Deploying and Maintaining the Soccer Application

15

ORACLE

Interactive Quizzes

Objectives

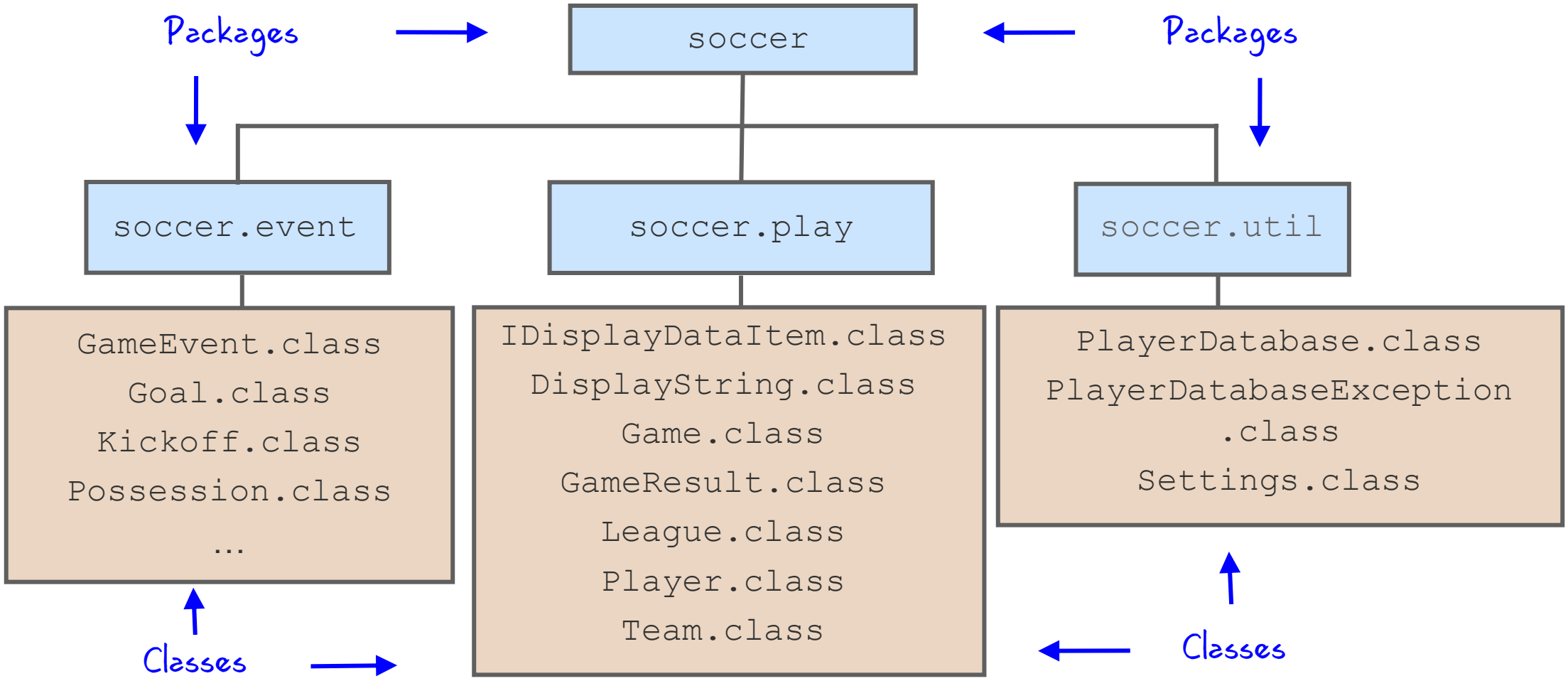
After completing this lesson, you should be able to:

- Deploy a simple application as a JAR file
- Describe the parts of a Java application, including the user interface and the back end
- Describe how classes can be extended to implement new capabilities in the application

