3

Creating a Java Main Class

Objectives

After completing this lesson, you should be able to:

- Use the NetBeans IDE to create and test Java classes
- Write a `main` method
- Use `System.out.println` to write a String literal to system output

Topics

- Java classes and packages
- The `main` method

Java Classes

A Java class is the building block of a Java application.

Includes code that:

- Allows a customer to add items to the shopping cart
- Provides visual confirmation to the customer

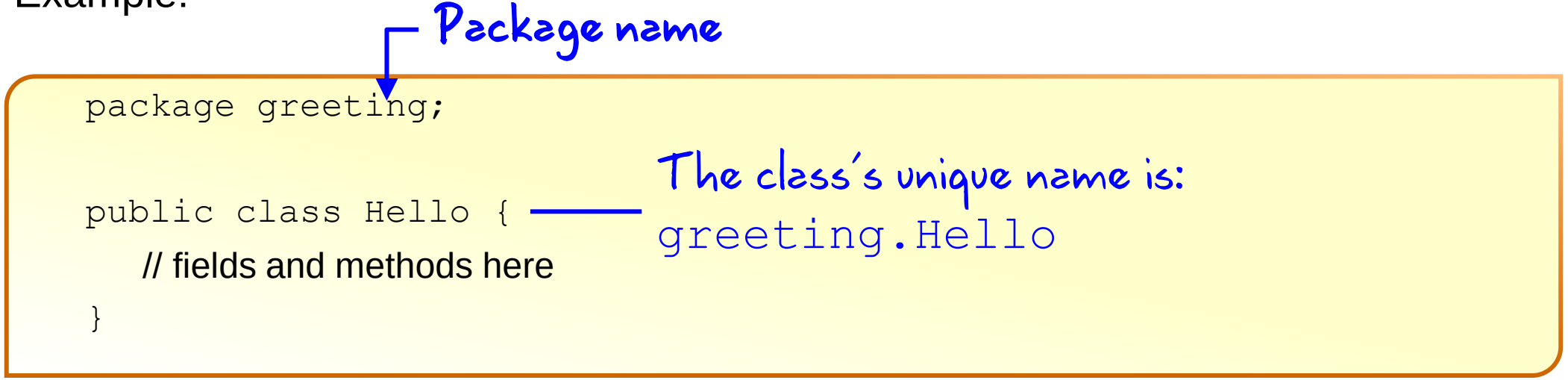
Program Structure

- A class consists of:
  - The class name. Class names begin with a capital letter.
  - The body of the class surrounded with braces { }
    - Data (called fields)
    - Operations (called methods)
- Example:


```
public class Hello {
   // fields of the class
   // methods
}
```

Java Packages

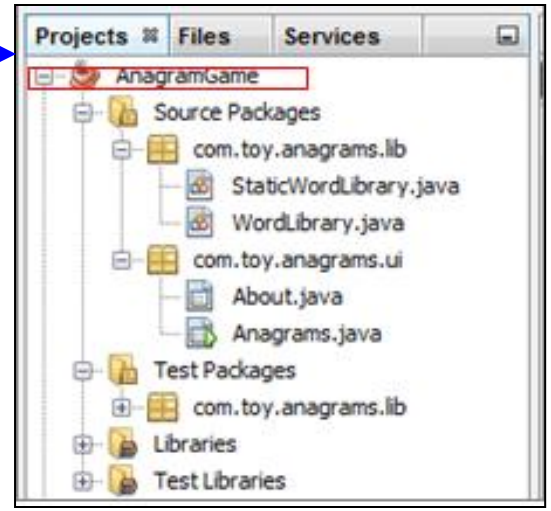
- A package provides a namespace for the class.
  - This is a folder in which the class will be saved.
  - The folder name (the package) is used to uniquely identify the class.
  - Package names begin with a lowercase letter.
- Example:



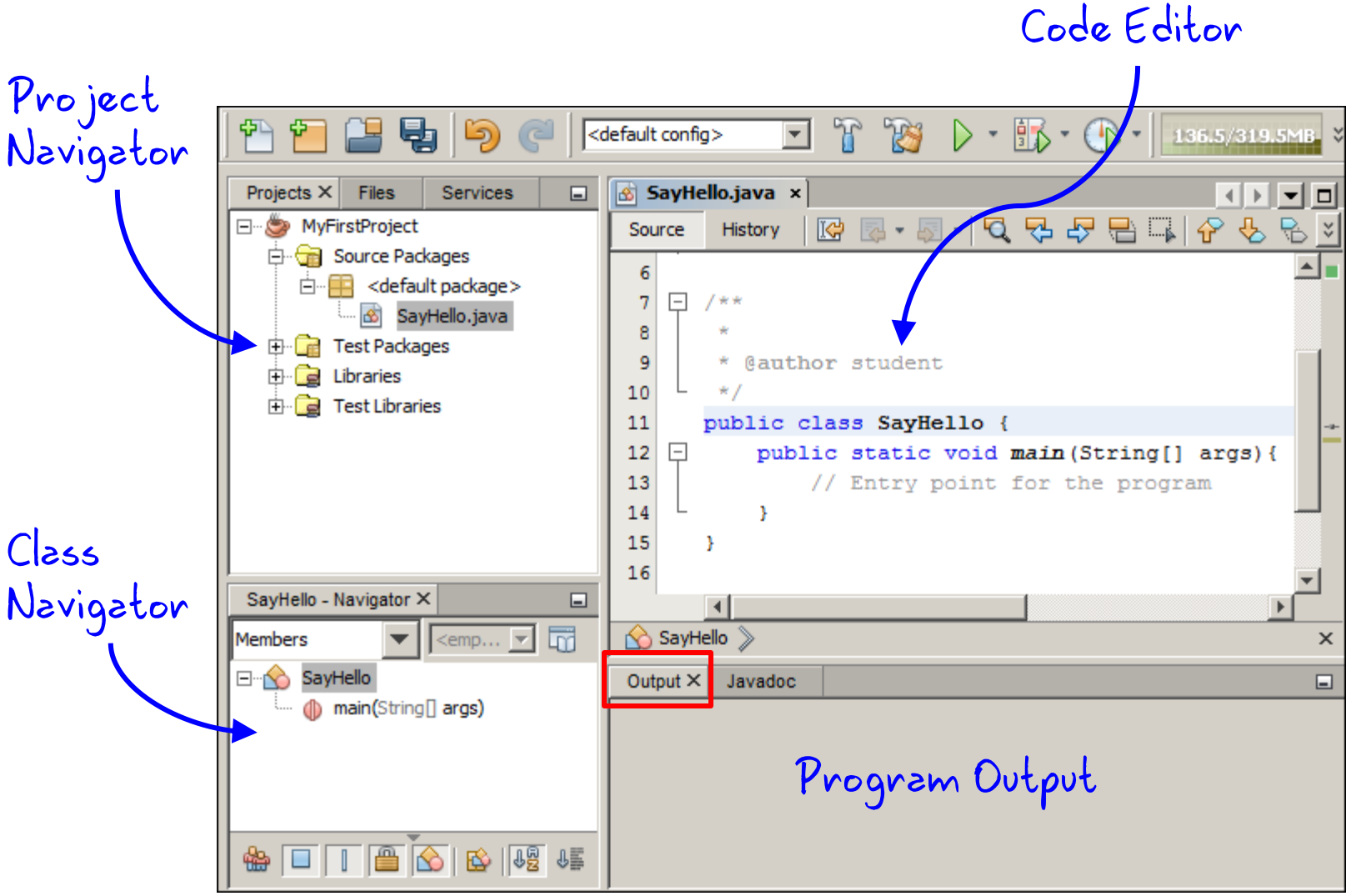
Java IDEs

A Java Integrated Development Environment (IDE) is a type of software that makes it easier to develop Java applications.

- An IDE provides:
  - Syntax checking
  - Various automation features
  - Runtime environment for testing
- It enables you to organize all your Java resources and environment settings into a *Project*.
- Projects contain packages.
- Packages contain files, such as `.java`.

