

# Visual Basic /VB.NET

PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

## UNIT -2

### Variable & variable declarations-

Variable यानि स्टोरेज एरिया का नाम जिसे हमारे program यूज़ कर सके। VB.Net में सभी Variable का स्पेसिफिक type होता है, जो Variable का size और memory layout, तय करता है।

VB.Net की कुछ बेसिक data types निम्नलिखित है:

डेटा टाइप	उदाहरण
Integral types	SByte, Byte, Short, UShort, Integer, UInteger, Long, ULong and Char
Floating point types	Single तथा Double
Decimal types	Decimal
Boolean types	इसमें सिर्फ दो वैल्यू होती है true or false
Date types	Date

### Variable Declaration-

Dim Statement का उपयोग variable declaration और एक या ज़्यादा variables को स्टोर करने के लिए किया जाता है |

Dim का पूरा नाम dimension है |

Dim Statement का उपयोग module, class, structure, procedure या block level पे होता है.

# Visual Basic /VB.NET

PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

Variable declaration की Syntax निम्नलिखित है:

**[< *attributelist*> ] [ *accessmodifier* ] [[ *Shared* ] [ *shadow* ] [ *Static* ]]  
[ *ReadOnly* ] *Dim* [ *WithEvents* ] *variablelist***

जहाँ ,

- *attributelist* का मतलब होता है कि *attributes* की लिस्ट जो वेरिएबल पे अप्लाई होती है यह फील्ड ऑप्शनल है.
- *accessmodifier* का उपयोग *variable* का *access levels* डिफाइन करने के लिए होता है , इसकी वैल्यू – Public, Protected, Friend, Protected Friend and Private दीजाती है य .ह फील्ड ऑप्शनल है.
- *Shared* का उपयोग ऐसे *shared variable* को डिक्लेर करने के लिए होता है,जो किसी स्पेसिफिक *class* या *structure* के *instance* के साथ यूज़ नहीं होता है पर सभी क *class* या *structure* के साथ होता हैयह. फील्ड ऑप्शनल है.
- जिस *variable* के साथ *Static* का उपयोग होता है उस *variable* की वैल्यू ,जिस *procedure* में *variable* डिक्लेअर हुआ है उस *procedure* के *termination* बाद भी स्टोर्डरहेगीयह. फील्ड ऑप्शनल है.
- जिस *variable* के साथ *ReadOnly* का उपयोग होता है उसकी वैल्यू को आप चेंज नहीं कर सकते यह. फील्ड ऑप्शनल है.
- *WithEvents* दर्शाता है की वेरिएबल का उपयोग *events* को *respond* करने के लिए होता है | यह फील्ड ऑप्शनल है.
- *Variablelist* डिक्लेर किये हुए *variable* की लिस्ट प्रोवाइड करता है.

*Variablelist* में प्रत्येक वेरिएबल का *syntax* निम्नलिखित होता है ,

***variablename* [ ( [ *boundslist* ] ) ] [ *As* [ *New* ] *datatype* ] [ = *initializer* ]**

जहाँ,

- *variablename* यानि *variable* का नाम।
- *boundslist* का उपयोग *array variable* के *dimension* के *bound* को डिक्लेर करने के लिए होता है।

# Visual Basic /VB.NET

PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

- New का उपयोग class के नई instance को क्रिएट करने के लिए होता है
- datatype का उपयोग Variable के data type को डिक्लेर करने के लिए होता है।

Dim EmployeeID As Integer

Dim EmployeeName As String

Dim Salary As Double

Dim count1, count2 As Integer

Dim status As Boolean

Dim exitButton As New System.Windows.Forms.Button

Dim lastTime, nextTime As Date

## Data Types-

हर Programming language में डाटा को ठीक तरह से समझने के लिए डाटा टाइप का use किया जाता है। डाटा टाइप से कंप्यूटर के compiler और interpreter को ये समझने में आसानी होती है की user उस डाटा को किस तरह से इस्तेमाल करना चाहता है।

डाटा type किसी भी डाटा का गुण या विशेषता है जो इस बात को निर्धारित करता है की उसमें आने वाले एलीमेंट कौन-कौन से हो सकते हैं तथा उन एलीमेंट्स पर आप कौन-कौन से ऑपरेशन लगा सकते हैं।

## Data Type

Computer memory में उस स्थान को कहते हैं । जहाँ पर अस्थायी रूप से मान को रखा जाता है । डेटा टाइप विजुअल बेसिक में As keyword से पहचाना जाता है ?

As keyword नहीं होने पर ,डेटा टाइप variant कहलाती है ।

विजुअल बेसिक में प्रयोग होने वाले या टाइप की सूची निम्नलिखित है

- Byte
- Integer
- Long
- Currency
- Single
- Double
- Boolean
- Date
- String
- Object
- Variant

**Byte** – यह integer को store करता है जिसका मान 0 से 255 के मध्य होता है ।

**Example :-** Dim num As Byte

**Integer :-** यह एक आंकिक परि उतनांक है , जिसका मान – 32768 +32767 ( 16 bits ) होता है

**Example –** Dim count As Integer

# Visual Basic /VB.NET

## PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

- **Short (Int16):** यह integer value को स्टोर करने लिए use होता है। इसमें -32768 से 32767 तक value store की जा सकती है और यह memory में 2 bytes लेती है।

Dim a As Short

a = 10000

- **Long (Int64):** यह integer value को स्टोर करने लिए use होता है। इसमें integer से अधिक value store की जा सकती है और यह memory में 8 bytes लेता है।

---

**Long :-** यह भी एक आंकिक परिवर्तनांक होता है। जिसका मान Integer data से अधिक होती है। मान - 2,147,483,648 से + 2,147,483,648 (32bits)

**Example :-** Dim bignum As Long

---

**Currency :-** यह एक आंकिक परिवर्तनांक होता है। जिसमें मौद्रिक मान आते हैं

**Example:-** Dim costprice As currency

---

**Single:-** यह एक ऐसा आंकिक परिवर्तनांक होता है। जिसमें दशमलव होता है

**Example:-** Dimgrad As Single

---

# Visual Basic /VB.NET

PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

**Double:-** यह भी एक दशमलव वाला आंकिक परिवर्तनांक होता है । जिसका मान सिंगल डेटा प्रकार से अधिक होता ।

**Example:-** Dim atom As Double

---

**Boolean :-** इस परिवर्तनांक में दो मान होते हैं True अथवा False

**Example :-** Dim YesNo As Boolean

---

**Date :-** इस डेटा टाइप में डेट मान स्टोर कराया जाता है

**Example :-** Dim Cdate As Date

---

**String :-** यह एक ऐसा डेटा टाइप है । जिसमा अक्षर अथवा अक्षर श्रृंखला हो सकता है , यह केवल text को store करने के लिए use किया जाता है। यह set of characters को स्टोर करता है। इसमे 2 GB तक text store कर सकते है।

- **Char :** यह single character को store करने के लिए use होता है। और memory मे 2 byte space लेता है।

**Example:-** Dim Strname As String

---

**Object:-** इस डेटा टाइप में विजुअल बेसिक और अन्य एप्लीकेशन में आब्जेक्ट का सन्दर्भ स्टोर कराया जा सकता है ।

**Example :-** Dim ob As Object

**Variant :-** यह एक सामान्य उद्देश्यीय डेटा टाइप है। जिनमें अन्य प्रकार के डेटा टाइप के मानों को स्टोर कराया जा सकता है।

**Example :-** Dim K

## Array क्या हैं ?

### Array क्या है?

**Array** एक ऐसे data structure है जिसके एक variable में एक ही समय में एक समान data type के एक से अधिक data items store किए जा सकते हैं। इसमें store सभी data items का एक ही नाम होता है लेकिन उनका index अलग-अलग होता है जिसकी सहायता से ही हम array के सभी data items को access करते हैं।

**उदाहरण:** यदि किसी collage में 1000 विद्यार्थी हो और हमें सभी विद्यार्थियों के रोल नंबर्स को store करने के लिए एक program लिखना हो तो यह दो तरीके से हो सकता है।

पहला तरीका तो यह है कि हम इनमे से प्रत्येक विद्यार्थी के लिए अलग-अलग variable बनाए और रोल नंबर्स को स्टोर करे लेकिन यह तरीका उतना प्रभावशाली नहीं होगा क्योंकि इसके लिए पहले तो हमें 1000 variables बनाने की जरूरत पड़ेगी जिसमें बहुत समय लगेगा। और यदि हम किसी तरह 1000 variable बना भी ले तो भी इसमें सभी विद्यार्थियों के रोल नंबर्स को input एवं output कराने के लिए हमें प्रत्येक विद्यार्थी के लिए अलग-अलग code लिखने की जरूरत पड़ेगी।

इस तरह कुल मिलाकर सामान्य variable से 1000 विद्यार्थियों को रोल नंबर्स को store करा पाना हमारे लिए असंभव होगा। अतः इस समस्या के समाधान के लिए हम एक array type का variable A[1000] बनाएंगे जिसमें एक ही variable में सभी विद्यार्थियों के रोल नंबरो को एक लूप का प्रयोग करके आसानी से input एवं output कराया जा सकता है।



## Array के प्रकार

Array निम्नलिखित 3 प्रकार के होते हैं:

- One Dimensional Array
- Two Dimensional Array
- Multi Dimensional Array

### **One Dimensional Array क्या है?**

इस प्रकार के array में row या column में data को store किया जाता है इसलिए इसे one dimensional array या linear array कहते हैं। इसमें single subscript (index) का प्रयोग किया जाता है इसलिए इसे single sub-scripted variable भी कहते हैं।

**उदाहरण:** माना A एक one dimensional array है जिसका subscript/index (i), word length (w) है तथा इसमें M elements store है। तब हम निम्नलिखित प्रकार से प्रदर्शित करेंगे  $-A[M]$

जहाँ,

- A - Name of array variable
- M - Size/length of array
- $A[j]$  -  $i^{\text{th}}$  element of array
- W - word length (Number of bytes occupied by each element)

### **Representation of One Dimensional Array in Memory:**

माना A एक one dimensional array है जिसमें 5 elements store है जिनका subscript/index (i) क्रमशः 1, 2, 3, 4, 5 व word length (w) 1 byte है तब इसे सामान्य रूप से  $A[5]$  से एवं memory में निम्नलिखित दो प्रकारों से row wise या column wise प्रदर्शित किया जा सकता है:



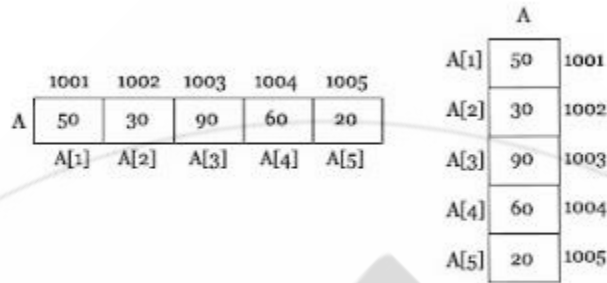


Fig: One Dimensional Array in Memory

जहाँ,

- A is name of array variable.
- 1,2,3,4,5 are index.
- 1001, 1002, 1003, 1004, 1005 are memory addresses of memory locations.
- 50, 30, 90, 60, 20 are elements.
- First index 1 is called lower bound (LB) of array
- Last index 5 is called the upper bound (UB) of array
- Address of first element A[1] that is 1001 is called base address (BA) of array

### Calculation of Total Elements in One Dimensional Array:

यदि हमें किसी one-dimensional array का lower bound (LB) तथा upper bound (UB) ज्ञात हैं तो इसमें elements की कुल संख्या निम्नलिखित सूत्र से ज्ञात किया जा सकता है:  $(UB-LB+1)$

ऊपर दिए गए array में lower bound 1 तथा upper bound 5 है। अतः  $\text{total elements} = (5-1+1) = 5$

### Two Dimensional Array क्या है?

इस प्रकार के array में row और column दोनों में data को store किया जाता है इसलिए इसे two dimensional array कहते हैं। इसमें double subscript (index) का प्रयोग किया जाता है इसलिए इसे double sub-scripted variable भी कहते हैं। Two dimensional array को गणित में matrix एवं software में table कहते हैं।

उदाहरण: माना A एक two dimensional array है जिसका subscript/index (i, j), word length (w) है तथा इसमें  $M \times N$  elements store है। तब हम इसे निम्नलिखित प्रकार से प्रदर्शित करेंगे-  $A[M, N]$

