

Packages & Interfaces

What are packages?

Containers for class.

A mechanism to group similar types of classes together.

Helps us to provide access control.

Built-in java packages: `java.lang`, `java.io`, `java.util`.



How a package looks like



Advantages of using packages

Code reuse: a single package has multiple classes.

Avoids naming collision when multiple packages have classes with same name.

Allow hiding of classes.

Allow us to use access modifiers.



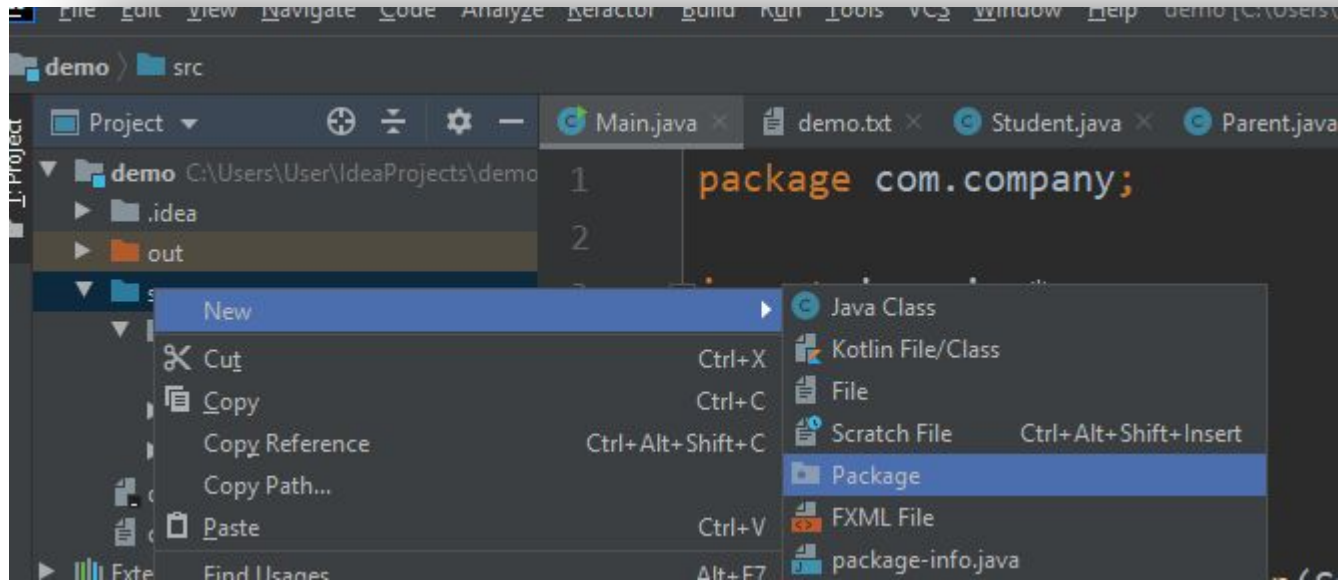
Types of packages

Built in: `java.lang`, `java.io`, `java.util`.

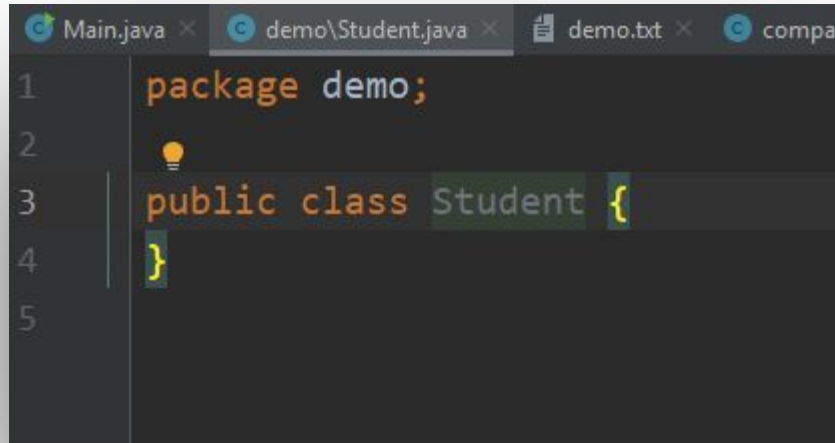
User defined: Created by us developers to hold our classes.



How to create a package?



How a package is defined



The screenshot shows an IDE with four tabs: 'Main.java', 'demo\Student.java', 'demo.txt', and 'compa'. The 'demo\Student.java' tab is active, displaying the following code:

```
1 package demo;  
2  
3 public class Student {  
4 }  
5
```

The code defines a package named 'demo' and a public class named 'Student' within that package. The class definition is currently empty, showing only the opening curly brace. A lightbulb icon is visible above the class definition, indicating a suggestion or tip.

Let's create a package.....

