OVERLOADED METHODS

In many cases a method with the same identifier has several declarations, each of which takes different arguments. When this happens, the method is said to be overloaded.

Example

The `print` method that is contained within the standard output object `System.out` has several overloaded versions. Here are the method specifications for several of them:

```java
void print( char c )
    // Prints the character in variable c.

void print( double d )
    // Prints the floating-point value in variable d.

void print( int i )
    // Prints the value of integer variable i.

void print( String s )
    // Prints the string s.
```

Because of these overloaded declarations, all of the calls to `print` below are valid.

```java
char symbol = '$';
double x = 3.29;
int m = 10;
System.out.print( symbol );
System.out.print( x );
System.out.print( m );
System.out.print( "x = " + x );
```