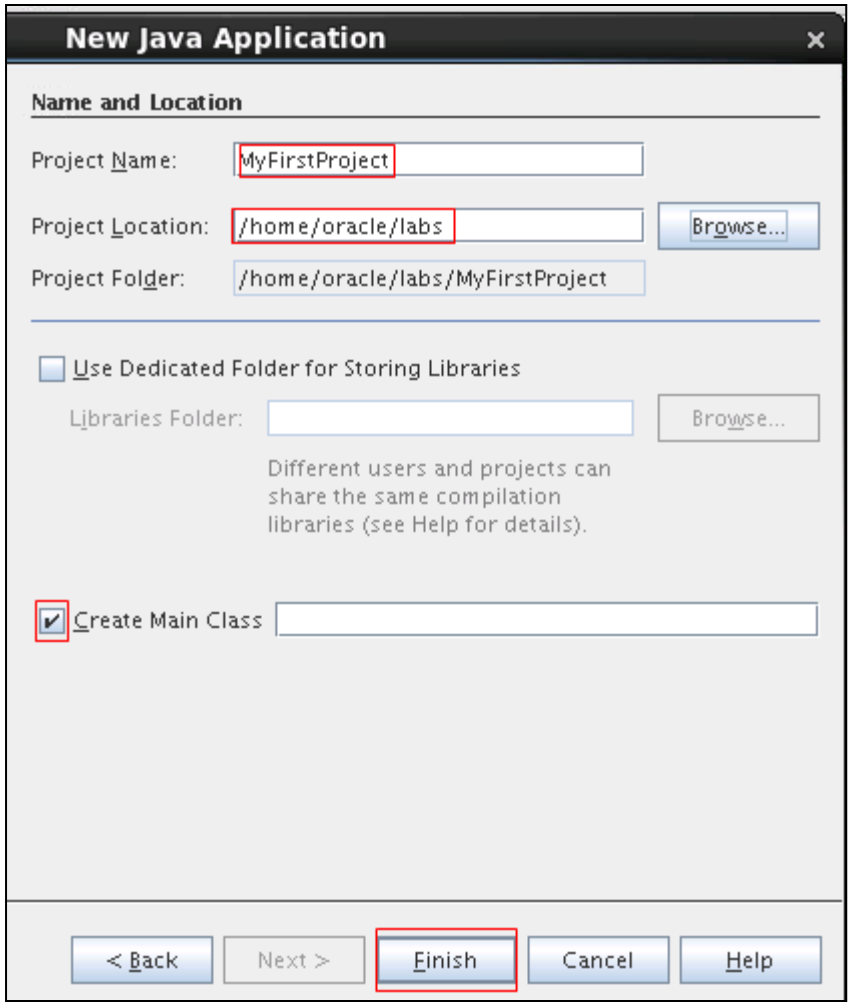


The NetBeans IDE



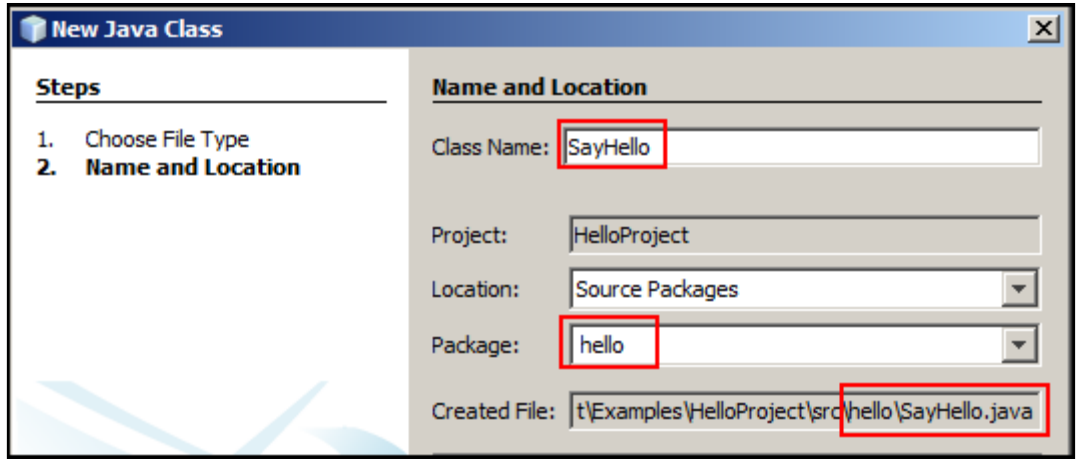
Creating a Java Project

1. Select **File > New Project**.
2. Select Java Application.
3. Name and set the location for the project.
4. Select “Create Main Class” if you want it done for you automatically.
5. Click **Finish**.



Creating a Java Class

1. Select **File > New File**.
2. Select your project and choose **Java Class**.
3. Name the class.
4. Assign a package.
5. Click **Finish**.



