For Friday

- Finish Becker, chapter 8, section 1

Program 5

- Any questions?

The Scanner Class

Interacting with the User

- Always ask for the data.
- User response should be on the same line as the prompt in almost all cases.
- Ask for only one item at a time.
- Be clear, courteous, and concise.

System.in

Practice Problem

- Write Java code to ask for a student's exam score out of 100. Your program is then to match the exam score to a letter grade and print the grade to the screen. The letter grade is to be calculated as follows:

90 and above	A
80-89	B
70-79	C
60-69	D
below 60	F

Practice

- Design and write a program that will prompt for, receive, and total a collection of payroll amounts entered by the user until a sentinel amount of -99 is entered. After the sentinel has been entered, display the total payroll amount on the screen.

Problem 3

- Write a program that sums a sequence of integers. Assume that the first integer read specifies the number of values remaining to be entered. Your program should read only one value at a time. A typical input sequence might be

  5 100 200 300 400 500

Problem 4

- Write a program that finds the smallest of several integers. Assume that input will end when a sentinel value of –999 is read. Do not count –999 as one of the integers to consider.

Interfaces

- Used to specify a particular set of methods that a class needs to implement.

TollBooth Problem

- Write a class to manage tolls.

Weight	Toll
1-5000	$0.35
5001-25000	$0.50
>25000	$1.50

Beginning the Class

```
public class TollBooth extends
{
  public TollBooth(                          ) …
  public       arrival(                      ) …
  public       departure(                    ) …
  public       collectCoin(                  ) …
  public       getAmountOwed(                ) …
  public       okToLiftGate(                 ) …
  public       getTotalCollected(            ) …
  public       getTotalVehicles(             ) …
}
```

- Add return types and parameters to the method names given in the problem statement. Turn each method into a “stub” for testing by adding just enough of the body so that it will compile.

ITollBooth and GUI

- Write a **method** to compute the sum of all integers between first and second (including first and second), where first and second are integers and first <= second. The method should return the sum. You may not change the value of either first or second.

- Write a method to find the smaller of two integers. The method will accept two integers and return the smaller of the two. If they are the same, then the method returns either one of them.

- A company gives bonuses based on production as follows:
  - 1000 units or fewer, the bonus is $25
  - 1001 to 3000 units, the bonus is $50
  - 3001 to 6000 units, the bonus is $100
  - 6001 units and up, the bonus is $200
- Write a method that accepts the number of units produced and determines the bonus for the employee. Return the bonus.

- Write a method to determine the purchaser's discount based on a code.
  - If the code is 7, the discount is 10%.
  - If the code is 3, the discount is 15%.
  - If the code is 12, the discount is 4%.
  - If the code is 1, there is no discount.
  - If the code is 8, the discount is 30%.
- The method should return the discount. Use a switch statement.