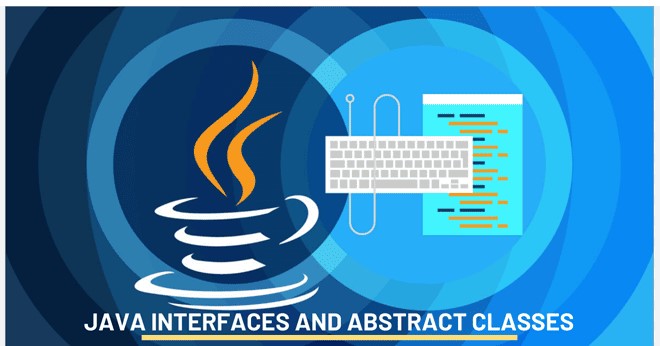
2025

Steve Stylin

Bellevue University

2/20/2025

Module 10 Introduction: Abstract Classes and Interfaces



The assignment requires the creation of an abstract class named Division. With specific fields and methods, along with two subclasses: InternationalDivision.java and DomesticDivision.java. Additionally, an application named UseDivision.java must be created to instantiate these classes. The question also highlights an error related to access modifiers in Java.

To address the requirements, we will create an abstract class division that contains fields for the division name and account number, as well as an abstract method display(). The subclasses will extend this class and implement the display() method while adding their specific fields

Let’s create the Divion.java class:

A screenshot of a computer program

AI-generated content may be incorrect.

We have created the *InternationalDivision.java* to represent an international division.

A computer screen shot of a program code

AI-generated content may be incorrect.

*The DomesticDivision.java* class is a domestic division.

A computer screen shot of a program

AI-generated content may be incorrect.

The last step is creating the UseDivision.java Application, which demonstrates the use of Division classes.

A screen shot of a computer code

AI-generated content may be incorrect.

will compile all the classes in one go. The Java compiler (javac) will resolve dependencies automatically.



A computer screen shot of white text

AI-generated content may be incorrect.

The errors encountered during compilation are due to the access modifiers of the fields in the Division class. The fields divisionName and accountNumber are declared as private, which means they cannot be accessed directly in the subclasses. To resolve this issue, we will change the access modifiers in Division.java from private to protected, allowing subclasses to access them.

A screenshot of a computer program

AI-generated content may be incorrect.

Let us recompile the Java files after modifying Division.java 

A black screen with white text

AI-generated content may be incorrect.

The Java files were compiled successfully. Now, we can execute the main method in the UseDivision.java application, creating instances of the divisions and displaying their details.

Output:

