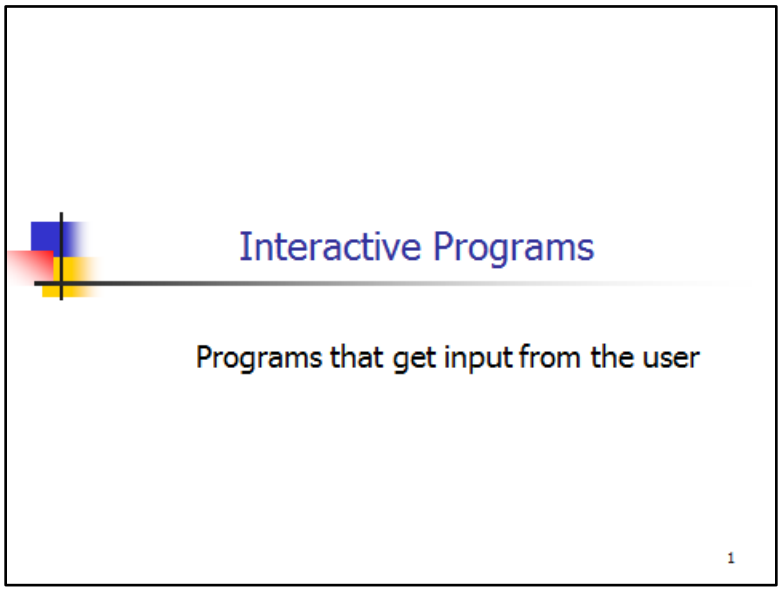
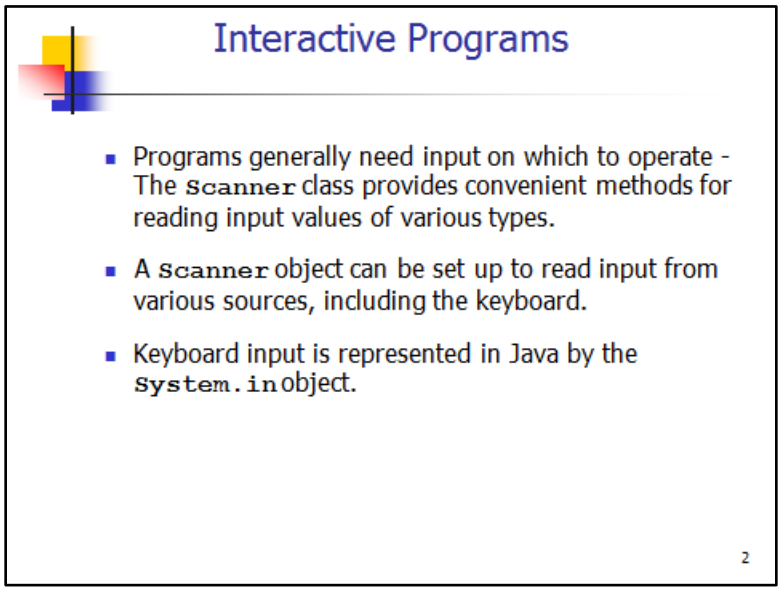


In this final section of today's class, we will look at how to get input from the user.

Assignments for this week are given at the end.

Getting input from a user in Java can be tricky and complicated. What we do today will just be a very brief introduction. You will learn more about getting input as the course progresses.