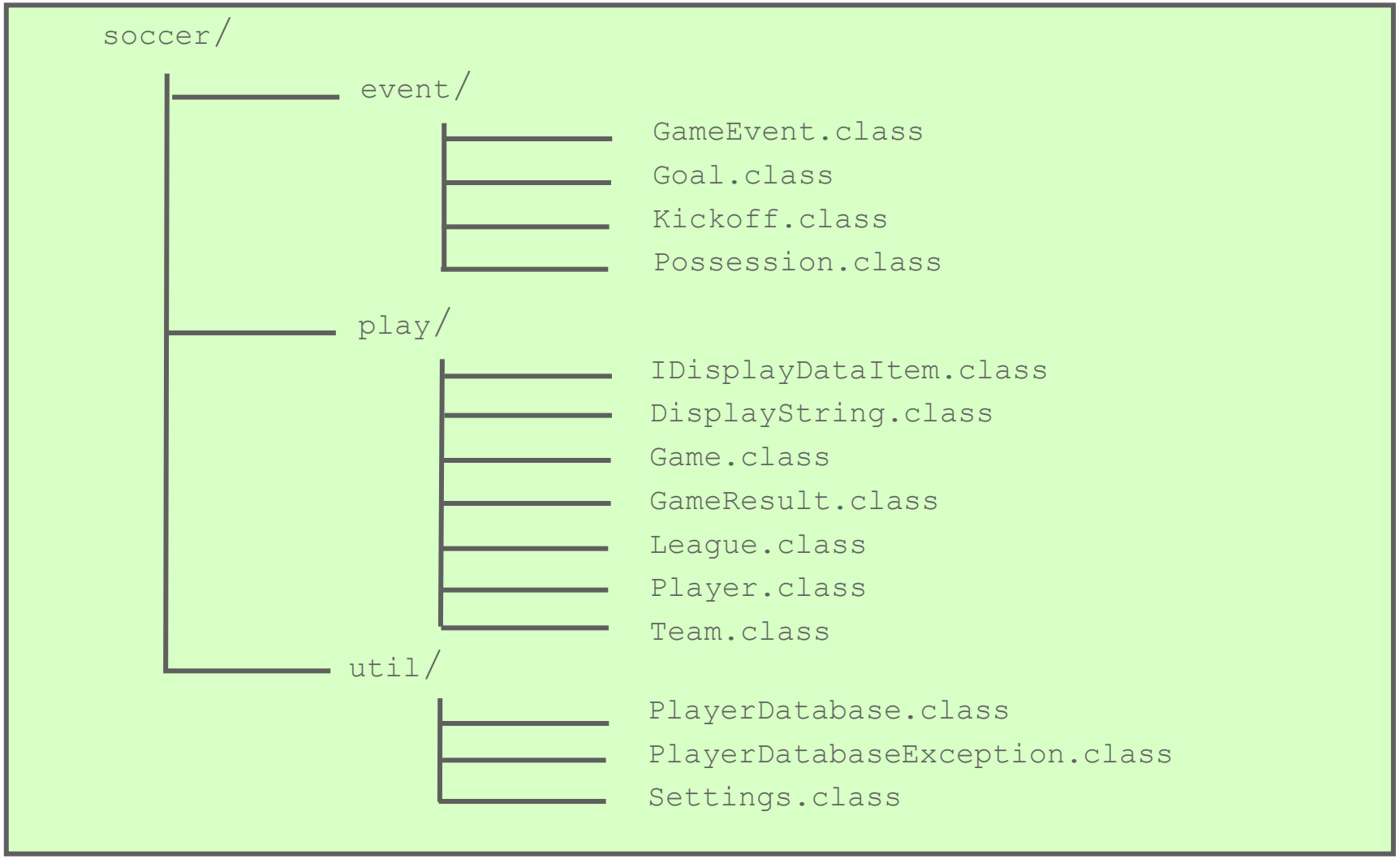
Topics

- Packages
- JARs and deployment
- Two-tier and three-tier architecture
- The Soccer application
- Application modifications and enhancements

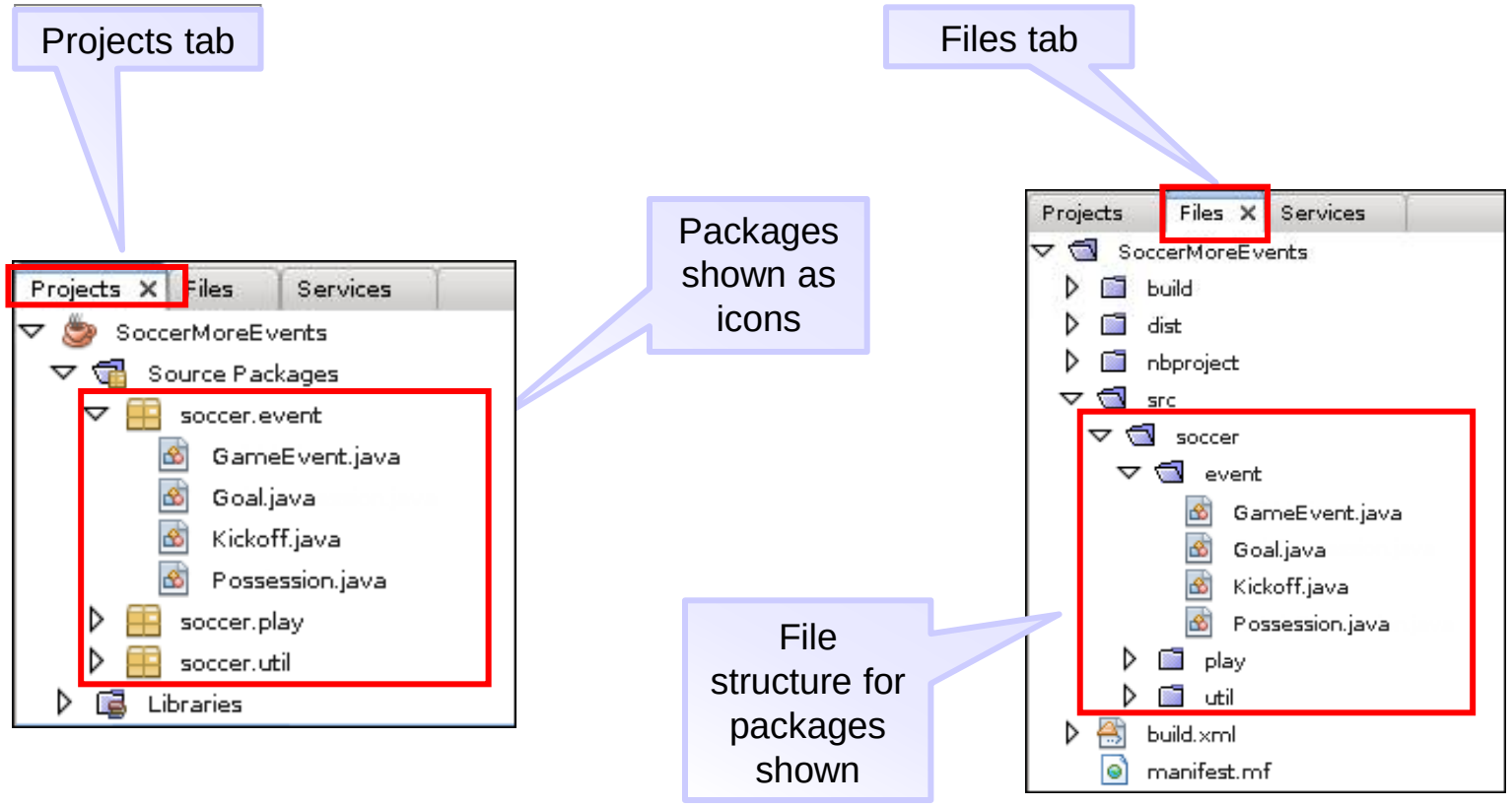
Packages



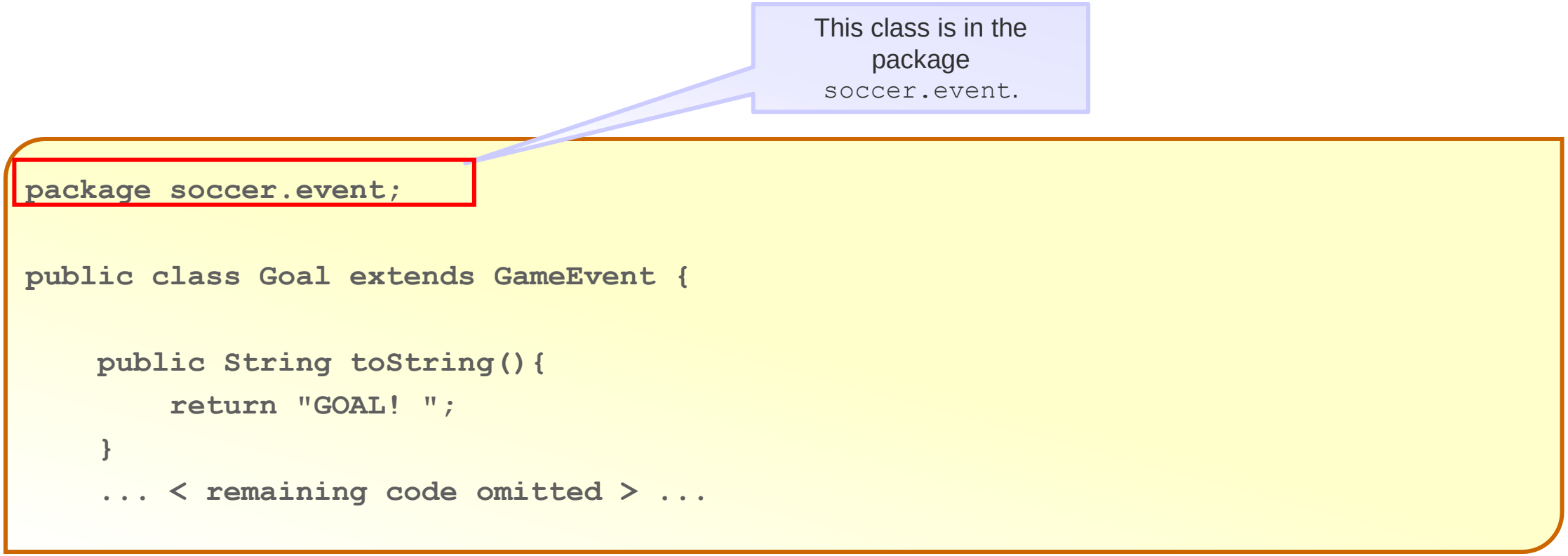
Packages Directory Structure



Packages in NetBeans



Packages in Source Code

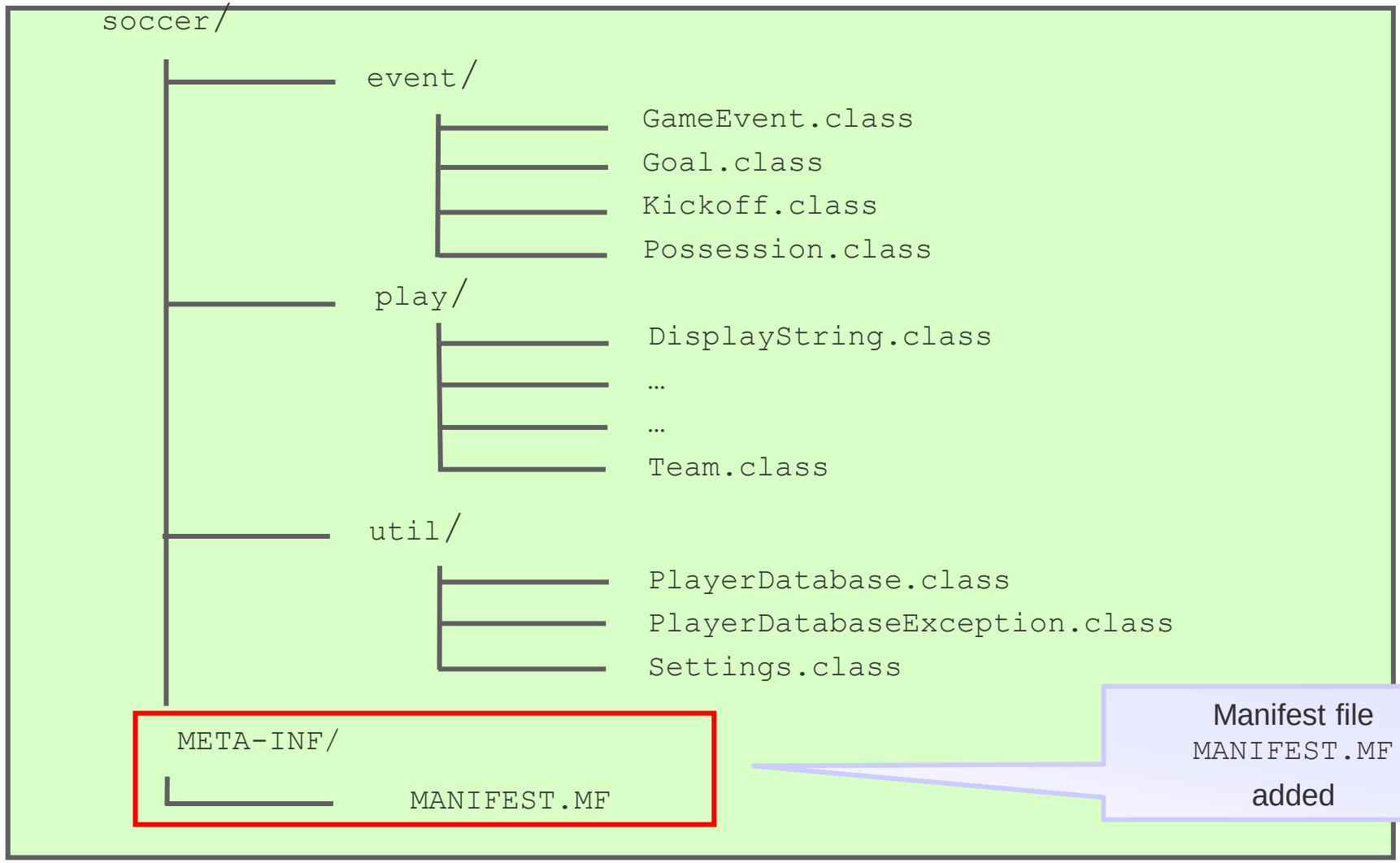


The package that a class belongs to is defined in the source code.

Topics