# Visual Basic /VB.NET

PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

जहाँ,

- A - Name of array variable
- M - Row's size/length of array
- M\*N - Size/length of array
- A[i,j] - [i,j]<sup>th</sup> element of array
- w - word length (Number of bytes occupied by each elements)

पुनः माना two dimensional array A में 3 row और 4 column है। Rows का index क्रमशः 1,2,3 तथा columns का index क्रमशः 1,2,3,4 है। तब इसे एक matrix में निम्नलिखित प्रकार से प्रदर्शित कर सकते हैं-

		Columns			
		1	2	3	4
Rows	1	A[1,1]	A[1,2]	A[1,3]	A[1,4]
	2	A[2,1]	A[2,2]	A[2,3]	A[2,4]
	3	A[3,1]	A[3,2]	A[3,3]	A[3,4]

Fig: Two Dimensional Array in Matrix Form

## Representation of Two Dimensional Array in memory:

Two dimensional array के elements को एक matrix के रूप में प्रदर्शित किया जा सकता है लेकिन इसके element भी memory में किसी one dimensional array के elements की तरह एक line या sequence में ही store होते हैं। इनके memory में store होने का तरीका programming language पर निर्भर करता है। programming language इसे memory में दो प्रकार से row wise अथवा column wise store किया जा सकता है। किसी two dimensional array को memory में row wise store किया जाता है तो इसे row-major-order एवं column wise store किया जाता है तो इसे column-major-order कहते हैं।

माना A एक two dimensional array है जिसमें 3×4 element store है जिनका पहला subscript/index (i) क्रमशः 1,2,3 व दूसरा subscript/index (j) क्रमशः 1,2,3,4 तथा word length (w) 1 byte है। तब इसे सामान्य रूप से A[3,4] से एवं memory में निम्नलिखित दो प्रकार से row-major-order या column-major-order में प्रदर्शित किया जा सकता है-

# Visual Basic /VB.NET

PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

	A	
1001	10	A[1,1]
1002	20	A[1,2]
1003	30	A[1,3]
1004	40	A[1,4]
1005	50	A[2,1]
1006	60	A[2,2]
1007	70	A[2,3]
1008	80	A[2,4]
1009	90	A[3,1]
1010	100	A[3,2]
1011	110	A[3,3]
1012	120	A[3,4]

Fig: Row Major Order

	A	
1001	10	A[1,1]
1002	20	A[2,1]
1003	30	A[3,1]
1004	40	A[1,2]
1005	50	A[2,2]
1006	60	A[3,2]
1007	70	A[1,3]
1008	80	A[2,3]
1009	90	A[3,3]
1010	100	A[1,4]
1011	110	A[2,4]
1012	120	A[3,4]

Fig: Column Major Order

जहाँ,

- A is name of array variable
- 1,2,3 are row's subscript/index (LB=1, UB=3)
- 1,2,3,4 are column's subscript/index (LB=1, UB=4)
- 1001.....1012 are memory addresses of memory location
- 10.....120 are element
- Address of first element A[1,1] that is 1001 is called base address (BA) of array

## Calculation of Total Element in Two Dimensional Array:

यदि हमें किसी two dimensional array के row एवं column के lower bound (LB) तथा upper bound (UB) ज्ञात है तो इसमें element की कुल संख्या निम्नलिखित सूत्र से ज्ञात किया जा सकता है-

$$(UBr-LBr+1)*(UBc-LBc+1)$$

ऊपर दिए गए array में row का lower bound 1 और upper bound 3 है। इसी प्रकार column का lower bound 1 और upper bound 4 है। अतः total elements =  $(3+1-1)*(4+1-1) = 12$

## **Multi Dimensional Array क्या हैं?**

इस प्रकार के array में दो से अधिक subscript का प्रयोग किया जाता है इसलिए इसे multi dimensional array कहते हैं।

**उदाहरण:** माना A एक multi dimensional array है जिसका subscript/index ( i, j, k ), word length (w) है तथा इसमें  $L \times M \times N$  elements store है। तब हम इसे निम्नलिखित प्रकार से प्रदर्शित करेंगे—  
 $A[L,M,N]$

