Component 4/Unit 5-3

Data Type

- Alphanumeric (Character set: A-Z, 0-9 and some special characters)
  - Customer address, name, phone number, Customer ID, Age
- Alphabetic (Character set: A-Z)
  - Alphanumeric data type is used instead of alphabetic most of the time.
- Numeric (Character set: 0-9)
  - Account balance, Age, Count of transactions, Commission rate.

Constants

- Must not change during at least one execution of the program
- Something whose value has been identified as not volatile
  - Examples
    - Number of days in the week
    - Number of months/periods in the business calendar
    - State's legal driving age
    - Pi

Categories of Source Code

- Definitions of variables, constants, and files
- Input and Output (I/O) operations
- Assignment statements
- Arithmetic expressions
- Exclusive options
- Repetitive execution
- Procedures (declaring and invoking)

VBA Code

```
        Definitions of variables, constants, and files (1 – 3)
1       Dim HoursWorked As Single
2       Dim PayRate As Currency
3      Dim GrossPay As Currency
        Procedures (declaring) (4, 9)
4   Private Sub cmdGrossPayMod_Click()
        Input and Output (I/O) operations (5 – 6)
5      PayRate = txtPayRate.Text
6      HoursWorked = txtHrsWrkd.Text
        Assignment statements & Arithmetic expressions (7)
7      GrossPay = PayRate * HoursWorked
        Input and Output (I/O) operations
8       lblGrossPay.Caption = GrossPay
9   End Sub
```


Logic Constructs

- Sequence
- Alternation (Selection, If-Then-Else)
- Iteration (Repetition, Looping, Do loops)
- Concurrency*
- Recursion*

* These two constructs were said to have been proved unnecessary by two Italian Mathematicians, C. Bohm and G. Jacopini in a paper they wrote in 1966[6]. Since then there has been some controversy over that proof[7], but it can be said that for at least most applications solved on a computer that these two logic constructs, although useful and perhaps more efficient at times, are not necessary for the solutions of those computer problems.

Sequence

- Statements are executed in the sequence they are written
- Analogous to following step by step instructions for putting something together
- In programming sequence can be critical for getting the correct results
- When designing a program the correct sequence of statements may not be apparent unless the process is looked at carefully so as to identify all the steps that are necessary

Alternation (Selection, IF-THEN-ELSE)

- Breaks up sequence
- Provides exclusive options
- Comes in many forms.
  - Simple One-tailed
  - Simple Two-tailed
  - Case structure
  - Nested or Dependent
  - Compound

Simple Alternation

(example code is shown in pseudocode)

- One-tailed
  - Example: If light is green
    Go
    End If
- Two-tailed
  - Example: If light is green
    Go
    ElseIf light is red
    Stop
    End If

Simple continued

- Two-tailed
  - Example: If light is green
    Go
    Else
    Stop
    End If

Other Forms of Alternation

- Case
- Nested or dependent
- Compound