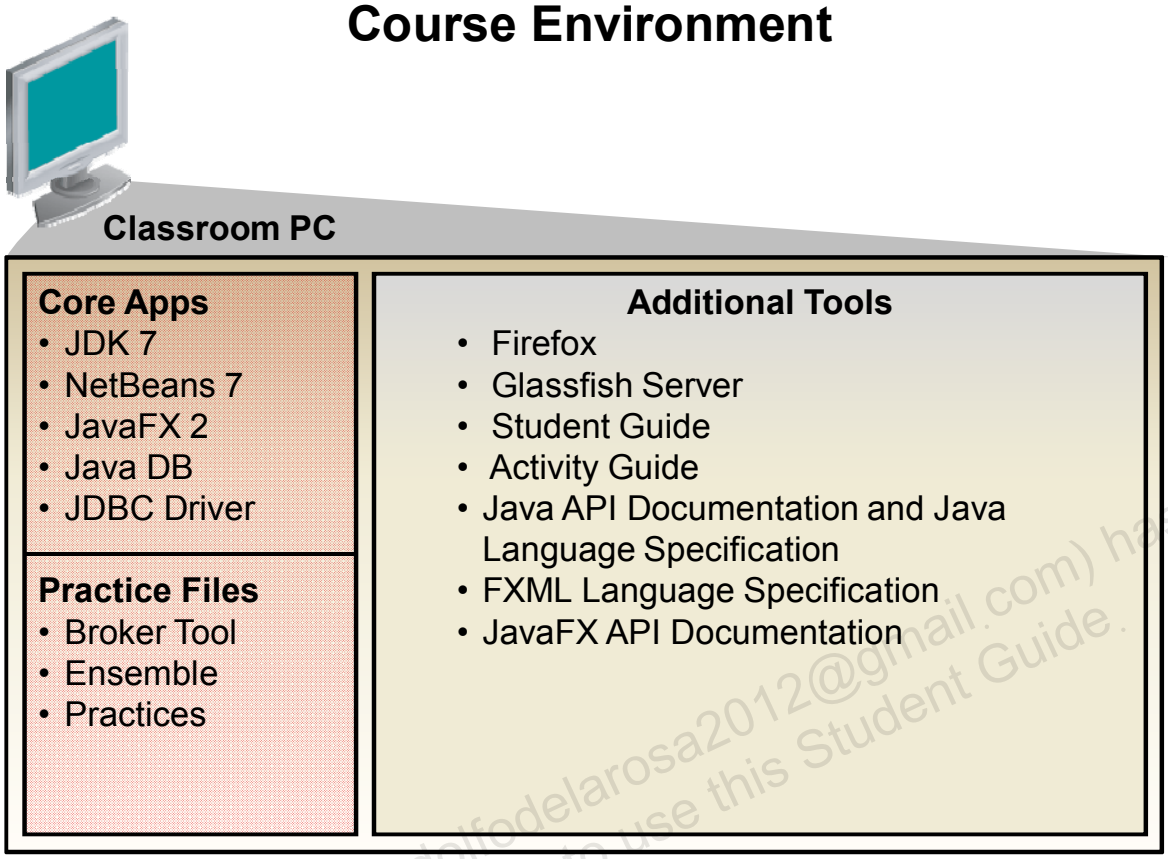
Schedule

- Day Three
  - Lesson 7: JavaFX Tables and Client GUI
  - Lesson 8: JavaFX Concurrency and Data Binding
  - Lesson 9: Java Persistence API (JPA)
- Day Four
  - Lesson 10: Applying the JPA
  - Lesson 11: Implementing a Multi-tier Design with RESTful Web Services
  - Lesson 12: Connecting to a RESTful Web Service
  - Lesson 13: Packaging and Deploying Applications

Schedule

- Day Five
  - Lesson 14: Developing Secure Applications
  - Lesson 15: Signing an Application and Authentication
  - Lesson 16: Logging
  - Lesson 17: Implementing Unit Testing and Using Version Control

Course Environment



In this course, the following products are preinstalled for the lesson practices:

- **JDK 7:** The Java SE Development Kit
- **Firefox:** A web browser is used to view the HTML documentation (Javadoc.
- **NetBeans IDE:** The NetBeans IDE is a free and open-source software development tool for professionals who create enterprise, web, desktop, and mobile applications.
- **JavaFX 2:** A tool used for creating GUIs
- **Java DB:** A database installed with NetBeans that can be used for creating databases. It is included in the JDK bundle.
- **JDBC Driver:** A database driver installed with NetBeans
- **Practice Files:** BrokerTool is the application you develop throughout the course. In addition, you work with JavaFX sample applications such as Ensemble. You use practice files to complete the practices for each lesson.
- **Glassfish Server:** This is an open-source server that is used to deploy applications.
- **Student Guide:** The guide has all of the materials that are discussed in class. In addition, the guide includes two appendixes that provide additional information about NetBeans IDE.

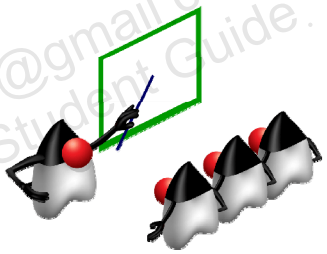
- **Activity Guide:** These are resources to use during the practice portions of the course.
- **Java API Documentation and the Java Language Specification:** The API documentation is the application programming interface specification, and the language specification describes specific language uses.
- **FXML Language Specification**
- **JavaFX API Documentation**

Quiz

a. What is your name?
b. What do you do for a living, and where do you work?
c. Why are you interested in Java?

Facilities in Your Location

- Enrollment, registration, sign-in
- Badges
- Parking
- Phones
- Internet
- Restrooms
- Labs
- Lunch
- Kitchen/snacks
- Hours
- Materials (paper, pens, and markers)



ORACLE

Summary

In this lesson, you reviewed the course objectives and the tentative class schedule. You met your fellow students, and you saw an overview of the course outline and computer environment that you will use during the course.

Enjoy the next five days of *Java SE 7: Develop Rich Client Applications*.