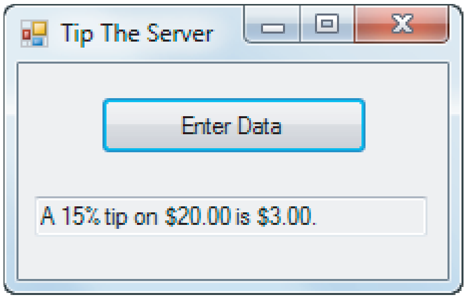
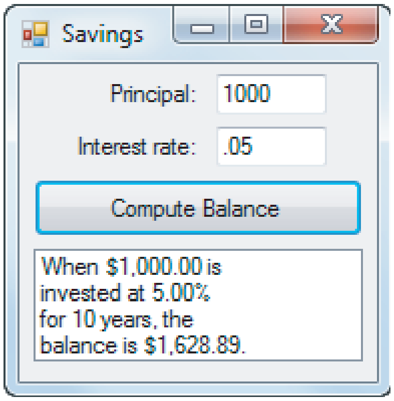


A 15% tip on $20.00 is $3.00.

FIGURE 3.14 Sample output of Exercise 74.



Principal: 1000

Interest rate: .05

When $1,000.00 is invested at 5.00% for 10 years, the balance is $1,628.89.

FIGURE 3.15 Sample output of Exercise 75.

**76.** Write a program to print the list of Internet lingo in Fig. 3.16.

```
PLS         Please
TAFN        That's all for now
HHOK        Ha, ha - only kidding
FWIW        For what its worth
IMO         In my opinion
```

FIGURE 3.16 Output of Exercise 76.

```
                      % of
Rank     Country      WW Users
1        USA          16.0%
2        China        11.9%
3        Japan        6.5%
```

FIGURE 3.17 Output of Exercise 77.

**77.** Write a program to print the top three ranking counties by the percentage of worldwide Internet users they contain as shown in Fig. 3.17.

### Solutions to Practice Problems 3.3

1. The first statement is correct, since FormatNumber evaluates to a string. Although the second statement is not incorrect, the use of CStr is redundant.
2. The outcomes are identical. In this text, we primarily use the second style.

## CHAPTER 3 SUMMARY

1. Three types of *literals* that can be stored and processed by Visual Basic are numbers, strings, and dates.
2. Many Visual Basic tasks are carried out by methods such as Clear (erases the contents of a text box or list box), Add (places an item into a list box), ToUpper (converts a string to uppercase), ToLower (converts a string to lowercase), Trim (removes leading and trailing spaces from a string), IndexOf (searches for a specified substring in a string and gives its position if found), and Substring (produces a sequence of consecutive characters from a string).
3. The *arithmetic operations* are +, −, *, /, ^, \, and Mod. The only string operation is &, concatenation. An *expression* is a combination of literals, variables, functions, and operations that can be evaluated.
4. A *variable* is a name used to refer to data. Variable names must begin with a letter or an underscore and may contain letters, digits, and underscores. Dim statements declare variables,

specify the data types of the variables, and assign initial values to the variables. In this book, most variables have data types Double, Integer, String, or Date.

5. Values are assigned to variables by *assignment statements.* The values appearing in assignment statements can be literals, variables, or expressions. String literals used in assignment statements must be surrounded by quotation marks. Date literals used in assignment statements must be surrounded by number signs.

6. *Comment statements* are used to explain formulas, state the purposes of variables, and articulate the purposes of various parts of a program.

7. *Option Explicit* requires that all variables be declared with Dim statements. *Option Strict* requires the use of conversion functions in certain situations.

8. The *Error List window* displays, and helps you find, errors in the code. The *Auto Correction* feature of IntelliSense suggests corrections when errors occur.

9. Line continuation is used to extend a Visual Basic statement over two or more lines.

10. The *scope* of a variable is the portion of the program in which the variable is visible and can be used. A variable declared inside an event procedure is said to have *local* scope and is visible only inside the procedure. A variable declared in the Declarations section of a program is said to have *class-level* scope and is visible throughout the entire program.

11. *Masked text boxes* help obtain correct input with a Mask property that specifies the kind of data that can be typed into the text box.

12. The *Date data type* facilitates computations involving dates.

13. An *input dialog box* is a window that pops up and displays a message for the user to respond to in a text box. The response is assigned to a variable.

14. A *message dialog box* is a window that pops up to display a message to the user.

15. *Named constants* store values that cannot change during the execution of a program. They are declared with Const statements.

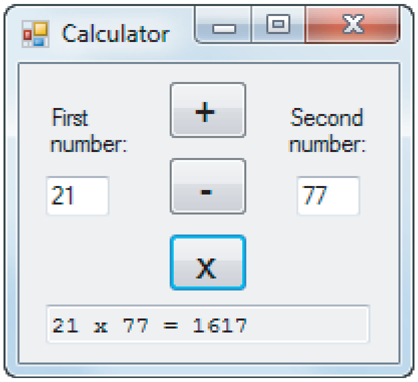
16. The *PrintDocument control* is used to send output to the printer, and the *PrintPreviewDialog control* is used to preview the output.

17. The following *functions* accept numbers, strings, or dates as input and return numbers or strings as output.

| FUNCTION | INPUT | OUTPUT |
|---|---|---|
| CDbl | string or number | number |
| CInt | string or number | number |
| CStr | string or number | string |
| FormatCurrency | number | string |
| FormatNumber | number | string |
| FormatPercent | number | string |
| FormatDateTime | date | string |
| DateDiff | date, date | number |
| InputBox | string, string | string |
| Int | number | number |
| Math.Round | number, number | number |
| Math.Sqrt | number | number |

## CHAPTER 3 PROGRAMMING PROJECTS

1. Write a program that allows the user to specify two numbers and then adds, subtracts, or multiplies them when the user clicks on the appropriate button. The output should give the type of arithmetic performed and the result. See Fig. 3.18. ***Note:*** If one of the numbers in an input text box is changed, the output text box should be cleared.

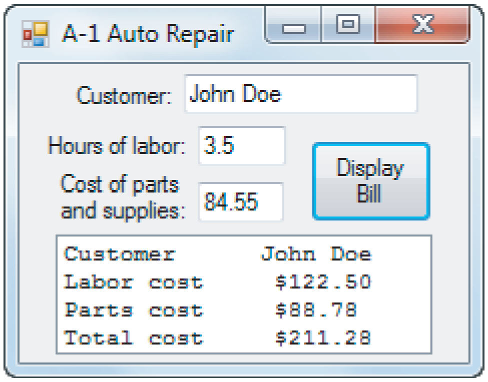


First number: 21

Second number: 77

```
21 x 77 = 1617
```

FIGURE 3.18 Possible outcome of Programming Project 1.



Customer: John Doe

Hours of labor: 3.5

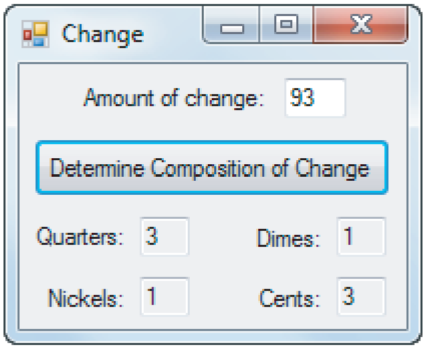
Cost of parts and supplies: 84.55

```
Customer        John Doe
Labor cost       $122.50
Parts cost       $88.78
Total cost       $211.28
```

FIGURE 3.19 Possible outcome of Programming Project 2.

2. Suppose automobile repair customers are billed at the rate of $35 per hour for labor. Also, suppose costs for parts and supplies are subject to a 5% sales tax. Write a program to display a simplified bill. The customer's name, the number of hours of labor, and the cost of parts and supplies should be entered into the program via text boxes. When a button is clicked, the customer's name and the three costs should be displayed in a list box, as shown in Fig. 3.19.

3. Write a program to make change for an amount of money from 0 through 99 cents input by the user. The output of the program should show the number of coins from each denomination used to make change. See Fig. 3.20.