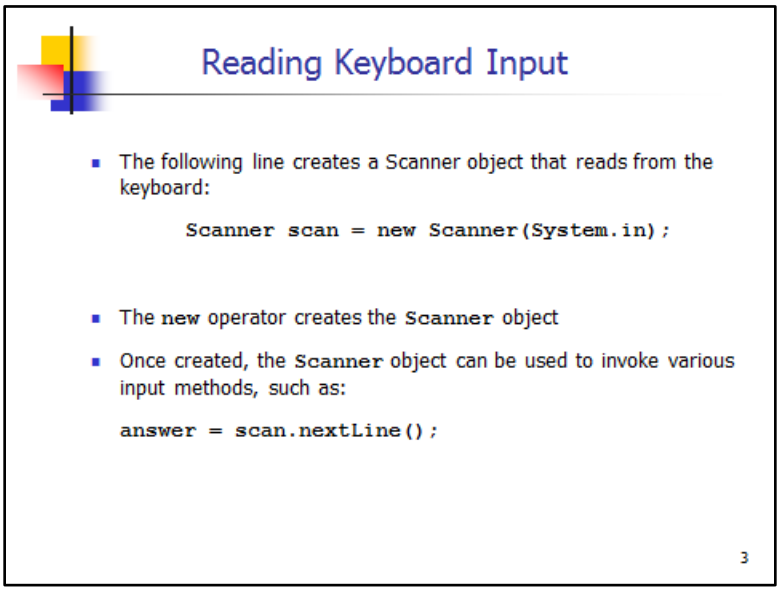
Most programs get some kind of input from the user. A program that always does the same thing and always produces the same output will not be very interesting.

A program that gets input from the user is called an *interactive* program, because it interacts with the user. The user input might be data on which the program performs a calculation. We will see an example of that shortly.

A program might be capable of performing several different function and let the user specify what it is to do.

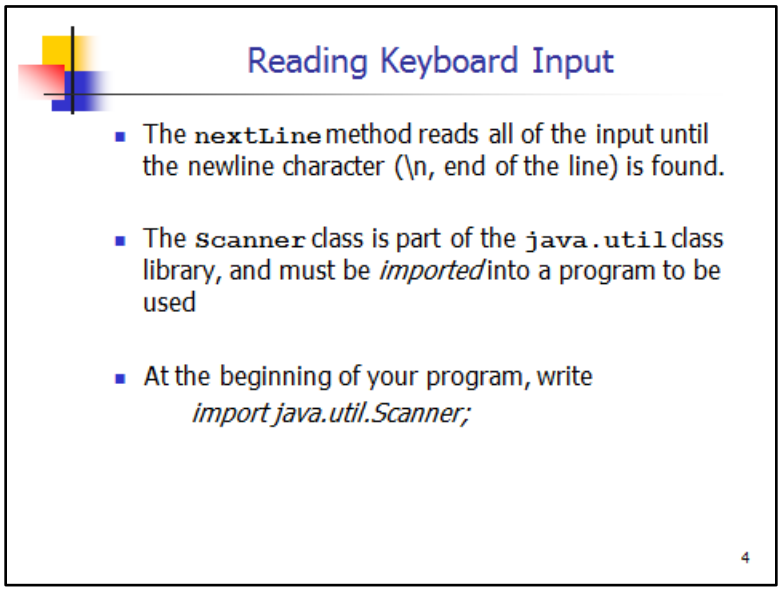
Java has library class called Scanner, which provides methods for getting various kinds of input. Reading a string, such as a person's name is not the same as reading a number. And reading an integer is not the same as reading a floating point number.

In order to get keyboard input we have to create a Scanner object. When we create the Scanner object, we pass it a built-in object called System.in. Compare this to the System.out object whose println method we use to output to the screen. Passing System.in to the Scanner object at the time we create it makes it a keyboard intput Scanner.

