VB का परिचय –

VB का उद्‌भव विन्डोज़ पर आधारित एप्लिकेशन की संरचना एवं विकास के लिए हुआ। विन्डोज़ पर आधारित एप्लिकेशन की प्रोग्रामिंग में किसी इण्टरफेस को प्राप्त करने के लिए बहुत कड़ी मेहनत एवं बहुत ज्यादा समय लगाना पड़ता था परन्तु VB की मदद से यह काम बहुत आसान हो गया। विजुअल बेसिक वर्ष 1991 में अस्तित्व में आया। VB को अस्तित्व में लाने के लिए जो सामान्य idea है वह है इसका रैपिड एक्शन डेवलपमेण्ट इनवायरन्मेण्ट। इसके द्वारा डेवलपर किसी भी इण्टरफेस को बहुत ही आसानी से बना सकता है। बिना इसको जाने भी यह कैसे बन रहा है। जबकि अन्य प्रोग्रामिंग लैंगवेजेज़ में किसी इण्टरफेस को बनाने में ज्यादा समय लगता है। सॉफिस्टिकेटेड प्रोफेशनल एप्लिकेशन बनाने के लिए VB एक आइडियल प्रोग्रामिंग लैंगवेज है। VB हमे रोबस्ट और पारफुल एपलिकेशन्स के लिए ग्राफिकल यूज़र इण्टरफेस बनाने में मदद करता है। ग्राफिकल यूज़र इण्टरफेस किसी भी यूज़र को किसी कार्य को आसानी से करने में चित्रों के माध्यम से मदद करता है।

1. **<u>Visual Design :</u>**
   VB में Drawing की सुविधा होती है, जिसकी सहायता से हम किसी भी Application का Front End तैयार करते है या User Interface तैयार करते है।

2. **<u>Event Driven Programming :</u>**
   जब आप VB में प्रोग्राम करते है तो आप यह निर्धारित करते है कि किस प्रकार से कोई Application प्रयोगकर्ता के साथ Interact करें। दूसरे शब्दों में आप यह निर्धारित करते है कि प्रत्येक Controls प्रयोगकर्ता के action पर कैसे कार्य करे। जैसे माउस Click आदि। इसे Event Driven Program कहा जाता है।

VB के मुख्य गुण –

1. उपयोग के लिए Icon और Picture होते है।
2. Mouse Click और Keyboard का प्रयोग करने पर कार्य सम्पादित होते हैं।
3. ActiveX का सहयोग।

आवश्यक Hardware तथा Software:

<u>Processor</u>: VB S.01/6.0 के लिए कम से कम 150 MHz Processing speed की आवश्यकता होता है। सामान्यतः 300 MH3 का Processing Speed होना चाहिए।

<u>Memory Size :</u>
VB5.0 के लिए कम से कम 2GB हार्ड डिस्क Space तथा 16 MB RAM होना चाहिए और VB 6.0 के लिए 4-3 GB और 68MB की RAM होनी चाहिए।

Operating System :

VB 5.0/6.0 के लिए Windows 95 और इससे ऊपर के Versions होने चाहिए।

Objects & Controls :

V.B में जो Controls में होते है तो उन्हें Class कहा जाता है, जब इन्हें फार्म में उपयोग करते है तो उन्हें Object कहा जाता है। कमान्ड बटन text box, आदि इन्हें क्लास कहा जाता है, जब इनको फार्म पर लाते हैं तो अब जो फार्म पर Control दिखाई देता है उसे Object कहा जाता है।

Properties :

Object के बदलाव वाले गुण को Properties कहा जाता है जैसे– Color, Size & Caption आदि। Property को तीन भागों में बाटा गया है।

1. Design-time
2. Run- time
3. Design & Run Time

Methods :

Method एक क्रिया होती है जिसे किसी Object पर प्रयोग किया जाता है या Object के द्वारा प्रयोग किया जाता है। method के द्वारा किसी Object के State को नहीं बदला जा सकता है। Object की किसी Property जैसे hide, move, open आदि को किसी भी order में access किया जा सकता है, जबकि method को एक specific order में ही execute किया जा सकता है।

Events :

VB में Event का मतलब होता है घटना। जब आप Object पर Click करते है तो कोई घटना घटित होती है। यह घटना mouse click & keyboard के key press करने पर होती है तो इसे Event कहते है।

Forms :

VB एक Work Space देता है जहाँ पर आप कन्ट्रोल draw करते है और Use करने लिए Front end तैयार करते है। इसे Forms कहते है।

IDE Visual Basic के Element :

इसके निम्नलिखित Element है।

Tool Bar, Menu bar, tool box, project explorer windows, properties window, forms lay out window, form designer.

