Chapter 14 Generics and the ArrayList Class

1. Generics: type parameter enabled coding
   1. applies to any class
2. ArrayList
   1. container that can grow and shrink
   2. has private instance variable: Array
      1. when Array is full, new larger Array is created and data is transferred
   3. less efficient than Array
   4. syntax

//BaseType can not be primitive type

ArrayList<BaseType> aList = new ArrayList<BaseType>();

//intial capacity of 20 items

ArrayList<BaseType> aList = new ArrayList<BaseType>(20);

aList.add(“something”);

int howMany = aList.size();

aList.set(index, “something else”);//replace index

String thing = aList.get(index);

boolean hasString = aList.contains(“something”);

aList.remove(index);

aList.clear();

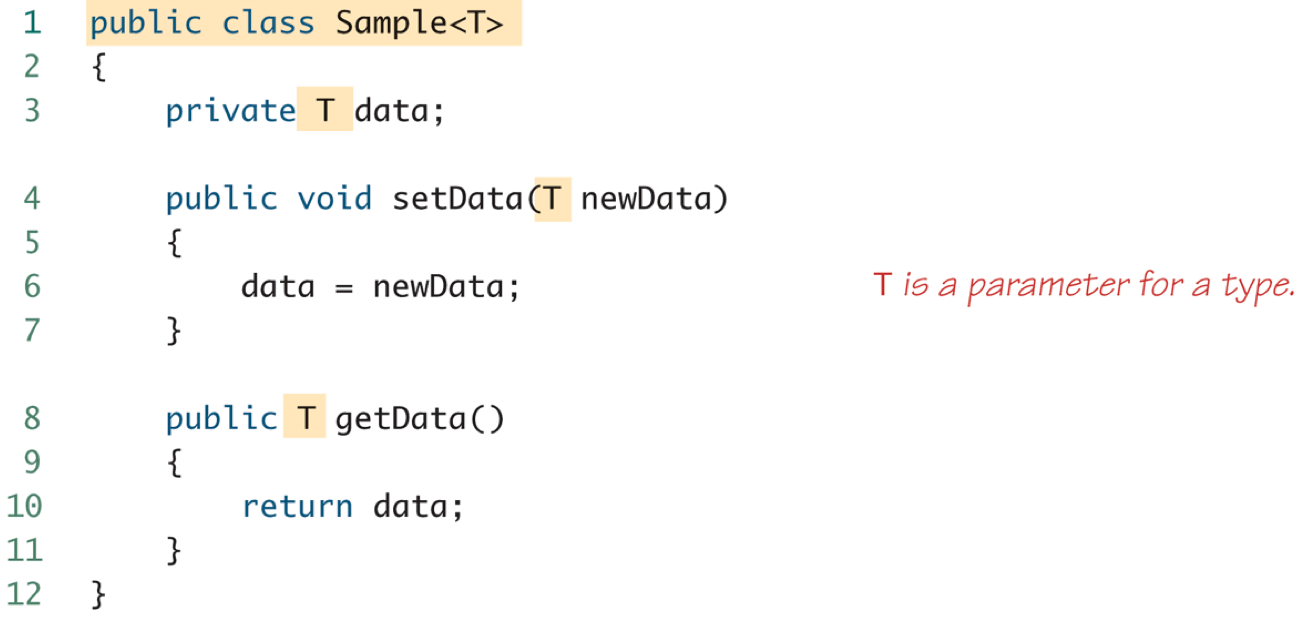
* 1. can be used with for-each

for (String entry: aList)

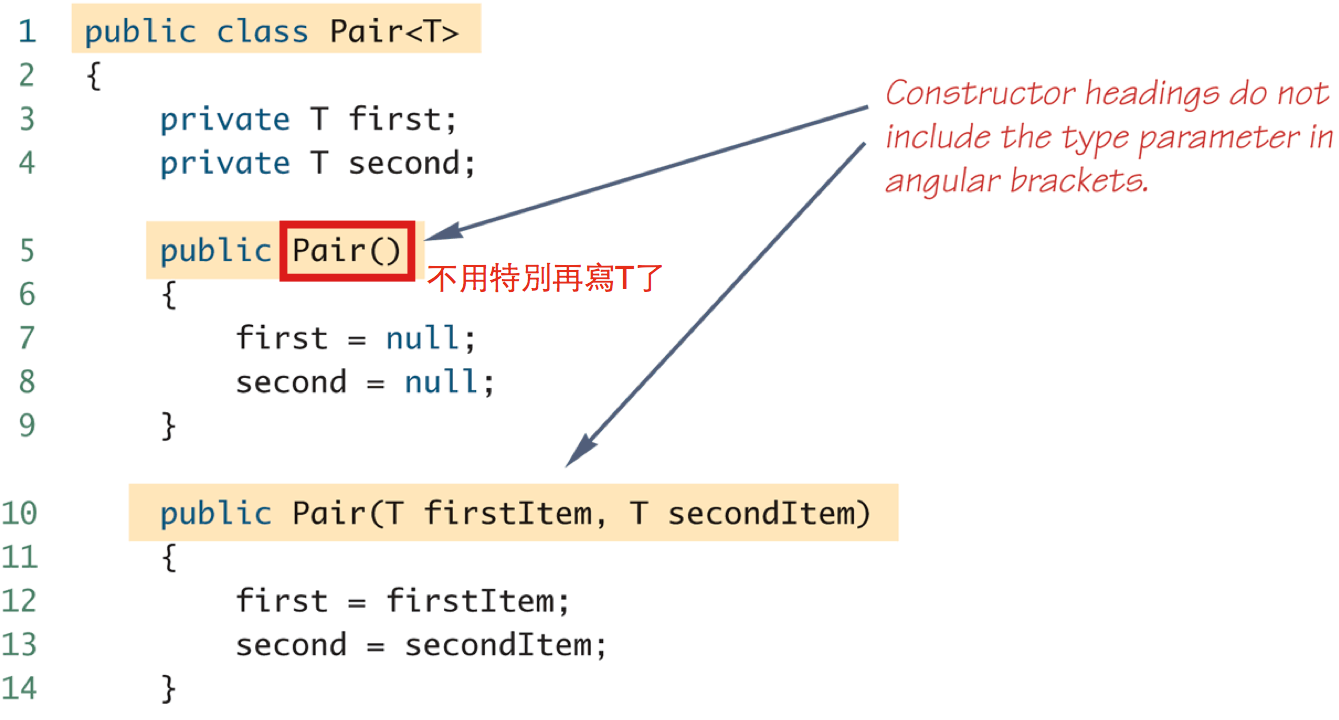
System.out.println(entry);