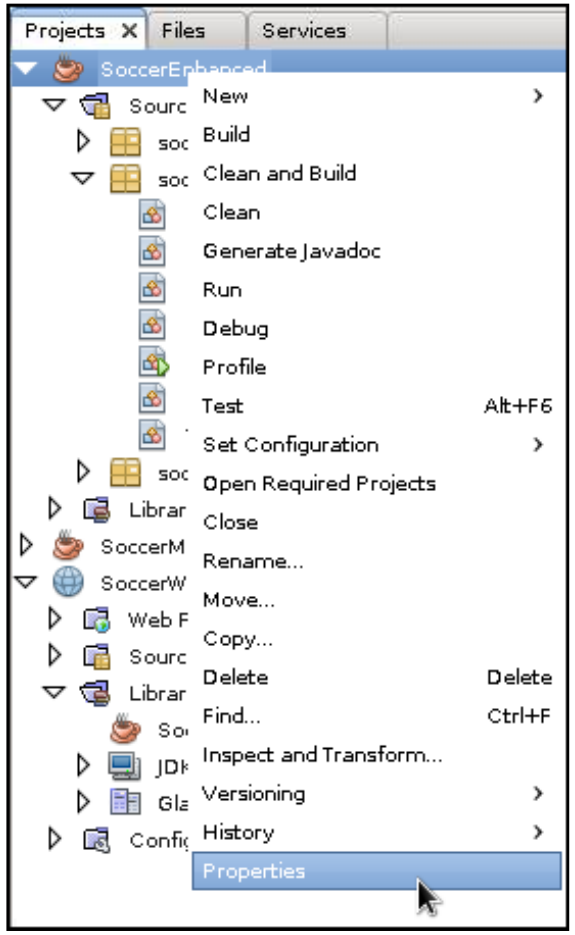
- Packages
- JARs and deployment
- Two-tier and three-tier architecture
- The Soccer application
- Application modifications and enhancements

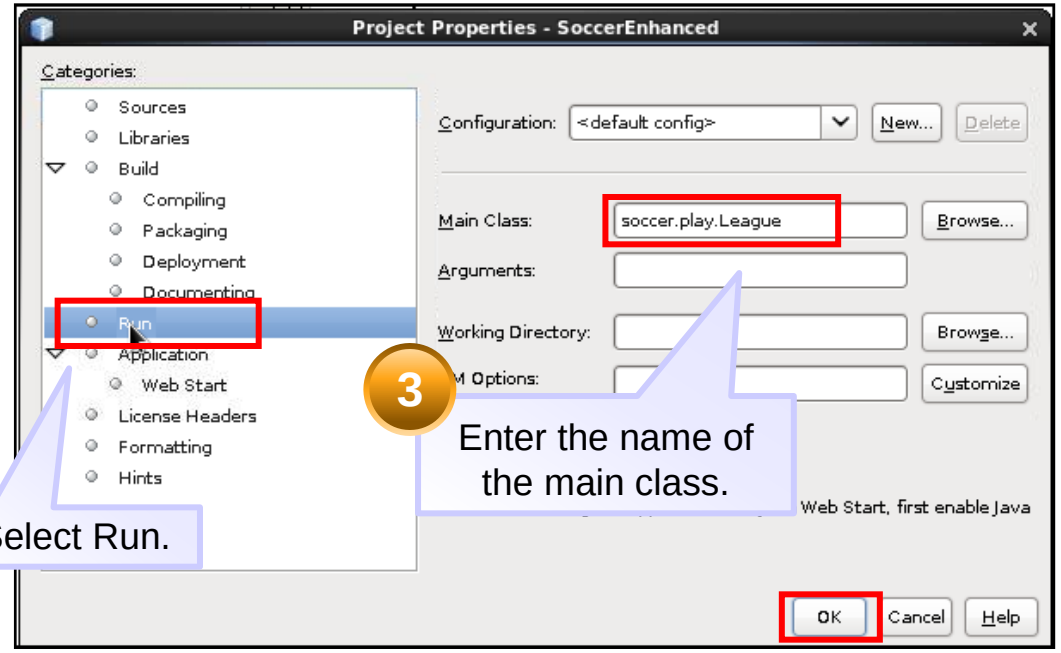
SoccerEnhanced.jar



The JAR file contains the entire class directory including the manifest file.

