**Year 12 Computer Studies**

**Week 10: Lesson Notes\***

**Lesson 81**

**LO**: Differentiating between variables and constants

-VB allows you to set up locations in memory and it gives location name.

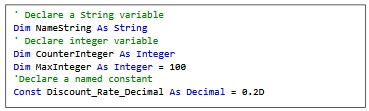
- The memory location is called Maximum Integer.

- Memory locations that hold data that can be changed during project execution are called **variables.**

- Locations that hold data that cannot change during execution are called **constants.**

- VB reserves an area of memory and assigns it a name called an **identifier.**

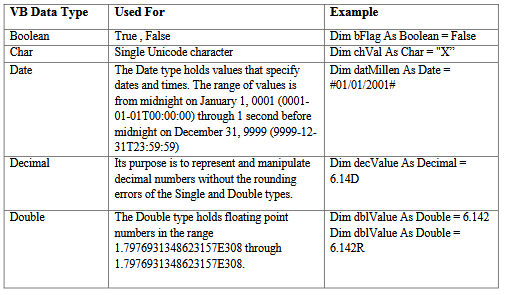
- The declaration statements establish your project’s variables and constants gives them name and specify the type of data type they hold.

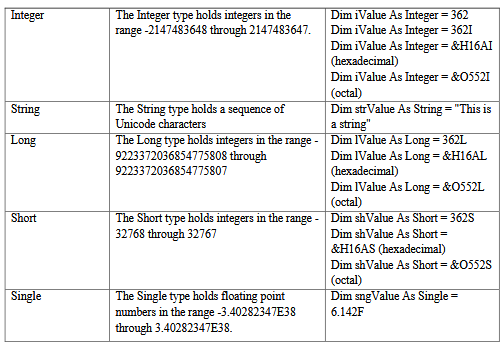


**Lesson 82**

**LO**: Discuss data types

* Data types shows what type of information will be stored in the allocated memory space like name/ dollar amount/ date or a total.
* Data types are referred to as classes and the variables are objects of the class.





**Lesson 83**

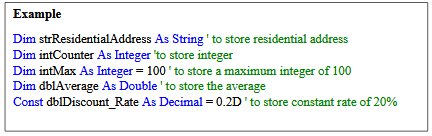
**LO**: Discuss the variable naming conventions

* VB needs identifiers for variables and named constants to follow these rules

1. Names may consist of letters, digits and underscores
2. They must begin with a letter.
3. They cannot contain any spaces
4. They may not be reserved/ key words.

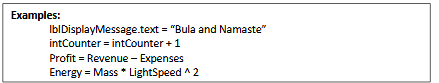
* We should follow some naming conventions:

1. **Identifiers must be meaningful**
2. **Include the class** (data type) of the variable.
3. **Begin with the data type and then capitalize**. Always use mixed case for variables, uppercase for constants.



**Lesson 84**

**LO**: Discuss the assignment statement and expressions

* Assignment statement consist of a variable name followed by the assignment operator (=), followed by some expression.
* 
* The assignment statement stores information
* Statements take up a single line with no terminator.
* Statement can be stacked by using a colon (:) to separate them.
* E. g. StartTime= Now: EndTime= StartTIme + 10
* If statements are very long, it may be continues to the next line using the continuation character, an underscore (\_).



* Comments statements may begin with the keyword **Rem** or a single quote (‘)

**Lesson 85**

**LO**: Discuss the VB operators

* The simplest operators carry out arithmetic operations. These operators in their order of precedence are:

|  |  |
| --- | --- |
| **Operator** | **Operation** |
| ^ | Exponential |
| \*/ | Multiplication and Division |
| \ | Integer division, Mod/ Modulus |
| +- | Addition and Subtraction |

* There are six **comparison** operators in VB

|  |  |
| --- | --- |
| **Operator** | **Comparison** |
| > | Greater than |
| < | Less than |
| >= | Greater than or equal to |
| <= | Less than or equal to |
| = | Equal to |
| <> | Not equal to |

* The result of a comparison operation is a Boolean value (True or False)
* We use three logical operators

|  |  |  |
| --- | --- | --- |
| **Operator** | **Operation** | **Effect** |
| Not | Logical not | Simply negates an operand |
| And | Logical and | Returns a true if both operand is true Else it returns a false |
| Or | Logical or | Returns a true if either of its operands is true, Else it returns False. |

Logical operators follow arithmetic operators in precedence.

**Week 10: Worksheet**

1. Define the following:
2. Identifier

­­­­­­­­­­­­­­­­­­­­­­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Variables \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. Constants \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
3. Define and describe data types.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. List the four rules when naming variables.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. When naming variables, describe the three major rules.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Discuss assignment statement.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. State the operation or the operator for the following table.

|  |  |
| --- | --- |
| **Operator** | **Operation** |
|  | Exponential |
| \*/ |  |
| \ |  |
| +- |  |
| **Operator** | **Comparison** |
| > |  |
|  | Less than |
|  | Greater than or equal to |
| <= |  |
| = |  |
|  | Not equal to |