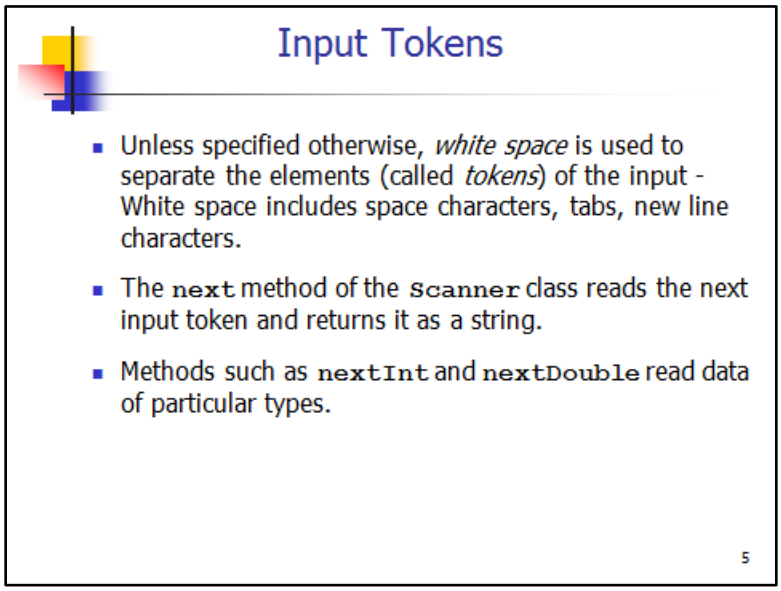


This slide shows how to create a keyboard input scanner object, using the Java operator *new*. Any program that gets input from the user will include a statement of this form.

You will see a lot more of the *new* operator as this course progresses.



The Scanner class is not built-in in the Java Compiler. It is included in a library file called java.util. In order to use the Scanner class, we have to *import* it from the java.util class library, using the statement shown on the last line of this slide. The import statement makes the imported class a part of the program as if we had written it. Every program that gets input from the user will have this line near its beginning.



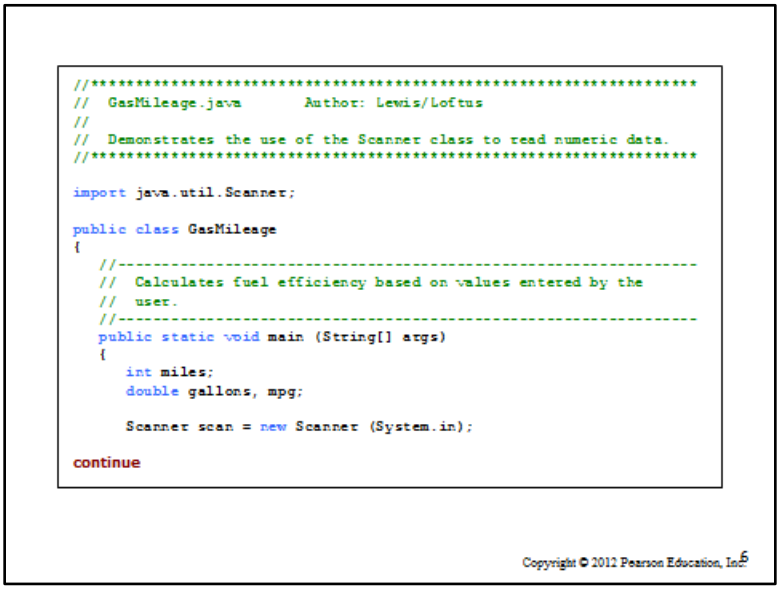
If we are reading multiple numbers from the user, in the form of keyboard input, we don't care about how the user separates the numbers. It is best not to constrain the user by requiring them to put each number on a separate line, or separate the numbers with a certain number of spaces or a tab.

The Scanner class can skip over white space and get then next string of non-whitespace characters typed by the user. We refer to these strings of non-whitespace characters as input *tokens*. The tokens are what we care about, not the white space.

The Scanner has a method called next that gets the next token from the keyboard input stream and returns it as a string.

If we want a particular kind of number, we can use the Scanner's nextInt or nextDouble methods.

