VITAL INFORMATION

Subject(s): Careers, Computer Fundamentals 1-2

Topic or Unit of Study: Integrated Unit

Grade/Level: 9-12

Objective: At the conclusion of this lesson students will be able to:

1. Add a PictureBox to another control (Form or TabPage).

2. Name the PictureBox and set simple properties such as SizeMode.

3. Configure an Image or write code to set the ImageLocation location at runtime.

Summary: Students add and configure one or more PictureBoxes for use in their Computer Integration project. The PictureBoxes are most likely unaffected by user input, but rather interaction with other controls is translated into instructions to change the Picturebox ImageLocation property. The locations should demonstrate or document student findings about computer integration. In adding and editing the controls, students learn how they work.

IMPLEMENTATION

Learning Context: Students have completed code for the Schedule tab of their application and are now adding the controls which will allow them to present their Computer Integration findings. Depending on their plans, they may add controls in any order. The PictureBox is likely controlled by a LinkLabel or ComboBox, which should be programmed first.

Procedure: 1. The activity is written up in fairly fine detail on a web page which is printed and attached to this lesson plan. It specifies where to find the PictureBox, how to name it, and how to arrange for it to show a picture. It does not review the big picture, so the stage should be reset using some of the learning context from the introduction to the integrated unit.

2. Ensure that students can find the web page, ask for questions, and have students start.

3. Since this is a long, sequential unit, more attention needs to be paid to holding it all together and keeping it synchronized. Visit students early and often to ensure that they are keeping up. It may be a good idea to have students demonstrate their progress beginning five

minutes before the bell rings.

Differentiated Instruction: There is little differentiation in instruction, but an expectation that the products are differentiated because of differing courses, schedules, and research findings.

Sample Student Products: The attached printout of a web page shows an example of how the program might look with one of these controls.

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 activity page from the class web site is printed and attached. Students will edit their programs both in Design View and Code View.

Attachments

1. **PictureBox**

Resources:

- Technology resources:
  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 :** *3.0 DEVELOP APPROPRIATE WORK HABITS FOR SUCCESSFUL EMPLOYMENT IN INFORMATION TECHNOLOGY
        - **Indicator :** 3.3 Complete tasks accurately
        - **Indicator :** 3.4 Complete tasks with minimal supervision
      - **Competency :** 27.C DEMONSTRATE PROGRAM ANALYSIS AND DESIGN
        - **Indicator :** 27.6c Use stepwise refinement to improve design
      - **Competency :** 28.C USE SOFTWARE TO CREATE PROGRAMS
        - **Indicator :** 28.1c Enter and modify code using a program editor
      - **Competency :** 29.C TEST AND DEBUG TO VERIFY PROGRAM OPERATION
        - **Indicator :** 29.1c Test individual program modules
      - **Competency :** 28.C USE SOFTWARE TO CREATE PROGRAMS
        - **Indicator :** 28.2c Compile and execute programs
        - **Indicator :** 28.5c Use recognized conventions for naming identifiers and formatting code
      - **Competency :** 34.C USE SIMPLE DATA TYPES AND STRINGS
        - **Indicator :** 34.4c Write assignment statements for initializing and modifying variables
      - **Competency :** 36.C IDENTIFY WAYS TO INPUT AND OUTPUT INFORMATION
        - **Indicator :** 36.5c Use graphics to create images at specified locations

- **Competency :** 38.C EMPLOY OBJECT-ORIENTED PROGRAMMING TECHNIQUES
  - **Indicator :** 38.5c Change the state of an object by invoking a modifier method

Assessment/Rubrics: By the end of the class period, students should have added, named, texted, and programmed one or more LinkLabels. Each activity contributes a few criteria to the larger rubric. This activity adds points for the name, text, and click handler as described by the LinkLabel rubric.

Rubrics

1. **PictureBox**