Exercise 3-1:Creating a New Project and Java Class

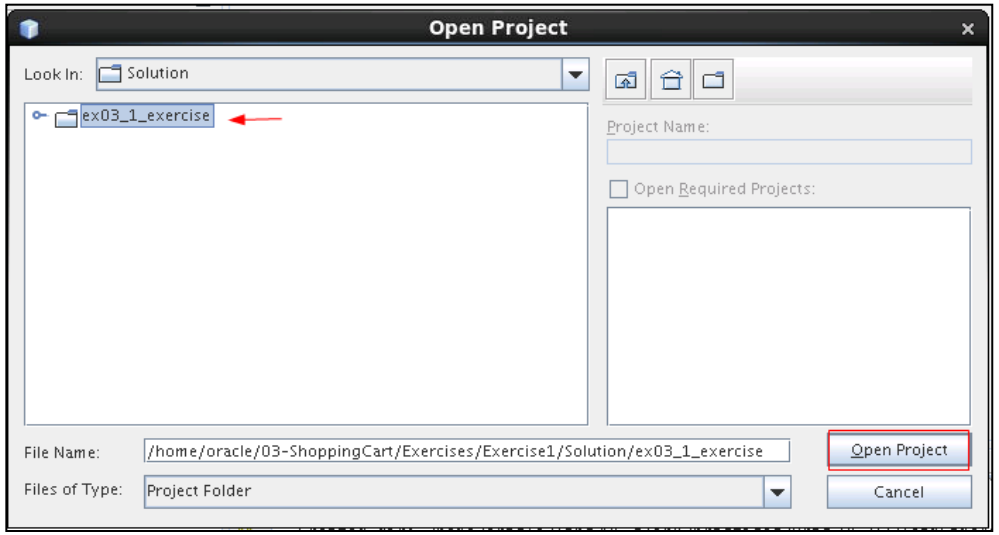
In this exercise, you use NetBeans to create a new Java Class.

1. Create a new project called **`Exercise_03-1`**.
   - Deselect the box to create the `main` method. You will write the `main` method yourself in the next exercise.
2. Create a new **Java Class** file in this project.
   - Class name = `ShoppingCart`
   - Package name =`exercise`

Opening an Existing Java Project

If you need to open an existing project in NetBeans, perform the following steps:

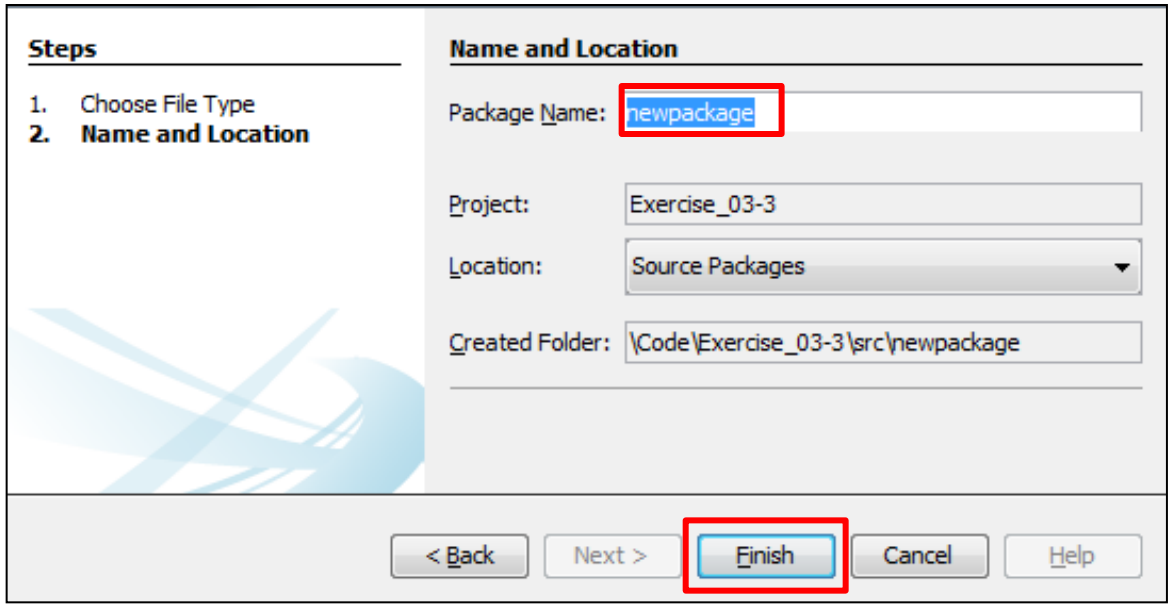
1. Select **File > Open Project.**
2. Navigate to the directory that contains your projects.
3. Select the project file you want. (This file must be unzipped.)
4. Click **Open Project**.