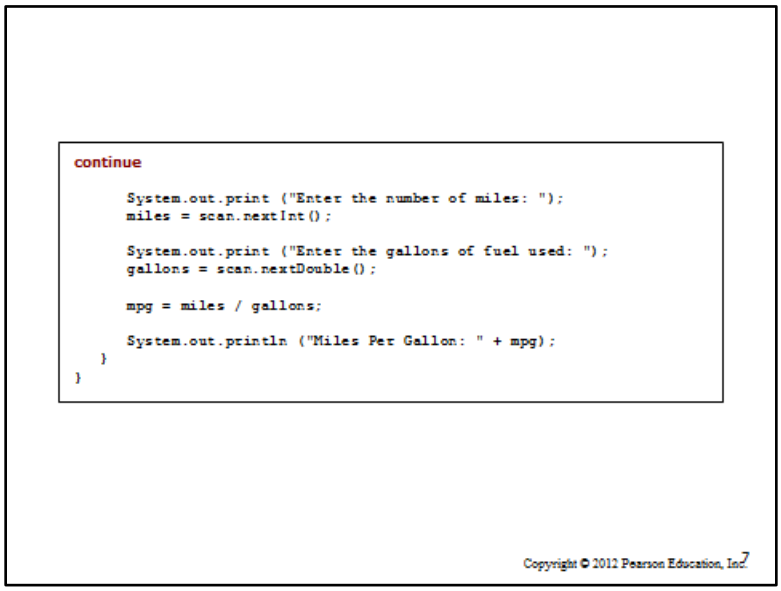
On the next slide is a program with examples of these methods. The program computes gasoline mileage. It promts the user to input a number of miles and the number of gallons of gasoline used. Then it computes and outputs the miles per gallon.



```
//********************************************************************
//  GasMileage.java       Author: Lewis/Loftus
//
//  Demonstrates the use of the Scanner class to read numeric data.
//********************************************************************

import java.util.Scanner;

public class GasMileage
{
   //-----------------------------------------------------------------
   //  Calculates fuel efficiency based on values entered by the
   //  user.
   //-----------------------------------------------------------------
   public static void main (String[] args)
   {
      int miles;
      double gallons, mpg;

      Scanner scan = new Scanner (System.in);

continue
```


This page has the declarations for variables that the program will use. And note the import statement near the top.

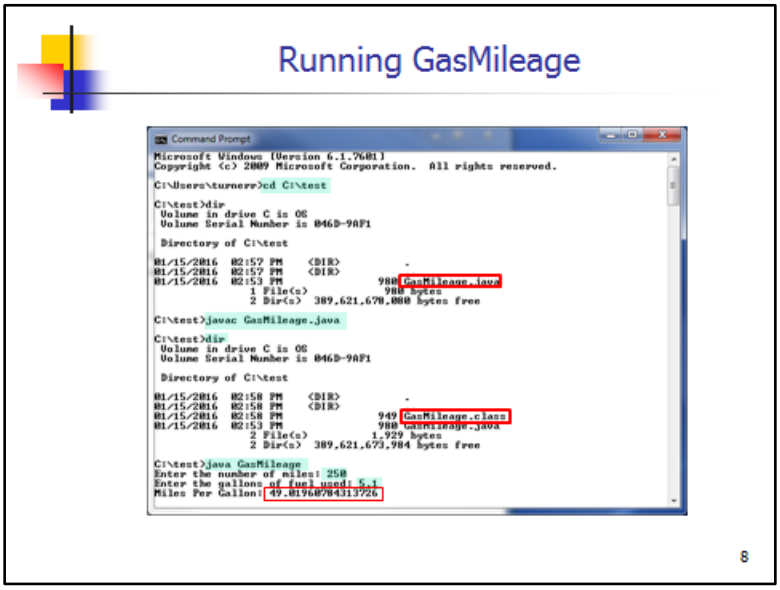
On this page, the program declares a variable, scan, to hold a Scanner object and then invokes the Java *new* operator to create the Scanner object, passing System.in as the argument.



On this page, the program prompts the user for number of miles and reads the input as an integer using the nextInt method of the scanner.

