**Packages** in java – is a grouping of related classes and interfaces into a single unit provides access control and namespace management.

It is recommended to combine related classes and interfaces into packages to make them easy to locate and use, prevent naming conflicts and control access.

The java platform classes and interfaces are members og numerous packages that group classes by function: fundamental classes are in java.lang, classes for reading and writing (I/O) are in java.io, classes for sql operations are in java.sql.

As a result, we may conclude that packages serve as an encapsulation mechanism for grouping similar classes and interfaces in a single category.

**Need of packages** in java –

1. **Naming conflicts** – since the package creates a new namespace, the names of the classes and interfaces will not conflict with the names of classes and interfaces in other packages. For example – there are two date classes in SQL package and one in util package. So, when using the package, we can distinguish between the java.util and java.sql packages.
2. **Security** – we can provide unrestricted access to classes and interfaces within the package while restricting access to classes and interfaces outside the pacage, which secures our component.
3. **Modularity** & maintainability – since packages are groups of similar classes and interfaces, using them improve the application’s modularity and maintenance.

**Types of packages** in java –

1. Built-in package – is one that is already specified and Is a part of the java API. Java.io, java.util, java.lang, java.sql, java.net, etc
2. User defined packages - which are created by a programmer.

**Creating a package** in java - we can create java package by using package keyword.

package package\_name;

compile and run the java package:

we can compile the file using

javac –d . packageExample.java

we can run the file using

java package.packageExample.java

**Accessing packages**:-

3 ways to access the public members from outside its package

1. Import an entire package
2. Import a package member
3. Preferring to a package member by its fully qualified name