Creating a New Java Package

If you ever need to create a new package, perform the following steps in NetBeans:

1. Right-click your project.
2. Select **New > Java Package**.
3. Name the package.
4. Click **Finish**.



Topics

- Java classes and packages
- The `main` method

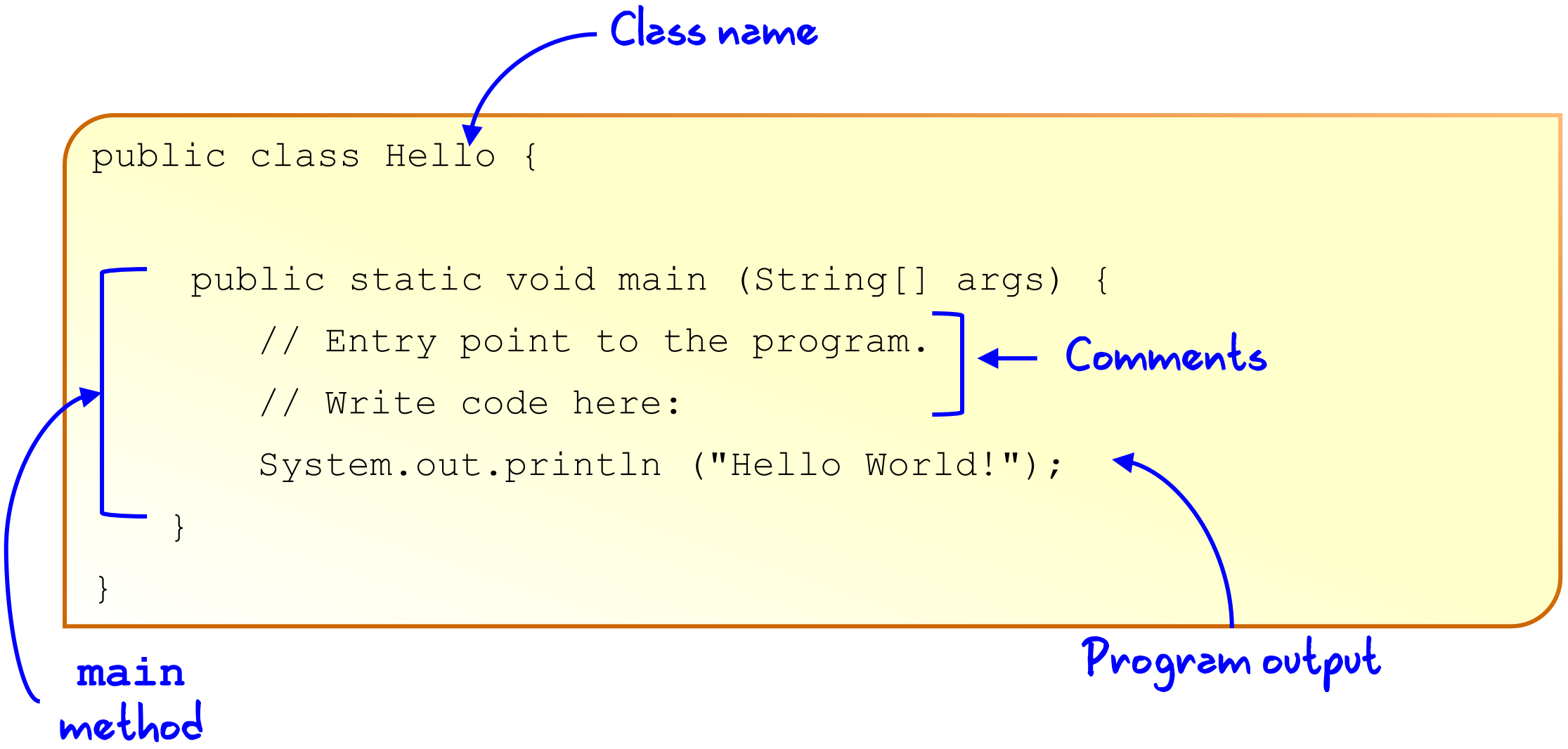
The `main` Method

- It is a special method that the JVM recognizes as the starting point for every Java program.
- The syntax is always the same:

```
public static void main (String[] args) {
   // code goes here in the code block
}
```

- It surrounds entire method body with braces { } .