Set Main Class of Project





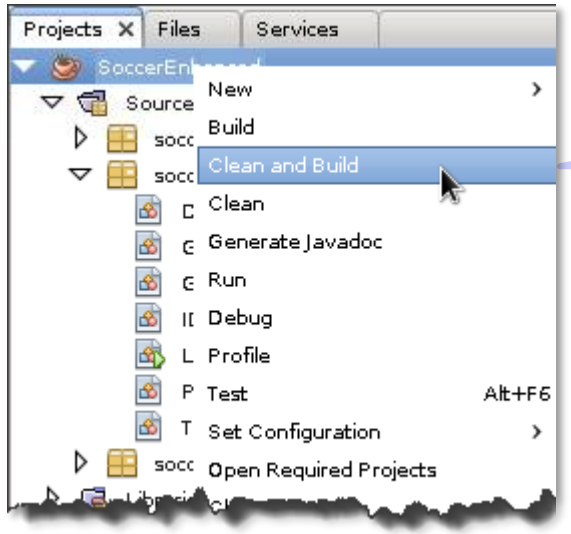
1 Right-click the project and select Properties.

2 Select Run.

3 Enter the name of the main class.

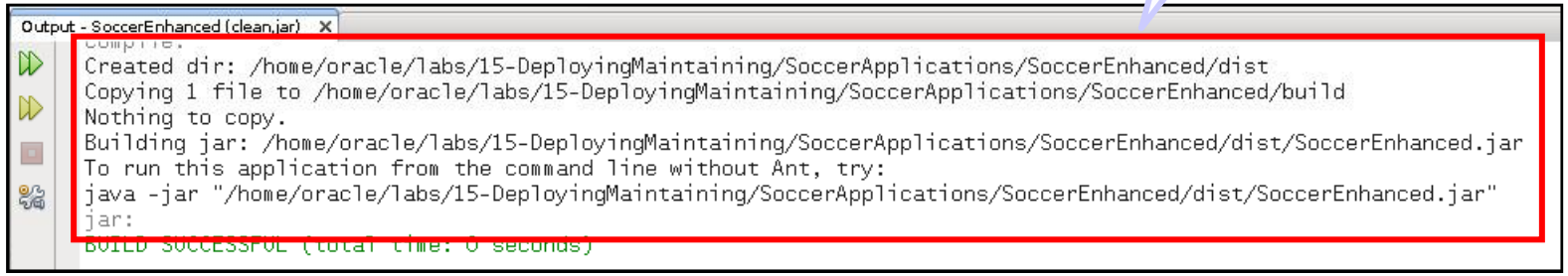
4 Click OK.

Creating the JAR File with NetBeans

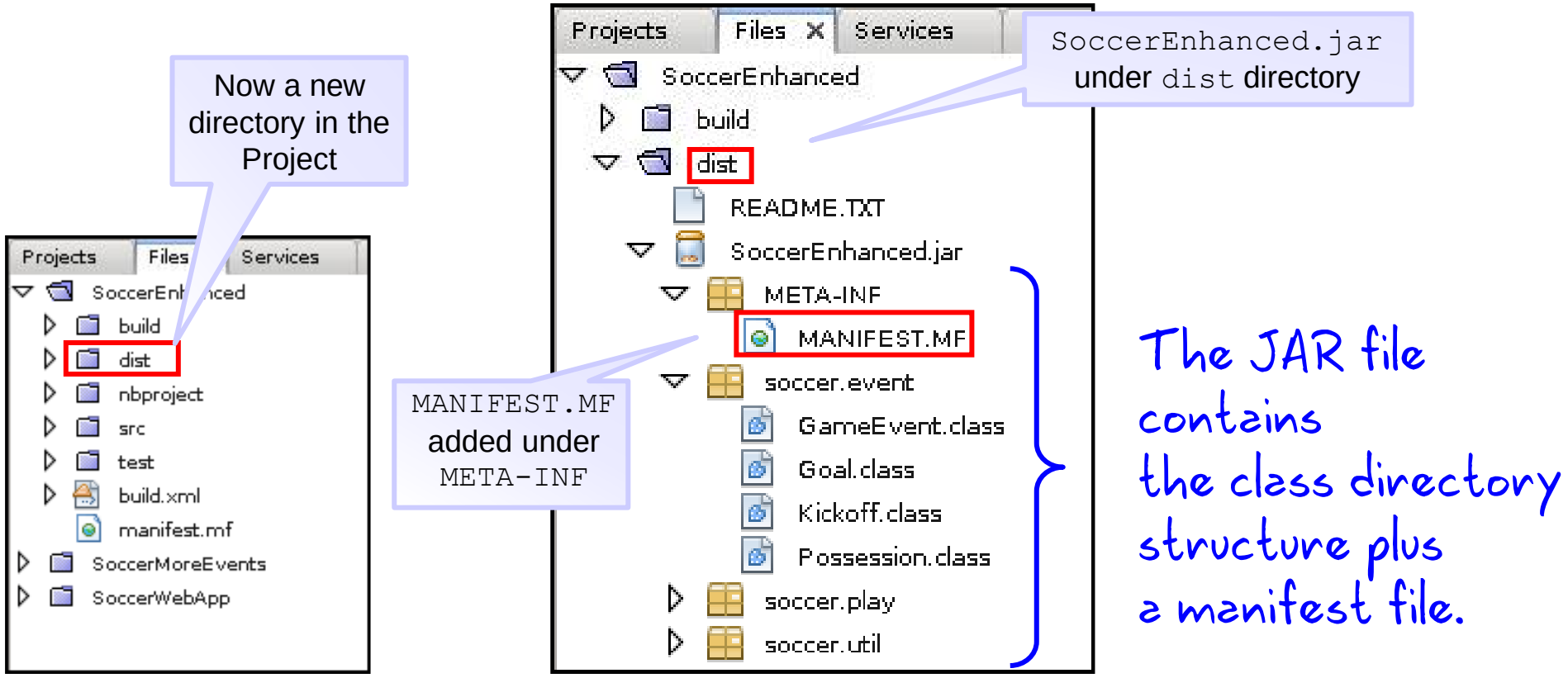


1. Right-click the project and select “Clean and Build.”

2. Check the output to ensure that the build is successful.



Creating the JAR File with NetBeans



The JAR file contains the class directory structure plus a manifest file.