* 1. ArrayList grows automatically but does not shrink automatically, so use trimToSize to save memory
  2. clone method makes a shallow copy
  3. Vector class is almost same as ArrayList, but ArrayList is newer and prefered

1. generic class (or parameterized class)



* 1. definition: class definition with type parameter
  2. type parameter is included in angular brackets(<>) after the class name in the class definition heading
  3. type parameter no needed to be included in the constructor



* 1. type parameter cannot be used in expressions to create a new object當別人使用這個class的時候，必須要指定好型態！

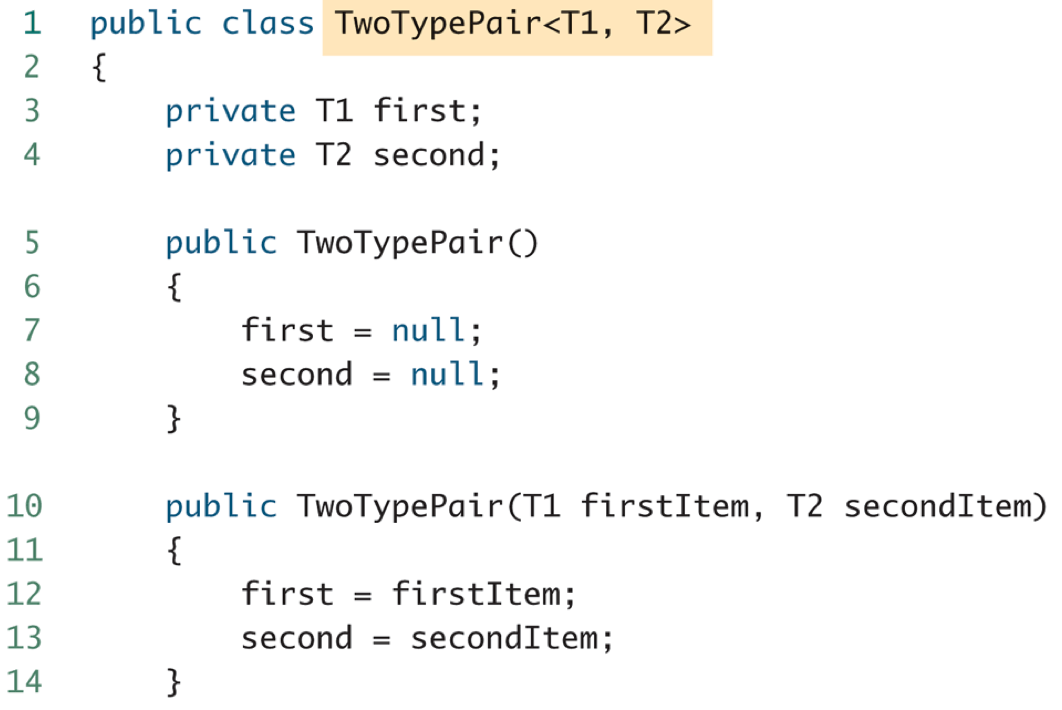
T object = new T(); //illegal

T[] a = new T[10]; //illegal

* 1. generic class cannot be an Array base type

Pair<String>[] a = new Pair<String>[10];

* 1. multiple type parameters



* 1. generic class cannot be an Exception class

public class Pair<T> extends Exception //illegal

* 1. bounds for type parameters
     1. restricting the possible types that can be plugged in for T

public class RClass<T extends Comparable>

public class ExClass<T extends Class1>

public class Two<T1 extends Class1, T2 extends Class2

& Comparable>

//不可以class & class，因為Java沒有多重繼承！

1. generic methods
   1. definition: type parameter used in the definitions of the methods for an ordinary class or a generic class
   2. type parameter of a generic method is local to that method, not to the class
   3. syntax

public static <T> T genMethod(T[] a) //<T>和T順序不可以錯！

String s = NonG.<String>genMethod(c);

1. inheritance with generic class