Constant :

ऐसी quantities जिनकी वैल्यु Programm execution के समय नही बदली जा सकती है उन्हे constant कहते है।

Syntax :

Const < Constant Name > [Astype] = <Value> e.g. const dimension as Single = 22.41

Variable :

Programming Language में Variable ऐसी quantities होती हैं जिनकी वैल्यु Programm execution के समय बदलती रहती है। Variable वैल्यु को Program execution के दौरान अस्थायी रूप से Store करता है।

Variable के नामकरण का नियम :

वैरियेबल का नाम किसी Letter से स्टार्ट होना चाहिए। इसकी अधिकतम लम्बाई 255 characters हो सकती है। इसमें कोई Special character नहीं होना चाहिए। किसी Variable का नाम V.B. के Keyword के जैसा नहीं होना चाहिए।

Declaring Variables :

Suntax – Dim <Variable Name> [as > type>]

Ex.- Dim num as integer
 Dim names as string

Print Statement :

Print Statement का प्रयोग करके आप किसी को भी message को Form के upper Left corner पर Print करा सकते हैं।

Print with Comma :

यह कमाण्ड स्क्रीन पर स्तम्भ में किसी message के Variable को Print करने के लिए प्रयोग किया जाता है। यह Number को Left तथा अक्षरों को Right Position से प्रिन्ट करता है।

Print with Tab :

जब कालम को tabular form में प्रिन्ट करना हो तो हम Print Tab का प्रयोग करते हैं। अर्थात् यदि किसी मैटर को इच्छित जगह देकर Print करना हो तो इस कमान्ड का प्रयोग किया जाता है।

Ex. – Print tab (20) "We", tab (30) "Love", tab (40) "India",

Operator In Visual Basic :

Operator का प्रयोग करके एक या एक से अधिक Value को Single Value में परिवर्तित किया जाता है। VB में निम्नलिखित Operators का प्रयोग किया जाता है।

1. Arthmetic Operators - -, +, *, \, %

 Num 1 = 1000.234

Num 2 = 10.2364
Result = Num1\ Num 2
^-exponent 2^3=8
Mod (modules) 12 mod 5=2
Assignment =
Comparison operator <, >, =, < = , < >,

Logical Operator:

And operator 1 < 2 and 3 > 4 = |
Or operator 1>2 or 2<4 => 1
Not operator not 2<4 => 0

Concatenation & तथा + का प्रयोग करके आप दो Strings को जोड़ सकते है।

A = "34"
B = "54"
Print A+B
OR
Print A&B

VB के लाभ:

1. V.B. event driven programming Language है।
2. V.B. object oriented के नियमों का पालन करती है।
3. V.B. के प्रयोग से आप Application Software का निर्माण करते है।

Design & Build User Interface :

V. B. में User के interface के लिए सबसे मुख्य भाग Form होता है। Form एक window है जो विभिन्न Object को धारण किये रहती है। वे सभी Object जो फार्म पर होते है Control कहलाते हैं। Labels, text Box, command Buttons और अन्य Control फार्म डिज़ाइनर के द्वारा फार्म पर लाये जाते है।

Code लिखना:

V.B. event driven programming के नियम का पालन करती है। कोड लिखने के लिये आपको जिस Event से सम्बन्धित कोड लिखना होता है, उससे सम्बन्धित Control पर Double click किया जाता है। अब हमारे सामने एक कोड window खुल जाती है और इसी विन्डो पर हम कोडिंग करते है।

Testing and Debugging :

जब हम VB में Programming करते है तो अनेकों लाइनों में कोड लिखते है। अब कोड लिखने के बाद प्रोग्राम को Test किया जाता है। यदि हमारे कोड में किसी तरह की कोई गलती है तो प्रोग्राम रूक जाता है। अब हम Debug करके इस गलती को ठीक करते है और फिर प्रोग्राम को Run करते है।

मुख्य Controls – Labels:

Label Control का प्रयोग करके हम किसी भी Information को दिखा सकते है। यह केवल पढ़ने के लिए होता है।
Label & Text दोनों एक ही जैसा होता है लेकिन अन्तर इतना है कि Label पर लिखा हुआ Text केवल पढ़ने के लिए होता है जबकि Text Box में लिखे हुए Text को हम Read & Write दोनों कर सकते है।

Text Box :

सामान्यतः सभी Application में text box का प्रयोग किया जाता है। Text Box में User Input करता है और आउटपुट भी प्राप्त करता है। Text Box में सभी आवश्यक events होते हैं जिसका उपयोग Application में किया जाता है। Text Box के पास सभी आवश्यक Property होती है। आप Text Box में Scrollbar जोड़ सकते हैं, फॉन्ट की साइज घटा तथा बढ़ा सकते है। इसका Back Color भी बदल सकते हैं। इसकी Multiline Property को Enable करके आप Text Box में कई लाईनों में Typing कर सकते हैं।

List Box & Combo Box :

User के सामने Data को List के रूप में दिखाने के लिए List Box का प्रयोग किया जाता है। हम List Box में दिखाई देने वाले Items में से किसी आइटम को Select कर सकते है। यदि Item की संख्या List Box की साइज से ज्यादा हो जाती है तो एक Scroll Bar स्वतः जुड़ जाता है। User, List Box के आइटमों को Edit नहीं कर सकता। Combo Box, list की ही तरह होता है। Combo box में List Box तथा Text Box दोनों के गुण होते है। हम Combo box के List में से किसी भी Item को Select कर सकते है तथा नये Item को जोड़ भी सकते है।

Radio Button & Checkboxes :

Options Button को रेडियो बटन कहते है। यदि किसी ग्रुप में एक से अधिक Radio Bottons हैं तो उनमें से कोई एक ही बटन "On" condition में होगा और अन्य Off Condition में होगें यदि किसी form में एक से अधिक Radio Buttons का ग्रुप लेना हो तो उसे अलग–अलग From Control में draw करें। Check box Contral Radio Button की ही तरह होता है, लेकिन इसमे ंएक समय में एक से अधिक "On" Condition में हो सकते हैं।

Timer Control :

यह Control time bomb की तरह होता है जो किसी घटना को निश्चित समय के बाद घटित कराता है। अर्थात् इस Control पर लिखा हुआ कोड कुछ निश्चित समय के बाद Execute होता रहता है।

Tool Box में विभिन्न 26 tools :

Painter, Picture, Label, Text Box, Frame, Command button Check box, Option button combo box, list box, horizontal scroll bar, vertical scroll bar, timer, drive list box directory list box, Fole list box, shape, line, image Data, Common dialog, grid, blist (data-bound list box) DB Combo (data – bound Combo box) DB gid (Data Bound Grid) ole container.

Tool Box से Custom Control को जोड़ना/हटाना –

सामान्यतः Custom Control Third Party Control होता है जिसको किसी बाहरी Vendor से खरीदा जाता है। e.g. Sheridan tabbed control इत्यादि। आप अपना Custom Control भी बना सकते है।

Visual Basic में Coding :

Form पर जितने भी Object होते है उनके Event पर हम Coding करते है। यदि किसी Object पर Double Click करते है तो एक नया Window खुल जाता है जिसे Code Window कहते है। इस Window पर कोड लिखा जाता है। प्रत्येक Code का एक Proper Syntax होता है।

Start up Form : -

किसी भी Application Form में एक से अधिक Form हो सकते है। जब आपका Application Start होता है तो main form load हो जाता है। इस कार्य के लिए आप पहले यह Select करते है कि जब आप अपना Application rum करें तो कौन सा Form सब से पहले Display हो।

Project >> Properties :

Star up form list से इच्छित Form Name को Select करते हैं और OK पर Click कर देते है। अब आप इस Form की विषेशता तय कर सकतें हैं।

Loading : Showing, Hiding form :

किसी form को दिखाने के लिए तीन Statement और Method है।

1. **Not Loaded :** Disk से फार्म Load हो लेकिन किसी Resource को उपयोग न करें।
2. **Load हो लेकिन दिखाई न दें –**
   Form मेमोरी में Load हो आवश्यक Resources का प्रयोग करे और दिखाई न दे।
3. **Load हो और दिखाई भी पड़े –** Form दिखाई पड़े और user इस फार्म से Interact भी हो।
4. **Load हो और Unload हो –** Load और Unload का प्रयोग करके आप फार्म को क्रमशः Load या Unload कर सकते है।

Syntax :

फार्म Load के लिए –

Load < Form Name >

Syntax :

फार्म Unload के लिए –

Unload < Form Name >

Unload Form / Unload me -

उपर्युक्त Syntax में Form Name, Variable होता है। जिस फार्म को Load या Unload करना होता है उस Form का नाम <Form Name> की जगह पर लिख देते है। Load Statement Form को नहीं दिखाता है Form को दिखाने के लिए Show method का प्रयोग किया जाता है। जब कोई Form Load हो जाता है तो यह आवश्यक Resources का प्रयोग करता है इसलिए जिस Form की आवश्यकता नहीं हो उसे Unload कर देना चाहिए। जब आप Form को Unload करते है तो यह सभी Resources को System को वापस कर देता है।

Showing Forms :

Form को दिखाने के लिए Show Mathod का प्रयोग करते है। यदि कोई form load हो जाता है लेकिन दिखाई नहीं दे रहा है तो Show method फॉर्म को सभी खुले हुए window के ऊपर खोल कर ला देता है। यदि form load नहीं है तो show method form को Load करता है और इसे दिखाता भी है।

Syntax :

< Form Name > Show [Mode]
0 Modules (default), 1- Module

Hiding form (छुपाना) :

यदि आपके Application में एक से ज्यादा Form खुले है और आप चाहते है कि उनमें से कुछ Form Hide हो जाये तो Hide method फॉर्म को छिपा देता है।

Syntax :

< Form Name > hide
(me.hide)

Scope of Variable :

Scope of Variable से यह निर्धारित किया जाता है कि किसी भी Variable को अपने Application के किसी Section में Access किया जा सकता है या नहीं। किसी भी Variable को तीन Scopes होते है।

1). Local – ऐसे Variable जो किसी Function या Proceeding के लिए विशेष रूप से declare किये जाते हैं Local Variable कहलाते हैं। इसे केवल function or procedure के भीतर ही Access किया जा सकता है। Local Variable declare करने के लिए Private Kesword का प्रयोग करते है।

2). Form wise/ Module wise – ऐसे Variable जो form पर किसी भी function से Access किये जा सकते हैं Form wise/ Module wise variable कहलातें है। ऐसे variable from wise declaration भी कहलाते हैं।

3). Global – ऐसे Variable जो प्रोजेक्ट में उपस्थित एक से अधिक Form के द्वारा Access किये जा सकतें है Global variable कहलातें है। इस प्रकार के declaration को Global Declaration कहते है। इस प्रकार के Variable को Declare करने के लिए Public Keyword का प्रयोग किया जाता है।

e.g. Public num as Integer.

इस Variable को Code window के General area में declare करते है।

User defined data type :

V.B.यह सुविधा प्रदान करती है कि युजर अपना खुद का Data Type तैयार कर सकता है।

Syntax :

```
Type <Variable Name>
        <Variable 1> as type
        <Variable 2> as type
End Type
```

Decision Making Statements :

Control Structure :

VB दो प्रकार के Control Structure प्रदान करता है।

1. If Structure
2. Select Structure

Structure :

यह निम्नलिखित रूप से हो सकता है।

If …………… then

If ……………. then else

Nisted if ……… then ………else

Multiple else if

If- Then- Else :

Decision लेने के लिए VB if, Then तथा Else की सुविधा प्रदान करता है।

उदा0– यदि दी हुई Condition सही है तो If part Execute होगा अन्यथा Else Part Executed होगा। यदि दी गयी Condition गलत हो जाये और Else Part

को परिभाषित नहीं किया गया हो तो Control अगले Statement में चला जायेगा।

उदाहरण :
If x = 0 Then msgbox “You enterd zero” else msgbox “You didn’t enter a zero".

```
If < condition> then
[Statement 1]
[Statement 2]
End if
```

If – Else – If Structure :

```
If < condition > then
[Statement 1]
[Statement 2]
```

Else – If – < Condition > :

```
[Statement 1]
[Statement 2]
End If
```

Immediate if :

जल्द से जल्द decision लेने के लिए iif ( ) function का प्रयोग करते है। Condition Test करने के बाद iif function दो Part में से किसी एक Part को वापस करता है। पूरे Expression के दो भाग होते है। एक If भाग और else भाग।

Example :

```
Result = iff (div ) = 60, "First", Second''
```

Select Case Statement :

इस Statement के द्वारा दिए हुए Expression के Value के आधार पर अनेको Statement में से कोई एक Execute होता है।

Syntax :

```
Select case <test expression>
            Case expression list
                  [statements]
            Case expression list
                  [statements]
            -----------------------
            -----------------------
            Case else
                  Statements
            End select:
```

Case else Statement तभी Execute होता है जब दी हुई Condition सही नहीं होती है।

<u>लूप संरचना :</u>

<u>लूप व्यक्तिकरण –</u>

लूप व्यक्तिकरण आपको यह सुविधा प्रदान करता है कि आप एक या एक से अधिक लाइनों में लिखे कोड को क्रमशः बार–बार क्रियान्वित कर सकते है।
VB निम्नलिखित Loop संरचना को Support करता है।

1. Do.............loop
2. While.......... wend
3. For............. Next

<u>Do ....Loop-</u>

Do ....Loop लिखे गये कोड को बार–बार क्रियान्वित करता है जब तक की दी गयी Condition सही रहती है।

Do ....Loop Statement के दो चर होते है। एक while तथा दूसरा Until

नियम–

```
1) Do
Statement ........
Body ..........
Loop while <Condition>

2) Do While<Condition>
Statement ........
Body ..........
Loop

3) Do Untill <Condition>
   Body
   Loop
```

Dim num as integer num = text1 text
Label1-visible = false
Text1- visible = false
I = 1
Do while i<=10
t = num

<u>FOR NEXT :</u>

For Next Loop का उपयोग हम उस जगह पर करते है, जहाँ हमको पता होता है कि हमारे द्वारा लिखे गये प्रोग्राम के कोड को कितनी बार *Run* करना होता है।

For <Variable> - <Start> to <end> Statement.

WHILE ...... WEND :
यह लूप किसी भी Statement को तभी क्रियान्वित करता है, जबकि दी गयी शर्त सही है।

NESTED LOOP :
जब हम एक लूप के भीतर दूसरा लूप और दूसरे लूप के अन्दर तीसरा लूप यानी लूप के अन्दर लूप का निर्माण करते हैं तो यह Nested Loop कहलाता है।

DO- WHILE LOOP :
Do- While लूप के अन्दर पहले कोड Run करता है फिर दी गयी शर्त चेक होती है।

अर्थात– यदि यदि दी गयी शर्त गलत हो तो भी आपका लिखा गया प्रोग्राम कम से कम एक बार Execute होगा।

ऐरे –
ऐसे Variable जो एक से अधिक Value को अपने अन्दर संग्रहित कर सकते है उन्हे ऐरे कहा जाता है। सामान्यत: ऐरे अपने अन्दर एक ही प्रकार के Data की Entry की सुविधा देते है। लेकिन यदि इसके डेटा प्रकार का निर्धारण कर दिया जाय तो ऐरे अपने अन्दर विभिन्न प्रकार के डेटा टाईप को भी रख सकता है।

Syntax :
Dim arrayname (Subscript) as Type

उदाहरण –

Dim name (10) as integer –

Collection
कीवर्ड Collection के द्वारा हम नया Collection बनाते है।

Collection में किसी Item को जोडना –
Add Method का प्रयोग करके हम नये Item को Collection में जोड़ते है।

<Collection Name> add <Value>, Key [before/ after]

Collection से किसी Item को हटाना –
Remove method के द्वारा Collection से छात्रा Jaya को हटाना

College remove "Jaya"

या

College- remove 2

Returning Items in a Collection –
Item method collection के किसी आइटम की वैल्यु को देता है।

M = College Item ("Jaya")

M = College Item (4)
M = College ("Jaya")

Counting a Collection –

Collection किसी के सभी Items को Scan करने के लिए VB for each------next संरचना का प्रतिपादन करता है।

Syntax :

```
For each <Item> in <Collection>
Statement
Next
```

Procedure :

Procedure हमें ऐसा वातावरण प्रदान करते हैं जिससे हम Complex Program को छोटे Logical Unit में बना सकते है।

इस प्रकार से हम अपने प्रोग्राम को छोटा तथा खण्डो में विभाजित कर लेते है और प्रत्येक खण्ड प्रोग्राम को चलाने के लिए जिम्मेदार होते है। इन खण्डों को Procedure कहते जाता है।

