* Generic class such as GenericStack and ArrayList used without a type parameter is called a raw type. Use of raw type is allowed for backward compatibility with the earlier versions of Java.
* Raw type is unsafe. For example, you might invoke the max method using

Max.max("Welcome", 23); // 23 is autoboxed into new Integer(23)

This would cause a runtime error, because you cannot compare a string with an integer object. The Java compiler displays a warning on line 3 when compiled with the option

–Xlint:unchecked, as shown in Figure 21.5.

* As another example, in the following code you may declare a raw type stack in line 1, assign new GenericStack<String> to it in line 2, and push a string and an integer object to the stack in lines 3 and 4.

1 GenericStack stack;

2 stack = new GenericStack<String>();

3 stack.push("Welcome to Java");

4 stack.push(new Integer(2));

Line 4 is unsafe because the stack is intended to store strings, but an Integer object is added into the stack. Line 3 should be OK, but the compiler will show warnings on both line 3 and line 4, because it cannot follow the semantic meaning of the program. All the compiler knows is that stack is a raw type, and performing certain operations is unsafe. Therefore, warnings are displayed to alert potential problems.