Topics

- Packages
- JARs and deployment
- Two-tier and three-tier architecture
- The Soccer application
- Application modifications and enhancements

Client/Server Two-Tier Architecture

Client/server computing involves two or more computers sharing tasks:

- Each computer performs logic appropriate to its design and stated function.
- The front-end client communicates with the back-end database.
- The client requests data from the back end.
- The server returns the appropriate results.
- The client handles and displays data.

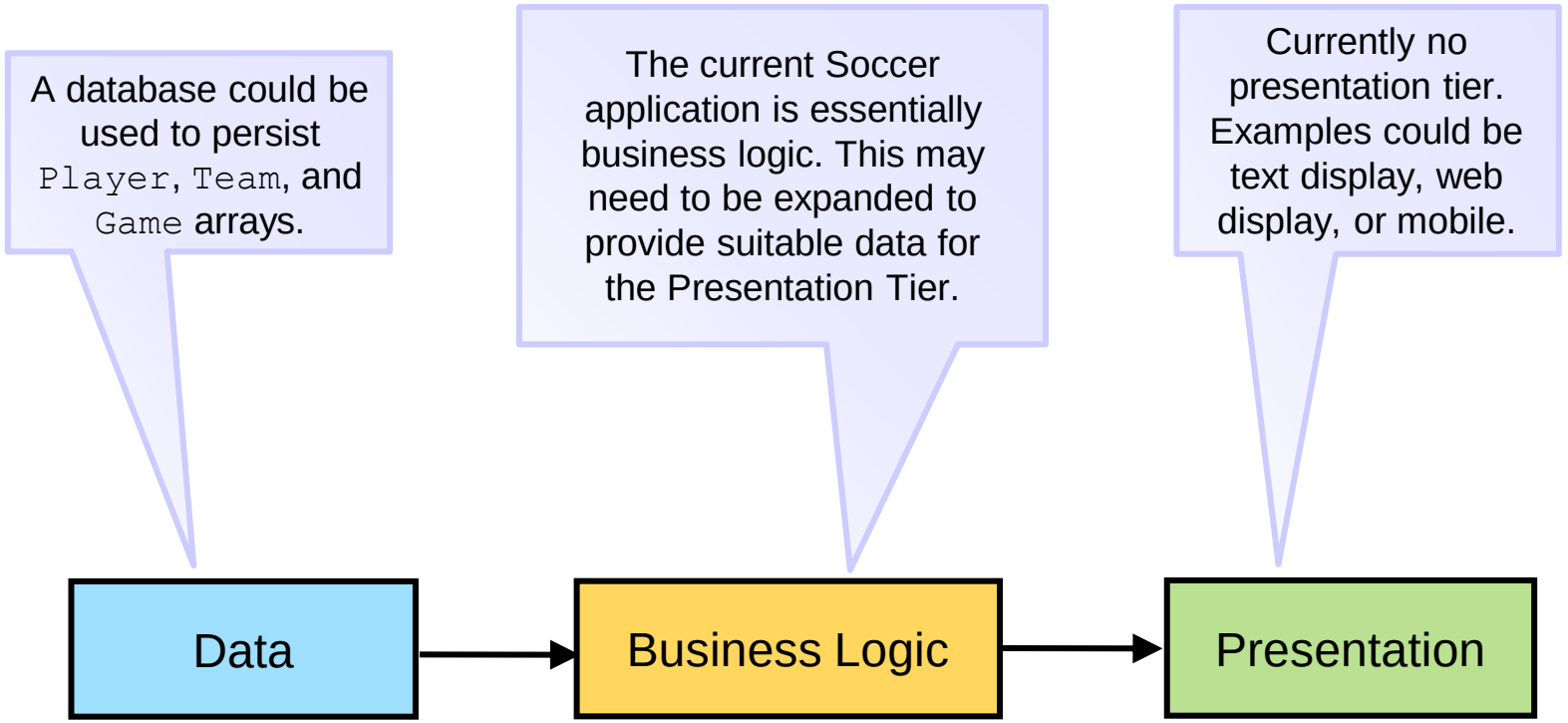
Client/Server Three-Tier Architecture

- Three-tier client/server is a more complex, flexible approach.
- Each tier can be replaced by a different implementation:
  - The data tier is an encapsulation of all existing data sources.
  - Business logic defines business rules.
  - Presentation can be GUI, web, smartphone, or even console.

Topics

- Packages
- JARs and deployment
- Two-tier and three-tier architecture
- The Soccer application
- Application modifications and enhancements

Client/Server Three-Tier Architecture



Client/Server Three-Tier Architecture

Business Logic

	The Robins	The Hens	The Owls	The Swallows	The Crows	Points	Goals
The Robins	X	2 - 2	2 - 2	2 - 1	3 - 3	9	19
The Hens	2 - 3	X	2 - 2	2 - 2	3 - 2	9	17
The Owls	3 - 3	1 - 1	X	2 - 2	2 - 2	9	16
The Swallows	2 - 2	2 - 2	1 - 2	X	2 - 2	7	14
The Crows	3 - 2	2 - 3	2 - 2	1 - 2	X	6	17

		Away Teams						
		The Pelicans	The Sparrows	The Robins	The Crows	The Magpies	Points	Goals
Home Teams	The Pelicans	X	3 - 2	1 - 2	3 - 2	2 - 1	12	19
	The Sparrows	3 - 3	X	3 - 2	2 - 2	3 - 2	9	18
	The Robins	2 - 3	2 - 2	X	3 - 2	1 - 2	8	16
	The Crows	1 - 2	1 - 2	2 - 2	X	1 - 0	6	13
	The Magpies	2 - 2	2 - 1	1 - 2	1 - 2	X	5	11

Different Outputs

A two-dimensional `String` array could provide the `String` output for each element of the grid, but this is inflexible:

- The presentation can only display the `String` provided.
- The presentation cannot access other useful information—for example, the data required to allow users to click on the score for more details.