A `main` Class Example

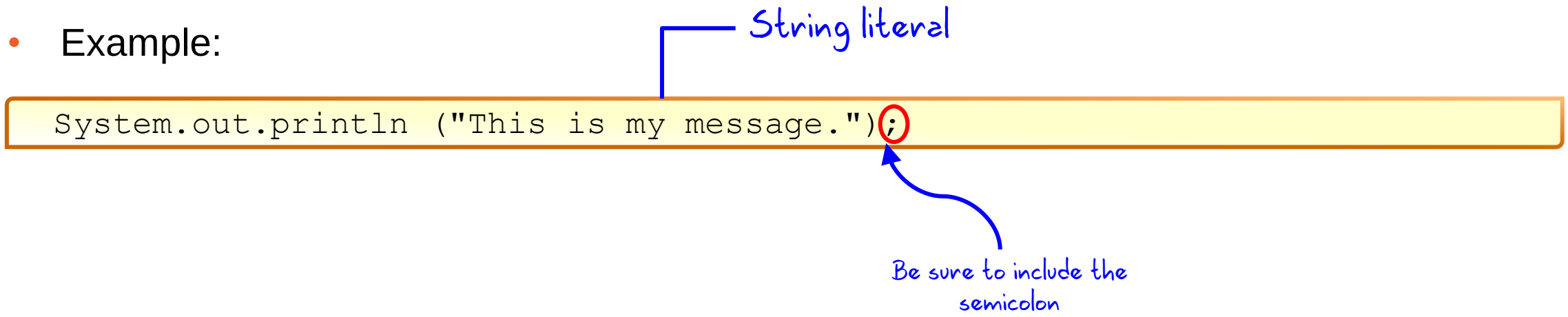


Output to the Console

- Syntax:

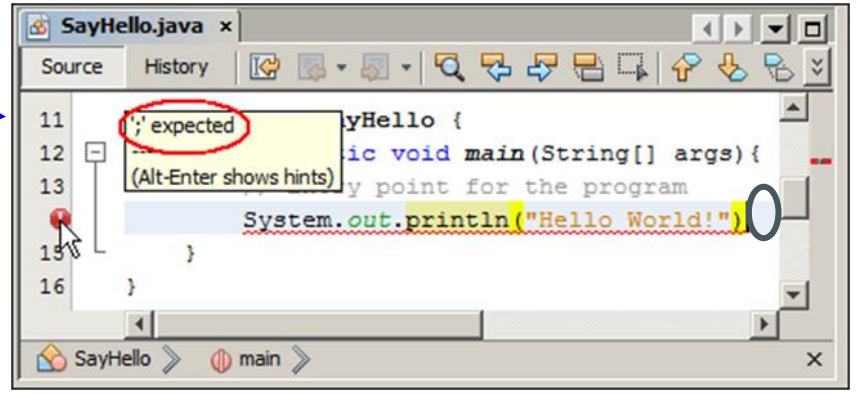
```
System.out.println (<some string value>);
```

- Example:

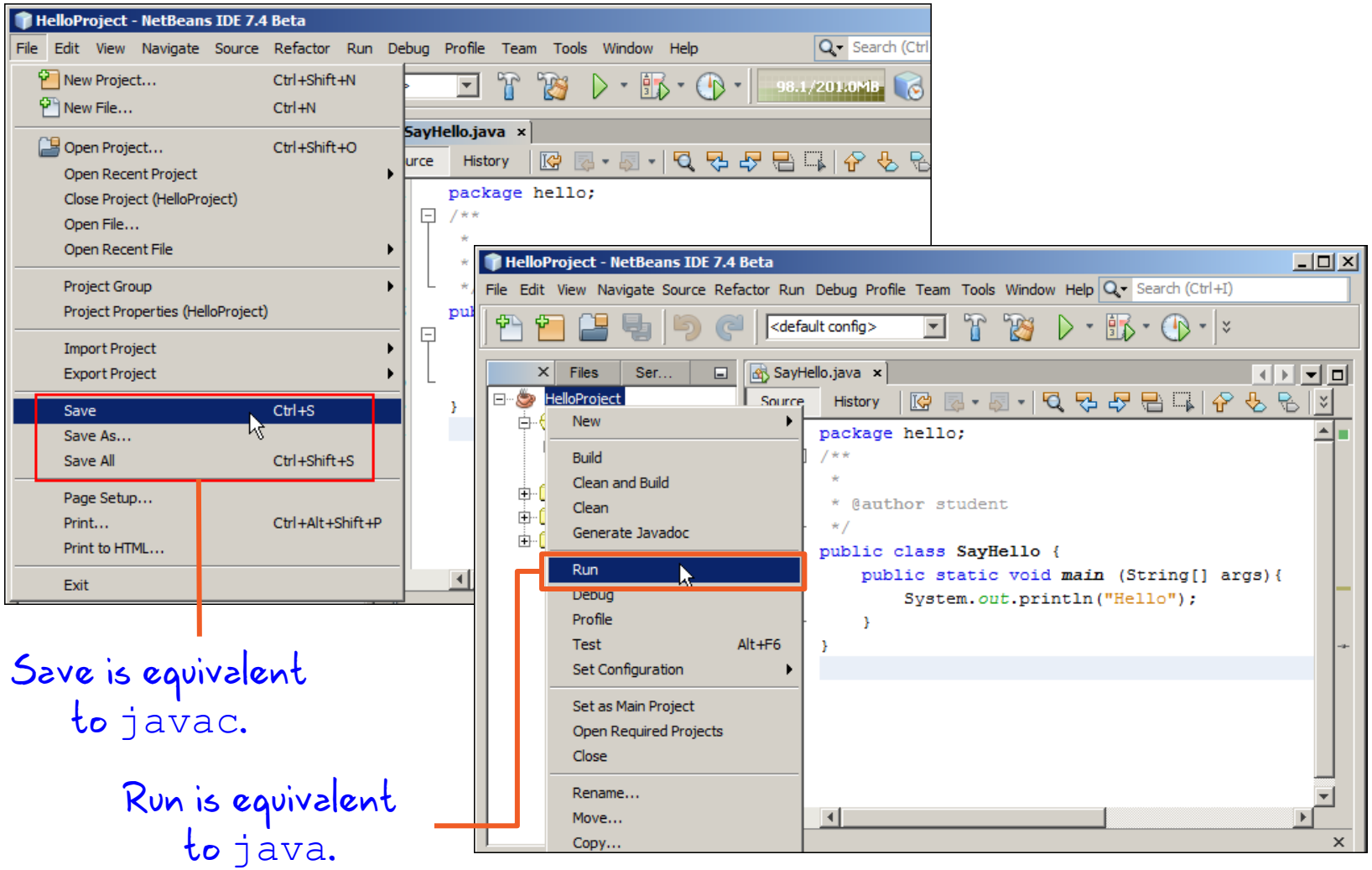


Avoiding Syntax Errors

- NetBeans will tell you if you have done something wrong.
- Common errors include:
  - Unrecognized word (check for case-sensitivity error)
  - Missing close quotation mark
  - Unmatched brace
  - Missing semicolon

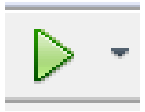


Compiling and Running a Program by Using NetBeans



Exercise 3-2: Creating a `main` Method

In this exercise, you manually enter a `main` method that prints a message to the console.

1. Continue editing **`Exercise_03-1`** or open **`Exercise_03-2`**.
2. In the code editor, add the `main` method structure to the ShoppingCart class.
3. In the code block of the `main` method, use a `System.out.println` method to print "Welcome to the Shopping Cart!"
4. Save your program.
5. Click the **Run** button to test program. 
   - Select **`exercise.ShoppingCart`** as the main class.

Quiz

Which `main` method syntax is correct?

a. `Public static void main (String[ ] args){ }`

b. `public Static void Main (String[ ] args){ }`

c. `public static void main (String ( ) args)[ ]`

d. `public static void main (String[ ] args){ }`

Summary

In this lesson, you should have learned how to:

- Use the NetBeans IDE to create and test Java classes
- Write a `main` method
- Use `System.out.println` to write a String literal to system output