* **Definition:**
  + A *package* is a grouping of related types providing access protection and name space management. Note that *a type refers* to classes, interfaces, enumerations, and annotation types. Enumerations and annotation types are special kinds of classes and interfaces, respectively, so *types* are often referred to in this lesson simply as *classes and interfaces*.
  + Fundamental classes are in java.lang
  + classes for reading and writing (input and output) are in java.io

You should bundle these classes and the interface in a package for several reasons, including the following:

* You and other programmers can easily determine that these types are related.
* You and other programmers know where to find types that can provide graphics-related functions.
* The names of your types won't conflict with the type names in other packages because the package creates a new namespace.
* You can allow types within the package to have unrestricted access to one another yet still restrict access for types outside the package.

**Naming Conventions**

* + Package names are written in all lowercase 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.dwit for a package named dwit 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.company.region.package).
  + Packages in the Java language itself begin with java. or javax.
  + In some cases, the internet domain name may not be a valid package name. This can occur if the domain name contains a hyphen or other special character, if the package name begins with a digit or other character that is illegal to use as the beginning of a Java name, or if the package name contains a reserved Java keyword, such as "int". In this event, the suggested convention is to add an underscore. For example:

### Importing a Package Member

import graphics.Rectangle;

### Importing an Entire Package

import graphics.\*;