	The Robins	The Hens	The Owls	The Swallows	The Crows	Points	Goals
The Robins	X	2 - 2	2 - 2	2 - 1	3 - 3	9	19
The Hens	2 - 3	X	2 - 2	2 - 2	3 - 2	9	17
The Owls	3 - 3	1 - 1	X	2 - 2	2 - 2	9	16
The Swallows	2 - 2	2 - 2	1 - 2	X	2 - 2	7	14
The Crows	3 - 2	2 - 3	2 - 2	1 - 2	X	6	17

		Away Teams						
		The Pelicans	The Sparrows	The Robins	The Crows	The Magpies	Points	Goals
Home Teams	The Pelicans	X	3 - 2	1 - 2	3 - 2	2 - 1	12	19
	The Sparrows	3 - 3	X	3 - 2	2 - 2	3 - 2	9	18
	The Robins	2 - 3	2 - 2	X	3 - 2	1 - 2	8	16
	The Crows	1 - 2	1 - 2	2 - 2	X	1 - 0	6	13
	The Magpies	2 - 2	2 - 1	1 - 2	1 - 2	X	5	11

The Soccer Application

- Abstract classes
  - `GameEvent`
    - Extended by `Goal` and other `GameEvent` classes
- Interfaces
  - `Comparable`
    - Implemented by `Team` and `Player` so that they can be ranked
  - `IDisplayDataItem`
    - Implemented by `Team`, `Game`, and `DisplayString`

IDisplayDataItem Interface

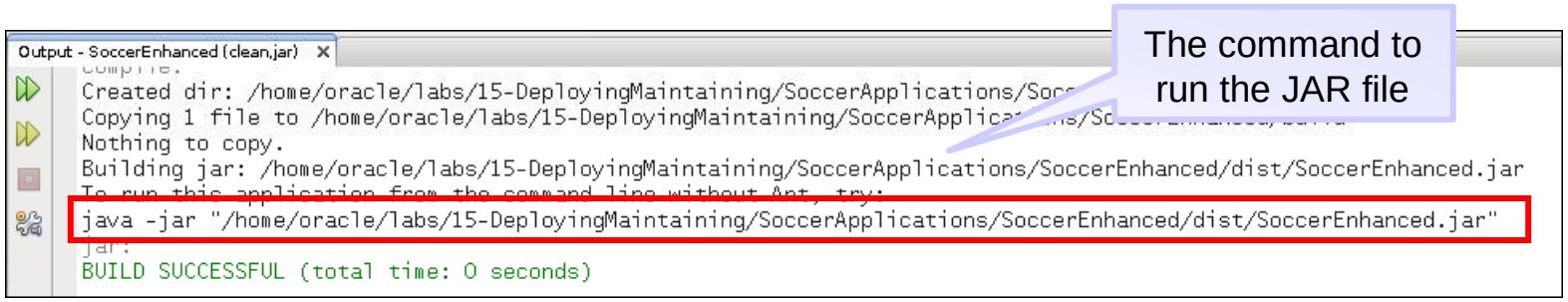
```
package soccer.play;

public interface IDisplayDataItem {

    public boolean isDetailAvailable ();
    public String getDisplayDetail();
    public int getID();
    public String getDetailType();

}
```

Running the JAR File from the Command Line




```
[oracle@EDBSR2P14 ~]$ java -jar "/home/oracle/labs/15-
      DeployingMaintaining/SoccerApplications/SoccerEnhanced/dist/SoccerEnhanced.jar"
-------------------------------------------------------------------------------------------
|              | The Robins | The Hens | The Owls | The Swallows | The Crows | Points | Goals |
-------------------------------------------------------------------------------------------
| The Robins   |  X         | 2 - 2    | 2 - 2    | 2 - 1        | 3 - 3     | 9      | 19    |
-------------------------------------------------------------------------------------------
| The Hens     | 2 - 3      |  X       | 2 - 2    | 2 - 2        | 3 - 2     | 9      | 17    |
-------------------------------------------------------------------------------------------
| The Owls     | 3 - 3      | 1 - 1    |  X       | 2 - 2        | 2 - 2     | 9      | 16    |
-------------------------------------------------------------------------------------------
| The Swallows | 2 - 2      | 2 - 2    | 1 - 2    |  X           | 2 - 2     | 7      | 14    |
-------------------------------------------------------------------------------------------
| The Crows    | 3 - 2      | 2 - 3    | 2 - 2    | 1 - 2        |  X        | 6      | 17    |
-------------------------------------------------------------------------------------------
```

