COURSEWARE

Packages

Contents

Overview

A package in Java is used to group related classes together.
Packages help us to write more maintainable code and can be divided into two categories;

- Built in packages which come from the Java API.
- User defined packages which are packages created by us.

The use of packages can also help provide controlled access to certain classes through the use of the protected and default access modifiers.

Built-in Packages

The Java API is a library of classes that contain components for input, database programming and much more.
The library is divided into packages, and has the classes within those packages, this allows us to import a single class or an entire package if we wanted to.

```
import java.util.Scanner;
import java.sql.*;
```

In the above example, the first import is only importing the scanner class from the java.util package; whereas the second import is importing the entire sql package.
Importing an entire package is generally considered bad practice due to the fact that we could end up importing classes that we don't actually need, so it is better to have multiple imports that get the specific classes we need.

User Defined Packages

We can define our own packages to group relevant classes together, Java uses a file system directory to store them, just like folders.

```
root
    -> package1
        -> Class1.java
    -> package2
        -> Class2.java
        -> Class3.java
```

Package Naming Convention

When naming packages there is a standard naming convention that we follow.
Firstly there should be no capital letters in the package name, the package should be entirely lower case.
Secondly the package name should follow this format:

```
package domain.companyname.packagename;
```

The package keyword at the beginning simply tells Java that we are defining the package name that the class resides within.
Generally companies use the reverse of their internet domain name.
So if we were to create a package for the controller section of a project named "ims" we would have the following package name.

```
package com.qa.ims.controller;
```

If we now wanted to import a class named `CustomerController` that resides within the controller package into a class that is in a different package we can use the following import.

```
import com.qa.ims.controller.CustomerController;
```

Java requires package definitions (if they exist) to be at the **top** of the file - before any imports or the class definition.

Tutorial

There is no tutorial for this exercise.

Exercises

There are no exercises for this module.