

Programming Logic - Beginning
152-101

Money Changer Program Evaluation Form

Name _____

Score _____ / 20

Update Value _____

[Make all corrections and resubmit to earn update points](#)

☒ Update Recommended

Money Changer Project

- ☐ (½ point) Appropriate project icon (can be same as form icon)
- ☐ (½ point) Form file names match form object names (frmXxxx).

Welcome Screen

- ☐ (½ point) Form centered on screen
- ☐ (½ point) Form title is appropriate
- ☐ (½ point) Form includes application name
- ☐ (½ point) Form appearance is professional
- ☐ (½ point) Timer included that launches application after 5 seconds
- ☐ (½ point) Button included to start application immediately
- ☐ (½ point) Button included to exit application immediately (Cancel button)
- ☐ (½ point) Method documentation (grammatically correct) included for all methods

Form Layout

- ☐ (½ point) Appropriate form icon
- ☐ (½ point) Form centered on screen
- ☐ (½ point) Form title is appropriate
- ☐ (½ point) Textbox used to allow user to enter total change amount due
- ☐ (½ point) Four coins needed labels included
- ☐ (½ point) Form includes four pieces of clipart for quarter, dime, nickel, penny
- ☐ (½ point) All form objects appropriately labeled
- ☐ (½ point) Form objects have professional layout
- ☐ (½ point) Tab order is correct

- ☐ (½ point) All form objects referenced in code are appropriately named
- ☐ (½ point) Button access keys are appropriate
- ☐ (½ point) Calculate button is form Accept button
- ☐ (½ point) Clear button is form Cancel button

Calculate Click Event

- ☐ (½ point) Variable created for input
- ☐ (½ point) Change due is a decimal
- ☐ (½ point) Variables created for all outputs
- ☐ (½ point) All outputs are integers
- ☐ (½ point) Variables have appropriate names (camelBack), types and descriptions (comments)
- ☐ (½ point) Input value properly converted to variable
- ☐ (½ point) Input value unformatted
- ☐ (½ point) Try-Catch used to capture total change needed entry error
- ☐ (½ point) Error message appropriate
- ☐ (½ point) Change needed converted from dollars and cents to cents
- ☐ (½ point) Number of each type of coin calculated using integer division
- ☐ (½ point) Remaining amounts after coin removed calculated using modulus
- ☐ (½ point) Changed needed entered by user formatted Currency
- ☐ (½ point) Focus set to Clear button after all processing complete

Clear Click Event

- ☐ (½ point) Input textbox cleared
- ☐ (½ point) All output labels cleared
- ☐ (½ point) Focus set to input field after all processing complete

Exit Click Event

- ☐ (½ point) Program exits properly

Form Load Event

- ☐ (½ point) Defines form Welcome variable pointer
- ☐ (½ point) Displays form as Dialog and saves result
- ☐ (½ point) If user selects exit button, application immediately ends

Indentation

- ☐ (½ point) Source code indented properly

Challenge Points

- ☐ (+1 point) If all requirements met (limited to +1 for all unit 4 programs)
 - ☐ (½ point) Version number displayed on the form
 - ☐ (½ point) Version number displayed in an appropriate location
 - ☐ (½ point) Version number defined using a constant
 - ☐ (½ point) Version constant added to form
- ☐ (+1 point) If all requirements met (limited to +1 for all unit 4 programs)
 - ☐ (½ point) Exit button anchored to bottom right
 - ☐ (½ point) Calc and Clear buttons anchored to bottom left
 - ☐ (½ point) Version number (if provided) anchored appropriately

Form Level Documentation

- ☐ (1/2 point) Methods alphabetized
- ☐ (1/2 point) Spacing used to make code readable
- ☐ (1/2 point) Appropriate in-line comments included

- ☐ (1/2 point) Each method includes a purpose description.
- ☐ (1/2 point) Purpose descriptions use complete sentences, grammatically correct, without spelling errors.
- ☐ (1/2 point) Purpose descriptions are accurate

Program Documentation

- ☐ (1/2 point) Documentation module created
- ☐ (1/2 point) Documentation appearance is professional
- ☐ (1/2 point) Program name included
- ☐ (1/2 point) Student's name included
- ☐ (1/2 point) Program date included
- ☐ (1/2 point) Program's purpose uses complete sentences, is grammatically correct, without spelling errors.
- ☐ (1/2 point) Program's purpose is accurate and complete

- ☐ (1/2 point) Forms listed alphabetically
- ☐ (1/2 point) Form purpose descriptions use complete sentences, grammatically correct, without spelling errors.
- ☐ (1/2 point) Form purpose descriptions are accurate and complete

- ☐ (1/2 point) Change log included
- ☐ (1/2 point) (Updates only) Updates documented in change log

Program Design

- ☐ (1/2 point) btnCalc sequence of commands same in design and code
- ☐ (1/2 point) btnClear sequence of commands same in design and code
- ☐ (1/2 point) Variable and object names in code same as in design
- ☐ (1/2 point)