Programming and Logic Design

- LESSON 6

- FUNCTIONS

Learning Objectives

6.1 Introduction to Functions: Generating Random Numbers
6.2 Writing Your Own Functions
6.3 More Library Functions

6.1 Introduction to Functions (1 of 2)

A **function** is a module that returns a value back to the part of the program that called it

- Many languages provide libraries of functions that you can use, such as Random Number Generator
- A function is like a module, but it returns a value that can be used in your program

Library functions

- Written functions that come with most languages
- Usually common tasks and save time for the programmer because it allows for code reuse

6.1 Introduction to Functions (2 of 2)

The Random Number Generator function is useful in:

- Game programs
- Simulation programs
- Statistical programs
- Computer security such as encryption

How random function works

- **Set number = random(1, 100)**
- 1 and 100 define the range of the number that can be returned, and are called arguments
- The function is called and a random number is returned and assigned to the variable **number**

6.2 Writing Your Own Functions (1 of 3)

Most languages allow coders to write functions

- The **function header** specifies the data type of the value that is returned, the name of the function, and any parameter variables
- The **function body** are the statements that execute when the function calls
- The **return statement** specifies the value that is returned when the function ends

6.2 Writing Your Own Functions (2 of 3)

Program 6-6

```
 1 Module main()
 2    // Local variables
 3    Declare Integer firstAge, secondAge, total
 4
 5    // Get the user's age and the user's
 6    // best friend's age.
 7    Display "Enter your age."
 8    Input firstAge
 9    Display "Enter your best friend's age."
10    Input secondAge
11
12    // Get the sum of both ages.
13    Set total = sum(firstAge, secondAge)
14
15    // Display the sum.
16    Display "Together you are ", total, " years old."
17 End Module
18
19 // The sum function accepts two Integer arguments and
20 // returns the sum of those arguments as an Integer.
21 Function Integer sum(Integer num1, Integer num2)
22    Declare Integer result
23    Set result = num1 + num2
24    Return result
25 End Function
```

Program Output (with Input Shown in Bold)

```
Enter your age.
22 [Enter]
Enter your best friend's age.
24 [Enter]
Together you are 46 years old.
```

6.2 Writing Your Own Functions (3 of 3)

Additional concerns

- While you can pass as many arguments into a function, you can only return one value
- Functions simplify code, increase the speed of development, and ease the facilitation of teamwork
- Each function should be flowcharted separately
- IPO (input, processing, and output), can be used to show what a function does

IPO Chart for the getRegularPrice Function

Input	Processing	Output
None	Prompts the user to enter an item's regular price	The item's regular price, as a Real

6.3 More Library Functions (1 of 6)

Mathematical Functions

- Functions typically accept one or more values as arguments, perform a mathematical operation using the arguments, and return the results

Set result = sqrt(16)

– Returns the square root of 16

Set area = power(4, 2)

– Raises the value of 4 to the power of 2

6.3 More Library Functions (2 of 6)

Other Common Mathematical Functions

- **abs** calculates the absolute value of a number
- **cos** returns the cosign of an argument
- **round** rounds to the nearest integer
- **sin** returns the sine of an argument
- **tan** returns the tangent of an argument

6.3 More Library Functions (3 of 6)

Data Type Conversion Functions

- Library functions that convert values from one data type to another
  - **toInteger** converts a real to an integer
  - **toReal** converts an integer to a real
- Real numbers can store integers
- Integers cannot store real numbers
- Type mismatch errors will occur without converting values

6.3 More Library Functions (4 of 6)

Formatting Functions

- Allow to format a number in a certain way
- **currencyFormat** will be used to format a number to a currency

Declare Real amount = 6450.879
Display currencyFormat(amount)

- Display would be **$6,450.88**

6.3 More Library Functions (5 of 6)

String Functions

- Allow for working with strings
- **length** function returns the length of a function
- **append** function joins multiple strings together
- **toUpper** and **toLower** converts a string to upper or lower case
- **substring** can extract a character or a portion of a string out of a string

6.3 More Library Functions (6 of 6)

- **contains** identifies similar strings within two strings
- **stringToInteger** and **stringToReal** converts string that stores a number, to a number data type
- **isInteger** and **isReal** test numbers to see if it can be converted to a string