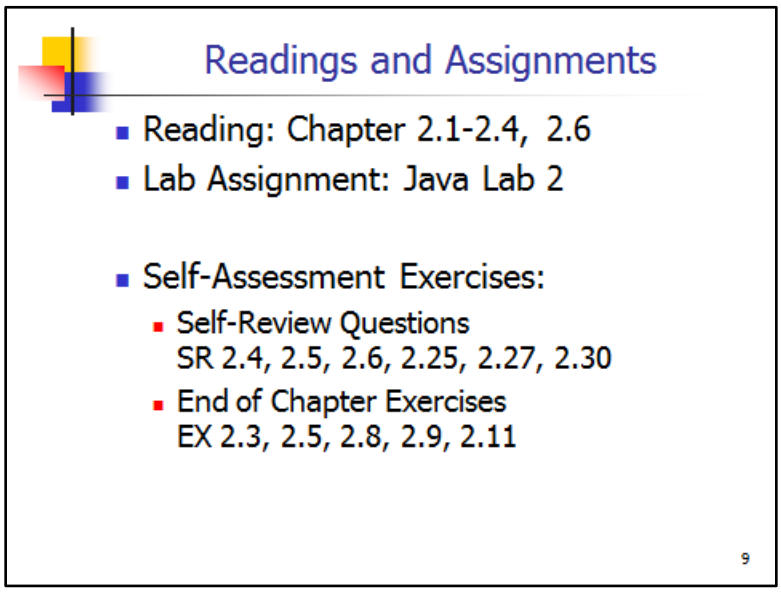
Then it prompts the user for number of gallons of fuel used and reads the input as a floating point number, using the nextDouble method.

It then computes miles per gallon and outputs the result using println.



This slide shows the GasMileage program being compiled and run. The source file is GasMileage.java. The Java compiler is run to compile GasMileage.java and write the bytecode file, GasMileage.class. Then the compiled program is run, using the java command.

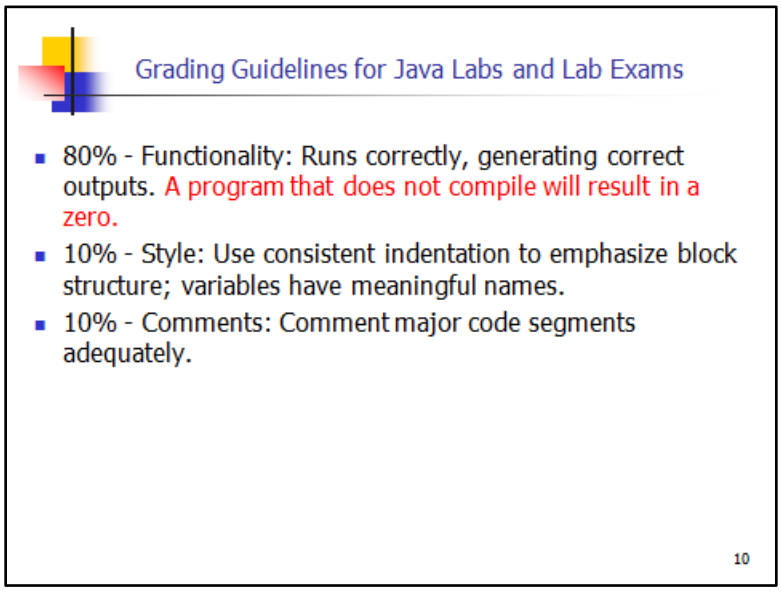
The final three lines show the program prompting for input, the user's input, and the program displaying its result.



Do the Reading and Self-Assessment Exercises on your own. If you have questions or if there is anything you don't understand, ask your TA in the lab session or ask your instructor in a help session.

See Canvas for the lab project. It is the only thing that will be graded and the only thing you should submit in Canvas.

I will send a message to the class when the assignment is available in Canvas.



(No narration)