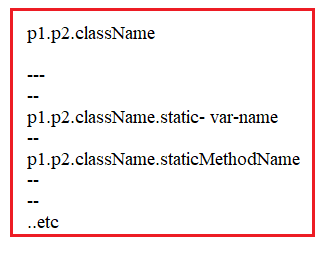
We can access the package class from another package class in two ways:

1. Using Fully Qualified Name.
2. Using import keyword.

Drawback of Fully Qualified Name:- The package class and its members can be accessed in other package class using fully qualified name. The code readability is decreased and Duplicate code is grown in program by fully qualified name.

Example:



To overcome this drawback, Java introduced “import” keyword.

Syntax:1



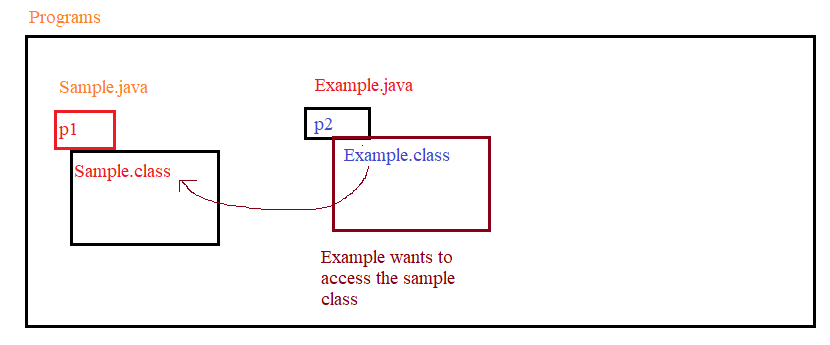
The all Classes , interfaces and enums which resides in imported package are available in current program.

Syntax:2

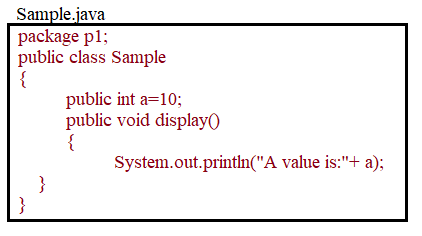


The particular class is only available in current program.

Example:1



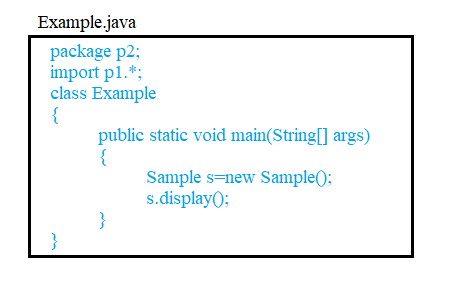
1.



2. Compile Sample.java



3.

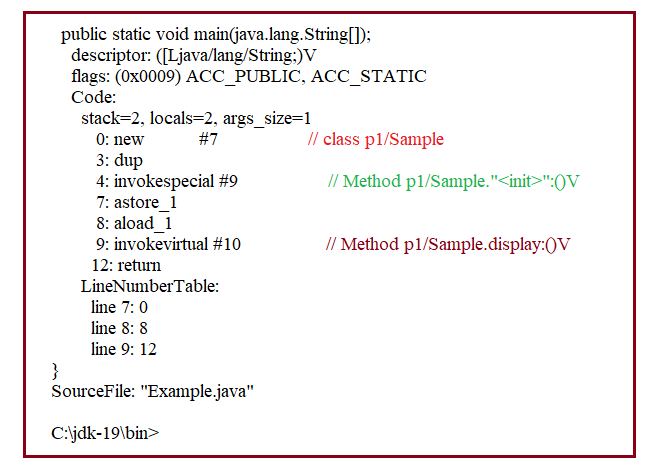


4. compile Example.java

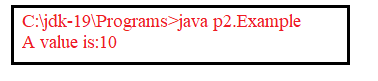


Note: During compilation, Java compiler removes all import statements and attaches the (packagename .) to their classes. Verify these changes by

Javap -v filename.class.



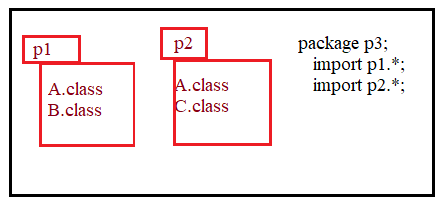
5. Run Example program.



Q) when should we use Fully Qualified Name?

If the class is presented in more than one imported package then to avoid naming conflict we must use fully qualified name.

Example:



When program access class A , compiler will show error because it got ambiguity. To avoid ambiguity, we should use fully qualified name.

**1.Static Import:** in java 5 version, we got new feature which is static import. By using this new feature, we can access static members of one packaged class from other packaged class with out using class Name.

Syntax:





Example:

import static java.lang.System.out;

class sample

{

public static void main(String[] args)

{

out.println("Demonstration of static import");

}

}

