



Object Oriented Programming with Java

(Subject Code: BCS-403)

Unit 1

Lecture 12

Lecture 12

- Import and Static Import
- Naming Convention for Packages
- Making JAR Files for Library Packages

Static import in Java

- In Java, static import concept is introduced in 1.5 version.
- With the help of static import, we can access the static members of a class directly without class name or any object.
- For Example: we always use `sqrt()` method of `Math` class by using `Math` class i.e. **`Math.sqrt()`**, but by using static import we can access `sqrt()` method directly.

With Static import

```
import static java.lang.Math.*;
class Test2 {
    public static void main(String[] args)
    {
        System.out.println(sqrt(4));
        System.out.println(pow(2, 2));
        System.out.println(abs(6.3));
    }
}
```

Ambiguity in static import

If two static members of the same name are imported from multiple different classes, the compiler will throw an error, as it will not be able to determine which member to use in the absence of class name qualification

Ambiguity in case of static import

```
package MyPackage;
import static java.lang.Integer.*;
import static java.lang.Byte.*;
public class MyMain {
    public static void main(String[] args)
    {
        System.out.println(MAX_VALUE);
    }
}
```

Error: Reference to MAX_VALUE is ambiguous

Difference between import and static import:

- With the help of import, we are able to access classes and interfaces which are present in any package. But using static import, we can access all the static members (variables and methods) of a class directly without explicitly calling class name.
- The main difference is Readability, `ClassName.dataMember (System.out)` is less readable when compared to `dataMember(out)`
- static import can make your program more readable

Naming Conventions

- Package names are written in all lower case to avoid conflict with the names of classes or interfaces.
- Companies use their reversed Internet domain name to begin their package names.

For example, `com.example.mypackage` for a package named `mypackage` created by a programmer at `example.com`.

- Name collisions that occur within a single company need to be handled by convention within that company, perhaps by including the region or the project name after the company name.

for example, `com.example.region.mypackage`).

Making Jar Files

- In Java, JAR stands for Java Archive, whose format is based on the zip format.
- The JAR files format is mainly used to aggregate a collection of files into a single one.
- It is a single cross-platform archive format that handles images, audio, and class files. With the existing applet code, it is backward-compatible.
- In Java, Jar files are completely written in the Java programming language.
- We can either download the JAR files from the browser or can write our own JAR files using Eclipse IDE.