Excel Macros (VBA)– Chapter 2

Variables, Data types & Constants

Chapter 2 :Variables, Data types & Constants

- What is a Variable & Why do we need variables
- How to declare a Variable
- Rules to be followed for a Variable name
- Implicit and Explicit declaration
- Forcing Variable Declaration (Option Explicit)
- Scope of a Variable
- Assigning values to Variables
- Data Types
- Constants

Chapter 2 :Variables, Data types & Constants

What is a Variable & Why do we need Variables:

VBA stores data in memory using a Variable. A variable is a name given by you, to which you assign a piece of data that will be stored in computer's memory and that can be retrieved when you need to later in the Macro.

Variables hold values of different Data types that are specified when the variable is declared.

How to declare a variable:

You declare a variable with declaration statement which contains four key words in particular order

1. The **Dim** statement (VBA's abbreviation for Dimension), which all variable declarations start with

2. The name of your variable, which you create

3. The Key word **as**

4. The type of data being stored (Data type)

Eg: Dim myname as String

Rules to be followed for a Variable name:

- They must begin with an alphabet
- They must contain only alphabets, numbers or the underscore (_) character – no spaces
- They must not exceed 40 characters
- They must not be a reserved word

Implicit and Explicit declaration:

Implicit Declaration:

You do not have to declare a variable before using it. You can just include the statement eg: tempval = 6

However, a problem with doing this is that it can lead to errors in your code when you misspell the name of the variable in a later statement. For example, if you refer to it as temval instead of tempval, VBA assumes that temval is a new variable and assigns it as such. The old variable tempval is still there but no longer being used. This leads to errors during run time and takes some time to correct your code.

Explicit Declaration:

To avoid the problem of misnaming variables, you can stipulate VBA to generate error message (while compilation) whenever it encounters an undeclared variable. To do this we need to give the below statement in the declaration section of the module

Option Explicit

This prevents implicit declarations of variables. Now you have to declare a variable before using it.

Using option explicit is the best practice and helps to avoid run time errors.

Chapter 2 :Variables, Data types & Constants

Scope of a Variable:

Local Variable:

If you declare a variable within a procedure, only code within that procedure can access that variable. The scope is local to that procedure. Local variable remain in existence only as long as the procedure is executing.

Module level variables:

A module level variable is declared in the declaration section of the module instead of with in the procedure. Module level variables are available to all procedures within that module but not to the rest of the application. Module level variables remain in existence for the life time of the application and preserve their values.

Chapter 2 :Variables, Data types & Constants

Scope of a Variable:

Global Variable:

Global variables are declared in the declarations part of a module with the Global statement and they can be accessed by any code with in the application. Global Variables exist and retain their values for the life time of the application.

Assigning values to Variables:

DEMO

Chapter 2 :Variables, Data types & Constants

Data Types:

A variable can be given a data type that determines the type of data it can store. This can have an effect on the efficiency of your code. If you don't specify a data type then the default data type is Variant

Variant:

A Variant can store all kinds of data whether it's a text, number, date …A Variant variable can freely change its type at run time which is not possible for other data types.

However, using Variant data type reduces the speed of the program. Also Variant data type occupies more memory compared to the other data types.

Other Data Types:

DATA TYPE	DESCRIPTION	MEMORY
Boolean	True or False; 1 or 0; On or Off.	2 bytes
Byte	An integer from 0 to 255.	1 byte
Currency	A positive or negative number with up to 15 digits to the left of the decimal point and up to 4 digits to the right of it.	8 bytes
Date	A floating-point number with the date to the left of the decimal point and the time to the right of it.	8 bytes
Single	A floating-point number ranging in value from –3.402823E38 to –1.401298E-45 for negative values and from 1.401298E-45 to 3.402823E38 for positive values.	4 bytes
Double	A floating point number ranging in value from –1.79769313486231E308 to –4.94065645841247E-324 for negative values and from 4.94065645841247E-324 to 1.79769313486232E308 for positive values.	8 bytes
Integer	An integer ranging from –32,768 to 32,767.	2 bytes
Long	An integer ranging from –2,147,483,648 to 2,147,483,647.	4 bytes
String	There are two kinds of strings: variable-length and fixed-length. A variable-length string can contain up to approximately 2 billion characters. A fixed-length string can contain 1 to approximately 64,000 characters.	For a variable-length string, 10 bytes plus storage for the string For a fixed-length string, the storage for the string
Variant	Data type for all variables that are not explicitly declared as some other type, which can contain any kind of data except fixed-length String data.	For containing numbers, 16 bytes For containing characters, 22 bytes plus storage for the characters

Chapter 2 :Variables, Data types & Constants

Other Data Types:

DATA TYPE	DESCRIPTION	MEMORY
Boolean	True or False; 1 or 0; On or Off.	2 bytes
Byte	An integer from 0 to 255.	1 byte
Currency	A positive or negative number with up to 15 digits to the left of the decimal point and up to 4 digits to the right of it.	8 bytes
Date	A floating-point number with the date to the left of the decimal point and the time to the right of it.	8 bytes
Single	A floating-point number ranging in value from –3.402823E38 to –1.401298E-45 for negative values and from 1.401298E-45 to 3.402823E38 for positive values.	4 bytes
Double	A floating point number ranging in value from –1.79769313486231E308 to –4.94065645841247E-324 for negative values and from 4.94065645841247E-324 to 1.79769313486232E308 for positive values.	8 bytes
Integer	An integer ranging from –32,768 to 32,767.	2 bytes
Long	An integer ranging from –2,147,483,648 to 2,147,483,647.	4 bytes
String	There are two kinds of strings: variable-length and fixed-length. A variable-length string can contain up to approximately 2 billion characters. A fixed-length string can contain 1 to approximately 64,000 characters.	For a variable-length string, 10 bytes plus storage for the string For a fixed-length string, the storage for the string
Variant	Data type for all variables that are not explicitly declared as some other type, which can contain any kind of data except fixed-length String data.	For containing numbers, 16 bytes For containing characters, 22 bytes plus storage for the characters

Chapter 2 :Variables, Data types & Constants

Constants:

A Variable's value may often change during the execution of the Macro. However a constant is a value in a Macro that does not change during the execution of the Macro. Essentially, constants are variables that do not change.

A constant is declared by using the key word const

Const pi as double = 3.14

A constant is used to store a fixed value and refer to that value multiple times in your macro with out hard-coding the actual value every time in the macro.

Re-Cap

- What is a Variable & Why do we need variables
- How to declare a Variable
- Rules to be followed for a Variable name
- Implicit and Explicit declaration
- Forcing Variable Declaration (Option Explicit)
- Scope of a Variable
- Assigning values to Variables
- Data Types
- Constants

Thank You

Email us – support@intellipaat.com

Visit us - https://intellipaat.com