**Assignment 19**

1. What are the data types used in VBA?

### Ans: VBA Data Types

In computers, we use data types to differentiate between the integers and strings, etc. The data types which we will assign to a variable, decide what should be stored in that variable, i.e., the values that need to be stored in the variable is depends on the data type of the variable. In VBA, data types are divided into two major types:

**1. Numeric Data Type:** Numeric data types are used to perform mathematical operations such as addition, subtraction, etc. It is used to handle the numbers in various representations format. In numeric data type, the **Integral Type** represents only the whole numbers(zero, positive & negative). **Non-Integral Types** represent both the integer and the fractional part.

|  |  |
| --- | --- |
| **Data Types** | **Memory Size** |
| Byte | 1 2yte |
| Integer | 2 bytes |
| Long | 4 bytes |
| Single | 4 bytes |
| Double | 8 bytes |
| Currency | 8 bytes |
| Decimal | 12 bytes |

**2. Non-Numeric Data Type:** Non-Numeric data types are not manipulated by the arithmetic operators. These are comprised of texts, data, etc.

|  |  |
| --- | --- |
| **Data Types** | **Memory Size** |
| String(fixed size/length) | Equivalent to String’s length(in bytes) |
| String(variable length) | String’s length + 10 bytes |
| Boolean | 2 bytes |
| Object | 4 bytes |
| Data | 8 bytes |
| Variant(numeric) | 16 bytes |
| Variant (text) | Text’s length + 22 bytes |

**Note**: If we do not declare any data type, the VBA will be default makes variable as a variant type.

1. What are variables and how do you declare them in VBA? What happens if you don’t declare a variable?

Ans: **Variable** is a named memory location used to hold a value that can be changed during the script execution. Following are the basic rules for naming a variable.

* You must use a letter as the first character.
* You can't use a space, period (.), exclamation mark (!), or the characters @, &, $, # in the name.
* Name can't exceed 255 characters in length.
* You cannot use Visual Basic reserved keywords as variable name.

**Syntax**

In VBA, you need to declare the variables before using them.

Dim <<variable\_name>> As <<variable\_type>>

The declaration is optional in VBA. If you don’t declare a variable and directly use it in the procedure, then it is called an Implicit Declaration. To avoid errors in the code and for better readability, it is recommended to declare a variable explicitly

1. What is a range object in VBA? What is a worksheet object?

**Ans**: Returns a [**Range**](https://learn.microsoft.com/en-us/office/vba/api/excel.range(object)) object that represents a cell or a range of cells.

The default member of **Range** forwards calls without parameters to the [**Value**](https://learn.microsoft.com/en-us/office/vba/api/excel.range.value) property and calls with parameters to the [**Item**](https://learn.microsoft.com/en-us/office/vba/api/excel.range.item) member. Accordingly, someRange = someOtherRange is equivalent to someRange.Value = someOtherRange.Value, someRange(1) to someRange.Item(1) and someRange(1,1) to someRange.Item(1,1).

The following properties and methods for returning a **Range** object are described in the **Example** section:

* [**Range**](https://learn.microsoft.com/en-us/office/vba/api/excel.worksheet.range) and [**Cells**](https://learn.microsoft.com/en-us/office/vba/api/excel.worksheet.cells) properties of the **Worksheet** object
* [**Range**](https://learn.microsoft.com/en-us/office/vba/api/excel.range.range) and [**Cells**](https://learn.microsoft.com/en-us/office/vba/api/excel.range.cells) properties of the **Range** object
* [**Rows**](https://learn.microsoft.com/en-us/office/vba/api/excel.worksheet.rows) and [**Columns**](https://learn.microsoft.com/en-us/office/vba/api/excel.worksheet.columns) properties of the **Worksheet** object
* [**Rows**](https://learn.microsoft.com/en-us/office/vba/api/excel.range.rows) and [**Columns**](https://learn.microsoft.com/en-us/office/vba/api/excel.range.columns) properties of the **Range** object
* [**Offset**](https://learn.microsoft.com/en-us/office/vba/api/excel.range.offset) property of the **Range** object
* [**Union**](https://learn.microsoft.com/en-us/office/vba/api/excel.application.union) method of the **Application** object

1. What is the difference between worksheet and sheet in excel?

**Ans**: Primary difference between these two is **Worksheets property identifies only the type "Worksheets" in excel but Sheets is more general and identifies all the types of sheets** (Worksheets, Charts, Modules / Macro, Dialog sheets). To test this and understand the concept clearly, just open a new workbook (default it displays Sheet1, Sheet2 and Sheet3)

1. What is the difference between A1 reference style and R1C1 Reference style? What are the advantages and disadvantages of using R1C1reference style?

**Ans**: The A1 reference style is the default reference style notation in Excel, and if you have used Excel even for a few hours, I’m assuming you have already used it.

In layman’s terms, a reference style is a style you use to refer to the cells in Excel.

When using the A1 reference style, you would refer to any sale by first specifying the column alphabet/letter for that cell, followed by the [row number](https://trumpexcel.com/number-rows-in-excel/) for that cell.

R1C1 is the other type of reference style that you can use in Excel.

Here, R refers to the Row and C refers to the column, so R1C1 would refer to the cell in the first row and first column.

### A1 vs R1C1 Reference Style

In the **A1** style reference, the columns are marked as letters **(A, B, C),** and rows are marked as numbers **(1,2,3).** On the other hand, both the rows and columns are marked as numbers **(1,2,3)** in the **R1C1** reference style.

The cell address of the very first cell in an Excel sheet is **A1** in **A1** style reference.

On the contrary, the cell address of the very first cell in an Excel sheet is **R1C1** in the **R1C1** style reference.

6. When is offset statement used for in VBA? Let’s suppose your current highlight cell is A1 in the below table. Using the OFFSET statement, write a VBA code to highlight the cell with “Hello” written in it.

A B C

1 25 354 362

2 36 6897 962

3 85 85 Hello

4 96 365 56

5 75 62 2662

**Ans**: Private Sub CommandButton1\_Click()

Range("C3").Offset(0).Select

End Sub