

VITAL INFORMATION

Subject(s):	Careers, Computer Fundamentals 1-2
Topic or Unit of Study:	Software Development
Grade/Level:	9-12
Objective:	At the conclusion of this lesson students will be able to: <ol style="list-style-type: none">1. Declare a String variable.2. Read user input from a Console application in Visual Basic.3. Explain what a While loop does and how it can be exited.4. Predict output of Val() for nearly every input.

Summary: Students create another Console Application, this time to investigate the Val() function which gave them a difficult time in the My Calculator activity. It provides them an opportunity to use a String variable, see another loop in action, and to learn about the many strange behaviors of Val().

IMPLEMENTATION

Learning Context: The Val() function appeared first in the My Calculator activity. As part of that activity, students were required to document irregularities in the calculations. Most were the result of strange and largely undocumented behaviors of Val(). Students received feedback on their specific findings, but were never informed of answers outside of their own, mostly because several students have yet to hand in the work. The end of the quarter is approaching and it's now or never. I think that students will be impressed with their collective findings and recognize much of their experience in the numbers.

Procedure:

1. The activity is written up in detail on a web page which is printed and attached to this lesson plan. Although students can read the instructions on their computers, they probably have questions and comments that interest the entire group. We therefore generally conduct such activities at the front of the room near the screen where everyone can see and hear. Such is the plan for this activity.
2. Work with the students through the background information, instructions, and their questions, and then set them to work creating their ValTest programs.
3. When students have completed their work, they should have collected a long list of input and output pairs which can be used as

evidence for learning and be assessed as described below.

Differentiated Instruction:

There is little differentiation in this assignment. Students may work at their own pace as soon as group discussion is complete, but the program involves only copying and pasting code, so everyone's programs and tables should be exactly the same.

Sample Student Products:

The complete source code is included on the web page. It can be run in order to produce the answer key.

Collaboration:

Students will work individually.

Time Allotment:

1 class period. 55 Min. per class.

Author's Comments & Reflections:

Reflections will follow in a diary entry.

MATERIALS AND RESOURCES

Instructional Materials:

The web page for this activity is attached.

Attachments

- | |
|--------------------|
| 1. Val Test |
|--------------------|

Resources:

- Technology resources:
Internet Explorer, Visual Basic

STANDARDS & ASSESSMENT

Standards:

 **AZ- Career and Technical Education Programs**

- **Level** : Career Preparation (Grades 10 - 12)
- **Program** : Information Technology CIP No. 15.1200
- **Option** : Software Development - Option C
- **Competency** : 27.C DEMONSTRATE PROGRAM ANALYSIS AND DESIGN
 - **Indicator** : 27.4c Determine input and output
 - **Indicator** : 27.7c Develop a testing plan
- **Competency** : 28.C USE SOFTWARE TO CREATE PROGRAMS
 - **Indicator** : 28.1c Enter and modify code using a program editor
 - **Indicator** : 28.2c Compile and execute programs
- **Competency** : 29.C TEST AND DEBUG TO VERIFY PROGRAM OPERATION
 - **Indicator** : 29.1c Test individual program modules
 - **Indicator** : 29.2c Identify boundary cases and generate appropriate test data
- **Competency** : 31.C EMPLOY MODULARITY IN WRITING PROGRAMS
 - **Indicator** : 31.1c Call standard library functions
 - **Indicator** : 31.2c Utilize parameters to pass data into program modules
- **Competency** : 33.C UTILIZE REPETITION STRUCTURES
 - **Indicator** : 33.1c Identify various types of repetition structures
- **Competency** : 36.C IDENTIFY WAYS TO INPUT AND OUTPUT INFORMATION
 - **Indicator** : 36.1c Provide user with means to input data on a console and/or GUI
 - **Indicator** : 36.2c Use input/output statements in a program

■ **Indicator** : 36.3c Assign input to variables

Assessment/Rubrics: The output of this activity is a table of input and output values for the Val() function. There is no simple way that the table could be produced without a functioning program, so the table is evidence of a complete program. Check student results against each other or have students compare their answers in small groups.