Java Packages

Topics : JAVA
Written on April 10, 2023

Java packages are a way of organizing related classes, interfaces, and sub-packages in a hierarchical structure. They help to avoid naming conflicts, improve code organization, and facilitate code reuse.

A package is simply a container for classes and other packages. Packages in Java are created using the `package` keyword at the beginning of a Java source file.

For example, if you have a set of classes that are related to networking, you might create a package called `network`. The package declaration for this package would look like this:

```
package network;
```

This declaration would go at the top of each source file in the `network` package.

Packages are usually organized in a hierarchical structure, where sub-packages contain more specific classes and interfaces. For example, you might have a `network.tcp` package that contains classes related to the TCP protocol, and a `network.udp` package that contains classes related to the UDP protocol.

To use classes from a package, you need to import them into your source file using the `import` statement. For example, to use the `Socket` class from the `java.net` package, you would import it like this:

```
import java.net.Socket;
```

This statement would allow you to use the `Socket` class in your code without having to use the fully qualified name (`java.net.Socket`) every time.

In summary, packages are a way of organizing related classes, interfaces, and sub-packages in Java. They help to avoid naming conflicts, improve code organization, and facilitate code reuse.

Here's an example of how to create and use a package in Java:

Let's say we have two classes called `Calculator` and `Main` that we want to organize in a package called `com.mycompany.math`. Here's how we would do that:

First, we create a directory called `com/mycompany/math` in our project's source directory (e.g., `src/main/java`).

Then, we create two Java files in that directory, one for each class:

```java
package com.mycompany.math;

public class Calculator {
    public int add(int a, int b) {
        return a + b;
    }

    public int subtract(int a, int b) {
        return a - b;
    }
}
```

```java
package com.mycompany.math;

public class Main {
    public static void main(String[] args) {
        Calculator calc = new Calculator();
        int result = calc.add(2, 3);
        System.out.println("2 + 3 = " + result);
    }
}
```

Note that both classes have the same package declaration (`package com.mycompany.math;`), which specifies that they belong to the `com.mycompany.math` package.

Now, if we want to use the `Calculator` class in another Java file (e.g., `MyApp.java`) outside of the `com.mycompany.math` package, we need to import it using the `import` statement:

```java
import com.mycompany.math.Calculator;

public class MyApp {
    public static void main(String[] args) {
        Calculator calc = new Calculator();
        int result = calc.add(4, 5);
        System.out.println("4 + 5 = " + result);
    }
}
```

Here, we import the `Calculator` class from the `com.mycompany.math` package using the

statement `import com.mycompany.math.Calculator;`. Now we can create an instance of the `Calculator` class and use its `add()` method in our code.