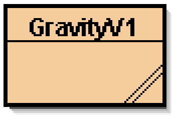
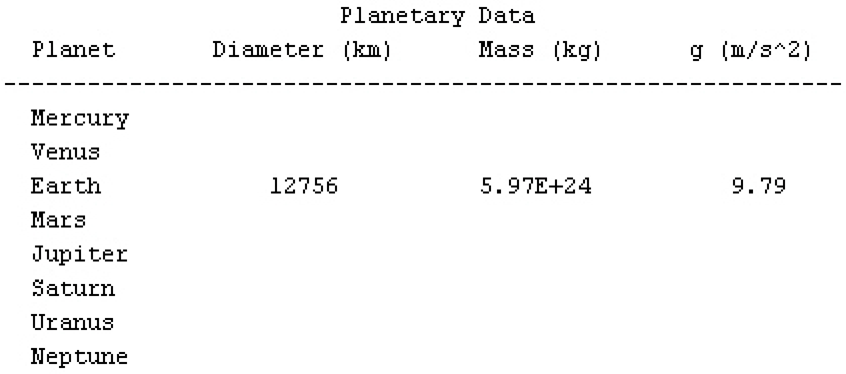
7.03 Assignment Instructions

Instructions: Write a program to determine the surface gravity (g) on each planet in our solar system. (The results of this assignment will be used in later lessons, but will not be submitted for a grade at this time.)



1. Create a class called GravityV1 in the 7.03 Declaring New Methods project folder.
2. Read the 7.03 Virtual Lecture Notes (Part 2) to learn how to calculate each planet's surface gravity.
3. Record the mass and diameter of each planet in the table at the end of the Assignment Instructions.
4. Assign planet names, mass, and diameter to separate arrays.
5. Calculate the surface gravity for each planet. Depending on the accuracy of your information source, calculated values should be reasonably accurate but may vary somewhat.
6. Display the information in a neatly formatted table. (See expected output.) Use a Formatting Grid to save time developing the layout for your output.
7. Write only the surface gravity data to a text file. When you inspect the file with Notepad, there should be nine numbers, one on each row. (You will need this file in the future.)

Expected Output: When your program runs correctly, the output should resemble the following screen shot. Only data for Earth are shown here to help you with formatting, but each column of your output will be filled in.



```
                          Planetary Data
   Planet       Diameter (km)      Mass (kg)      g (m/s^2)
-------------------------------------------------------------
   Mercury
   Venus
   Earth            12756          5.97E+24          9.79
   Mars
   Jupiter
   Saturn
   Uranus
   Neptune
```

* The carat symbol (**^**) is used in the heading of the last column to indicate that the unit for seconds (s) is squared In some languages, arithmetic expressions can be raised to a power with the carat symbol, but in Java you must use the **`Math.pow()`** method.. It "works" in this case because **^** is part of a **String** literal, not an arithmetic expression.

** The data in column three are printed in scientific notation (e.g. 3.30E+23). The **`printf()`** method has additional type converters that can be used in the format specifier. The letter "E" (upper- or lowercase) is the conversion specifier for scientific notation.

Assessment: You will continue working on this assignment in Lesson 07.04, so it will not be graded at this time; however, the following rubric indicates the important features of the program.

Grading Rubric	**Pts**
Comments include name, date, and purpose of program.	1
Data assigned to each array with single line.	1
Separate method to calculate surface gravity of the planets.	3
Separate method to print results to the screen.	3
Separate method to write the results to a text file.	3
Arithmetic statements written correctly.	2
Output formatted to display with **`printf()`**	2
Results written to text file as specified.	2
No compiler or runtime errors.	1
Output is correct and user-friendly.	1
Thoughtful PMR included.	1

Submission: Do not submit GravityV1.java at this time next lesson.

Data: In order to complete this assignment, fill-in the missing data for diameter and mass of each planet. If you locate information for the surface gravity of each planet, record it in the last column so you can compare the accuracy of your program's results.

Planetary Data

Planet	**Diameter (km)**	**Mass (kg)**	**g (m/sec^2)**
Mercury			
Venus			
Earth			
Mars			
Jupiter			
Saturn			
Uranus			
Neptune			