Text Presentation of the League

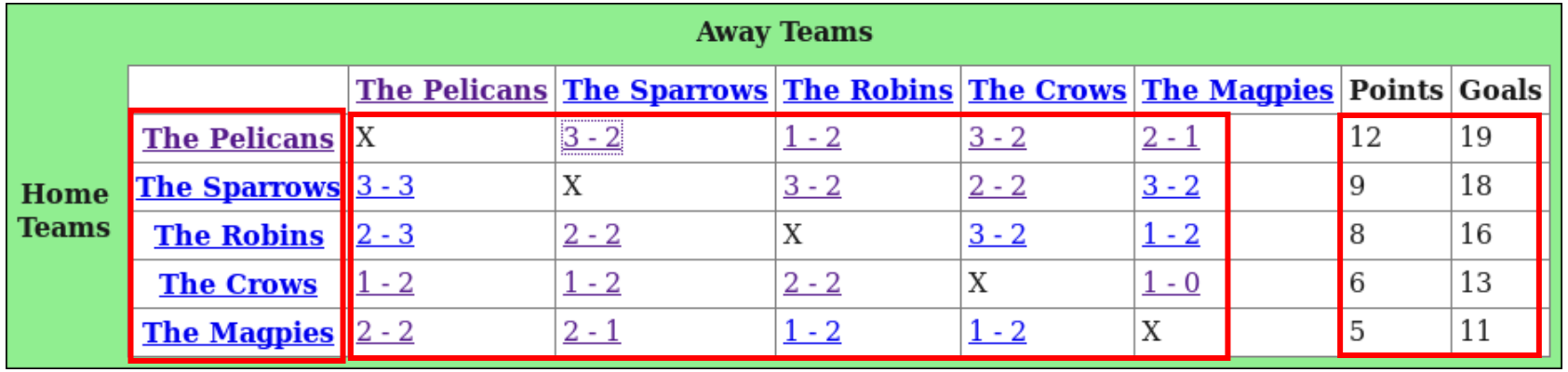
```
[oracle@EDBSR2P14 ~]$ java -jar "/home/oracle/labs/15-
      DeployingMaintaining/SoccerApplications/SoccerEnhanced/dist/SoccerEnhanced.jar"
-------------------------------------------------------------------------------------------
|              | The Robins | The Hens | The Owls | The Swallows | The Crows | Points | Goals |
-------------------------------------------------------------------------------------------
| The Robins   |  X         | 2 - 2    | 2 - 2    | 2 - 1        | 3 - 3     | 9      | 19    |
-------------------------------------------------------------------------------------------
| The Hens     | 2 - 3      |  X       | 2 - 2    | 2 - 2        | 3 - 2     | 9      | 17    |
-------------------------------------------------------------------------------------------
| The Owls     | 3 - 3      | 1 - 1    |  X       | 2 - 2        | 2 - 2     | 9      | 16    |
-------------------------------------------------------------------------------------------
| The Swallows | 2 - 2      | 2 - 2    | 1 - 2    |  X           | 2 - 2     | 7      | 14    |
-------------------------------------------------------------------------------------------
| The Crows    | 3 - 2      | 2 - 3    | 2 - 2    | 1 - 2        |  X        | 6      | 17    |
-------------------------------------------------------------------------------------------
```

The object type behind these data elements is `Team`.

The object type behind these data elements (except for the output Xs) is `Game`.

The object type behind these data elements is `DispayString`.

Web Presentation of the League



Home Teams \ Away Teams	The Pelicans	The Sparrows	The Robins	The Crows	The Magpies	Points	Goals
The Pelicans	X	3 - 2	1 - 2	3 - 2	2 - 1	12	19
The Sparrows	3 - 3	X	3 - 2	2 - 2	3 - 2	9	18
The Robins	2 - 3	2 - 2	X	3 - 2	1 - 2	8	16
The Crows	1 - 2	1 - 2	2 - 2	X	1 - 0	6	13
The Magpies	2 - 2	2 - 1	1 - 2	1 - 2	X	5	11

The object type behind these data elements is `Team`.

The object type behind these data elements (except for the output Xs) is `Game`.

The object type behind these data elements is `DispayString`.

Topics

- Packages
- JARs and deployment
- Two-tier and three-tier architecture
- The Soccer application
- Application modifications and enhancements

Enhancing the Application

- Well-designed Java software minimizes the time required for:
  - Maintenance
  - Enhancements
  - Upgrades
- For the Soccer application, it should be easy to:
  - Add new `GameEvent` subclasses (business logic)
  - Develop new clients (presentation)
    - Take the application to a smartphone (for example)
  - Change the storage system (data)

Adding a New `GameEvent Kickoff`

It is possible to add a new `GameEvent` to record kickoffs by:

- Creating a new `Kickoff` class that extends the `GameEvent` class
- Adding any new unique features for the item
- Modifying any other classes that need to know about this new class

Game Record Including `Kickoff`

The Magpies vs. The Sparrows (2 - 3)			
Event	**Team**	**Player**	**Time**
Kickoff	The Sparrows	Dorothy Parker	0
Possession	The Sparrows	Jane Austin	15
Possession	The Sparrows	J. M. Synge	19
Possession	The Sparrows	Brendan Behan	20
Possession	The Sparrows	Dorothy Parker	26
GOAL!	The Sparrows	Dorothy Parker	32
Kickoff	The Magpies	G. K. Chesterton	34
Possession	The Magpies	Oscar Wilde	35
Possession	The Magpies	G. K. Chesterton	41
GOAL!	The Magpies	G. K. Chesterton	43
Kickoff	The Sparrows	Dorothy Parker	50
Possession	The Sparrows	J. M. Synge	54
GOAL!	The Sparrows	J. M. Synge	55
Kickoff	The Magpies	Wilkie Collins	59
Possession	The Magpies	G. K. Chesterton	62
Possession	The Magpies	Arthur Conan Doyle	63
Possession	The Magpies	Oscar Wilde	64
GOAL!	The Magpies	Oscar Wilde	74
Kickoff	The Sparrows	Frank O'Connor	75
Possession	The Sparrows	Frank O'Connor	81
GOAL!	The Sparrows	Frank O'Connor	83

The new event, `Kickoff`, has been added.

Summary

In this lesson, you should have learned how to:

- Deploy a simple application as a JAR file
- Describe the parts of a Java application, including the user interface and the back end
- Describe how classes can be extended to implement new capabilities in the application