## **Application of Arrays**

Array एक महत्वपूर्ण एवं उपयोगी data structure होता है इसका उपयोग निम्नलिखित कार्यों के लिए किया जाता है:

### **1. Storing Similar Data**

One dimensional array का उपयोग एक समान data items को बड़ी संख्या में store करने संबंधी समस्या को हल करने के लिए किया जाता है। उदाहरण के लिए collage के सभी विद्यार्थियों के रोल नंबर्स को एक array में store करके रख सकते हैं।

### **2. Handling Strings**

One dimensional array का उपयोग characters के group को store करने के लिए किया जाता जिसे string कहते हैं। इसकी सहायता से किसी व्यक्ति का पहचान, नाम, पता, आदि को store किया जाता है।

### **3. Creating Table**

Two dimensional array का उपयोग table को बनाने के लिए किया जाता है जो rows और columns से मिलकर बना होता है। इसकी सहायता से spreadsheet software बनाए जाते हैं।

### **4. Manipulating Matrix**

# Visual Basic /VB.NET

PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

Two dimensional array का उपयोग matrix का बनाने के लिए किया जाता है जो rows और columns से मिलकर बना होता है। इसकी सहायता से किन्हीं दो matrix का योग, गुणनफल और ट्रांसपोज़ ज्ञात किया जाता है।

## 5. Creating Stack

Stack एक ऐसे data structure है जिसमें जिस ओर से data को insert किया जाता है उसी ओर से ही उसे delete भी किया जाता है। इसका प्रयोग गणितीय equations एवं expressions को हल करने के लिए किया जाता है। Linear array की सहायता से stack तैयार कर सकते हैं।

## 6. Creating Queue

Queue एक ऐसा data structure है जिसमें एक ओर से data को insert एवं दूसरे ओर से उसे delete किया जाता है। इसका प्रयोग processor के लिए process scheduling में किया जाता है। Linear array की सहायता से queue तैयार कर सकते हैं।

## Array और Linked Lists में अंतर:

### *Array*

- (1) Array के elements के मध्य linear relation memory location की सहायता से दिया जाता है।
- (2) Array में element का index होता है।
- (3) Array के किसी भी element को direct access किया जा सकता है।
- (4) Array के element successive या consecutive memory location में store होता है।
- (5) Array का size पहले से ही fix होता है जिसे run time में परिवर्तित नहीं किया जा सकता है। इसमें static memory allocation होता है।
- (6) Array में insertion एवं deletion करने के लिए बहुत अधिक elements को move करना पड़ता है जिससे समय खर्च होता है।

### *Linked List*

- (1) Linked List के nodes के मध्य linear relation pointer की सहायता से दिया जाता है।
- (2) Linked List में nodes का index नहीं होता है।
- (3) Linked List के किसी भी nodes को direct access नहीं किया जा सकता है।
- (4) Linked List के nodes successive या consecutive memory location में store नहीं होता है।
- (5) Linked List का size पहले से ही fix नहीं होता है जिसे run time में आवश्यकता के अनुशार परिवर्तित किया जा सकता है। इसमें dynamic memory allocation होता है।
- (6) Linked List में insertion एवं deletion करने के लिए बहुत अधिक node को move करना नहीं पड़ता है जिससे समय की बचत होती है।

# Visual Basic /VB.NET

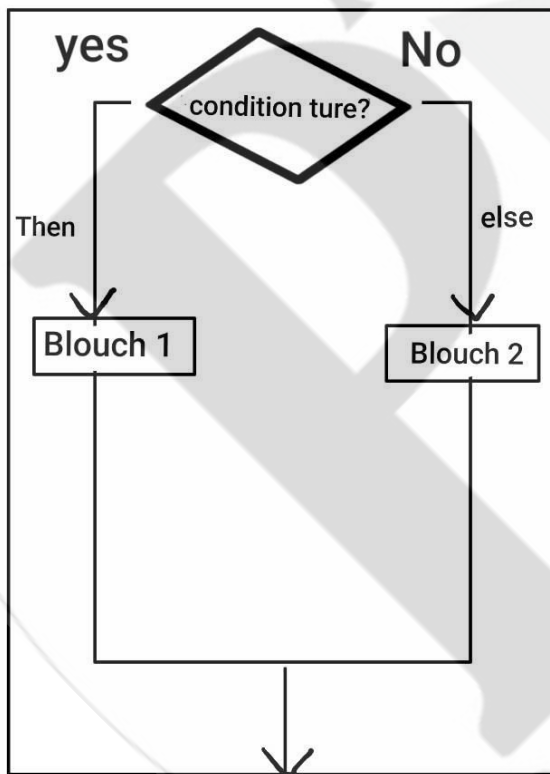
PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

