Chapter 21 Generics

1. (a) will compile fine, but (b) has a compilation error on Line 3, because dates is declared as a list of Date objects. You cannot assign a string to the list.

2. casting is needed in (a), but no casting is necessary in (b) with the generic type ArrayList<Date>.

3. One important benefit is improving reliability and robustness. Potential errors can be detected by the compiler.

4. No.

5. No.

6. Yes.

7. To declare a generic type for a class, place the generic type after the class name, such as GenericStack<E>. To declare a generic type for a method, place the generic type for the method return type, such as <E> void max(E o1, E o2).

8. Bounded generic type such as <E extends AClass> specifies that a generic type must be a subclass of AClass.

9. When you use generic type without specifying an actual parameter, it is called a raw type. GenericStack is roughly equivalent to GenericStack<Object>, but they are not the same. GenericStack<Object> is a generic instantiation, but GenericStack is a raw type.

10. ? is unbounded wildcard
    ? extends T is bounded wildcard
    ? super T is lower bounded wildcard

11. No. Only ArrayList is loaded.

12. Since all instances of a generic class have the same runtime class, the static variables and methods of a generic class is shared by all its instances. Therefore, it is illegal to refer a generic type parameter for a class in a static method or initializer.