Procedure के प्रकार –

Procedure दो प्रकार के होते है।

1. Sub procedure / Subroutines
2. Function Procedure

Sup- procedure :

Sub procedure, Statements का Block होता है जिसका कोई निश्चित कार्य होता है। Sub procedure कोई वैल्यु वापस नहीं करता है।

Subprocedure की Coding Sub........End Sub को बीच लिखी जाती है और सभी Procedure को Sub-direct कहते है। यदि आप Sub-routine से हमेशा के लिए बाहर होना चाहते है तो Exit Statement का प्रयोग करेगें।

Syntax :

```
[Private/ Public] [Static] Sub <Procedure Name>
<Arguments>
Body of Procedure
End Sub
```

Visual Basic दो प्रकार के Sub Procedure को Support करता है।

<Event Subprocedure>

<General Sub procedure>

Event Sub procedure :

यह Procedure event के साथ जोड़ा जाता है। आप कन्ट्रोल और फार्म के लिए Event तथा Procedure लिख सकते है।

Event Procedure को Event handler कहा जाता है।

जब कोई Subject event को पहचानता है, तो इस event से सम्बन्धित event procedure स्वतः क्रियान्वित होते है।

Syntax :

Private Sub < Control Name>_<Event Name> ([<Argument List>])
Body of procedure
End Sub.

General Sub procedure:

जनरल Procedure हमारी मदद करता है कि किस प्रकार से किसी कार्य को किया जाये जो किसी Specific Event से सम्बन्धित न हो।
एक बार जनरल Procedure परिभाषित कर दिया तो इसे अपने Application में बुलाया (call) जा सकता है।
यदि एक से अधिक Event के लिए Code common है तो हम एक अलग से Procedure बनाते है, जो कि General Procedure होता हैं।

Access Specifer :

Access Specifier तीन प्रकार के होते हैं।

1. Private
2. Public
3. Static

इन Specifirs के द्वारा हम किसी भी Procedure का Scope निर्धारित करते है। Bydefault Subprocedure Public होता है जिसको हम अपने Application में कही पर भी बुला सकते है।

Static -

इस Keyword का प्रयोग करके किसी Variable को program run होने के दौरान किसी निश्चित मान के लिए प्रतिबन्धित कर दिया जाता है चाहे जितनी बार भी किसी Procedure को Call किया जाये।

Sub :

यह Keyword Compiler को यह बताता है कि यहाँ पर एक Procedure परिभाषित किया गया है।

Procedure Name :

Procedure का नाम रखने के लिए निम्नलिखित नियम का पालन करना पड़ता है।

1. पहला अक्षर कोई Alphabet या Under Score होना चाहिए।
2. Special Character जैसे– +, -, *, space,/,\,.,& का प्रयोग नहीं किया जाता है।
3. Procedure का नाम अर्थपूर्ण होना चाहिए।

Arguments :

इसमें Event पर आधारित Arguments की लिस्ट समाहित रहती है। आप अपने एप्लीकेशन में Event Procedure के द्वारा General Procedure को बुला सकते है।

उदा0 –

```
Sub show – Date (   )
Msg. Box – Date (   )
End Sub.-
```

Function Procedure :

Function Procedure, Sub. Procedure की ही तरह होता है, लेकिन यह रिजल्ट भी देता है।

Function Statement को Function तथा End Function के बीच में लिखते है।

Creating Procedure :

कोड–विन्डो पर टाइप करके आप Procedure बना सकते है और Procedure को Tools >> Add Procedure के द्वारा भी बना सकते है।

Statement :

Procedure or Fanction के भीतर या बाहर जो कोड लिखा जाता है, उसे Statement कहते है।

System Defined Function:

VB में बहुत सारे Function पहले से बने –बनाये मिलते है, जैसे– Scr, Chr, Date, abs, आदि जिनका प्रयोग हम अपने विशिष्ट कार्यो के लिए कहते है।

Syntax :

```
[Private/ Public] [Static]
Function <Function Name> ([<Argument>])
[As type]
Statement
End Function
```

उदाहरण –

```
Function absl (num As doube) As double
If abs(num)>0 Then
      absl=num
Else
      absl = - num
Endif
End function
```

Input box Function –

Input box का प्रयोग करके आप किसी बटन पर Click करें तो एक Box आयेगा ताकी जिसमें आप कुछ Input कर सकें।

Syntax :

Input Box (Prompt) [didl]
[default] [X- Position] [Y position]