(7) Array में elements को store करने के लिए (7) Linked List में प्रत्येक nodes को store करने के लिए कोई कोई अतिरिक्त memory आवश्यकता नहीं होती है। अतिरिक्त memory आवश्यकता होती है।

## Conditional Statement

### Conditional Statement-

बेसिक में Conditional Statement का उपयोग आपके program में कुछ शर्तों के आधार पर (निर्णय) decision लेने के लिए किया जा सकता है।



Conditional Statement In Visual Basic In



## 1) If Then Statement

### ***Example :-***

Module Condition

Sub Main()

Dim accountBalance As Integer = 0

If (accountBalance < 1000) Then

Console.WriteLine("Close Account!")

Console.ReadLine()

End If

End Sub

End Module

### **Output**

#### ***Close Account!***

उपरोक्त उदाहरण में, हम if स्टेटमेंट में Account Balance चेक कर रहे हैं।

यदि खाता शेष 1000 से कम आता है, तो हम खाते को बंद करने के लिए एक लाइन प्रिंट कर रहे हैं।



## 2) If then Else Statement

### Example :-

Module Condition

Sub Main()

Dim accountBalance As Integer = 1001

If (accountBalance < 1000) Then

Console.WriteLine("Close Account!")

Else

Console.WriteLine("We love having you with us!")

End If

Console.ReadLine()

End Sub

End Module

### Output

***We love having you with us!***

उपरोक्त उदाहरण में, यदि खाता शेष राशि > = 1000 है, तो हम एक संदेश print कर रहे हैं।

## 3) Nested If Then

हम कई statement दे सकते हैं। इसका उपयोग करते समय सावधान रहें क्योंकि यह पढ़ने के लिए एक गड़बड़ बन सकता है। आप कुछ परिदृश्यों में इससे बचने के लिए && या || ऑपरेटर का उपयोग कर सकते हैं।

### Example

Module Condition

Sub Main()

Dim accountBalance As Integer = 501

If (accountBalance < 1000) Then

If (accountBalance < 500) Then

Console.WriteLine("Close Account!")

Else

Console.WriteLine("Maintain a minimum balance! You got 5 days time.")

End If

ElseIf (accountBalance > 1000000) Then

Console.WriteLine("Please find a Europe tour cruise package in your mailbox!")

Else

Console.WriteLine("We love having you with us!")

# Visual Basic /VB.NET

PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

```
End If      Console.ReadLine()
```

```
End Sub
```

```
End Module
```

## Output

**Maintain a minimum balance ! You got 5 days time.**

उपरोक्त उदाहरण में, हमने एक और if-else जोड़ा हैं।

उच्च मूल्यवान ग्राहकों के लिए, एक बड़ी सीमा से अधिक पैसे है, हमने उनके लिए एक अलग संदेश print किया हैं।

## 4) Switch Statement –

एक switch statement expression को लेता हैं और उसे evaluate हैं evaluation के प्रत्येक result के लिए, यह statement का उपयोग करता हैं। इसे ही हम switch statement कहते हैं। ब्रेक स्टेटमेंट प्रत्येक case के अंत में मौजूद होना चाहिए। यदि कोई case match नहीं होता है तो डिफॉल्ट केस implement किया जाता है।

### Example:-

```
Module Condition
```

```
Sub Main()
```

```
Dim num As Integer = 0
```

```
Select Case num
```

```
Case 1
```

```
Console.WriteLine("One")
```

```
Case 2
```

# Visual Basic /VB.NET

PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

```
Console.WriteLine("Two")
```

Case 3

```
Console.WriteLine("Three")
```

Case 4

```
Console.WriteLine("Four")
```

Case Else

```
Console.WriteLine("Other Number!")
```

```
End Select
```

```
End Sub
```

```
End Module
```