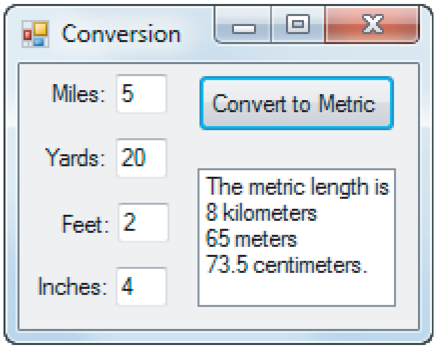


Amount of change: 93

Quarters: 3 Dimes: 1

Nickels: 1 Cents: 3

FIGURE 3.20 Possible outcome of Programming Project 3.



Miles: 5

Yards: 20

Feet: 2

Inches: 4

The metric length is
8 kilometers
65 meters
73.5 centimeters.

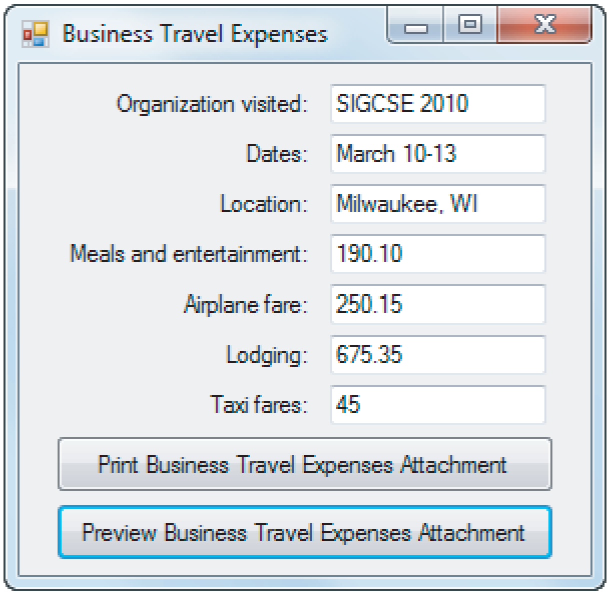
FIGURE 3.21 Possible outcome of Programming Project 4.

4. Write a program to convert a U.S. Customary System length in miles, yards, feet, and inches to a Metric System length in kilometers, meters, and centimeters. A sample run is shown in Fig. 3.21. After the numbers of miles, yards, feet, and inches are read from the text boxes, the length should be converted entirely to inches and then divided by 39.37 to obtain the value in meters. The Int function should be used to break the total number of

meters into a whole number of kilometers and meters. The number of centimeters should be displayed to one decimal place. The needed formulas are as follows:

$$\text{total inches} = 63360 * \text{miles} + 36 * \text{yards} + 12 * \text{feet} + \text{inches}$$
$$\text{total meters} = \text{total inches}/39.37$$
$$\text{kilometers} = \text{Int}(\text{meters}/1000)$$

5. Write a program to print a business travel expenses attachment for an income tax return. The program should request as input the name of the organization visited, the dates and location of the visit, and the expenses for meals and entertainment, airplane fare, lodging, and taxi fares. (Only 50% of the expenses for meals and entertainment are deductible.) A possible form layout and run are shown in Figs. 3.22 and 3.23, respectively.



Organization visited: SIGCSE 2010
Dates: March 10-13
Location: Milwaukee, WI
Meals and entertainment: 190.10
Airplane fare: 250.15
Lodging: 675.35
Taxi fares: 45

Print Business Travel Expenses Attachment
Preview Business Travel Expenses Attachment

FIGURE 3.22 Form with sample data for Programming Project 5.

```
Business Travel Expenses

Trip to attend meeting of
SIGCSE 2010
March 10–13 in Milwaukee, WI

Meals and entertainment:      $190.10
Airplane fare:                $250.15
Lodging:                      $675.35
Taxi fares:                   $45.00

Total other than meals and entertainment: $970.50
50% of meals and entertainment: $95.05

TOTAL DEDUCTIBLE EXPENSES: $1,065.55
```

FIGURE 3.23 Output for sample run of Programming Project 5.