## **Output:-**

### **Other Number!**

उपरोक्त उदाहरण में, हमने चार alphabet पर एक स्विच स्टेटमेंट जोड़ा है। हम चार के मूल्य के आधार पर एक sample alphabet print कर रहे हैं |

## **Control statements(structures) –**

प्रोग्रामिंग भाषा में, प्रोग्राम के execution के flow को नियंत्रित करने के लिए जो स्टेटमेंट्स या structures प्रयोग किये जाते हैं उन्हें control statements(structures) कहते हैं.

control statements दो प्रकार के होते हैं:-

**1:-conditional statements**

**2:-loop statements**

## 1:-conditional statements:-

ये निम्नलिखित प्रकार के होते हैं.

### (a):-if statement:-

if statement एक control statement है जिसका प्रयोग एक विशेष कंडीशन को test करने के लिए किया जाता है. इसमें कंडीशन केवल एक बार execute होती है जब कंडीशन सत्य होती है.

#### **syntax-**

```
if(condition)
{
statements
}
```

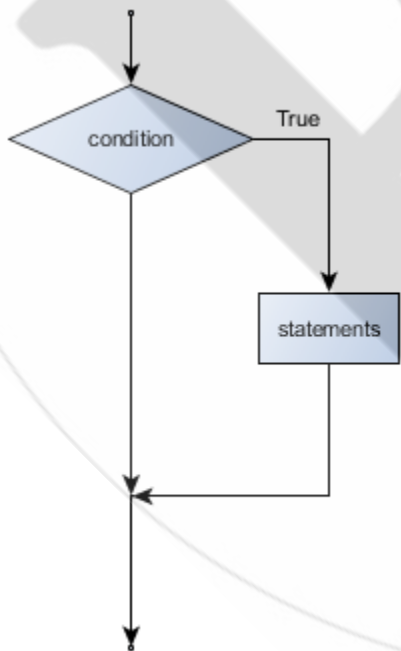


Fig:+if फ्लो चार्ट

### (b):-if-else statements:-

# Visual Basic /VB.NET

PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

किसी विशेष कंडीशन को test करने के लिए if-else statement का प्रयोग किया जाता है. यदि कंडीशन सत्य है तो if statement को execute किया जाता है अगर कंडीशन असत्य है तो else statement को execute किया जाता है.

## **syntax-**

```
if(condition)
{
    statement
}
else
{
    statement
}
```

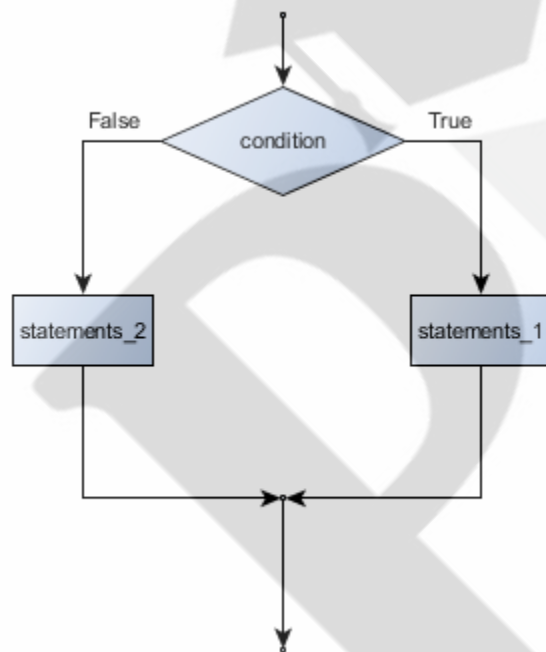


fig:-if else flow chart

## **(c):- switch statements:-**

switch statement एक multi-way branch स्टेटमेंट है.

यह statement भी एक selection स्टेटमेंट जो कि एक प्रोग्राम के execution के लिए विभिन्न paths को डिफाइन करता है.

यह if-else स्टेटमेंट के लिए विकल्प के तौर पर कार्य करता है.

## **syntax:-**

```
switch(variable)
{
```

# Visual Basic /VB.NET

PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

```
case constant 1;  
statements(s);  
break;  
case constant 2;  
statement(s);  
break;  
case constant 3;  
statement(s);  
break;  
_____  
default  
statement(s);  
}
```

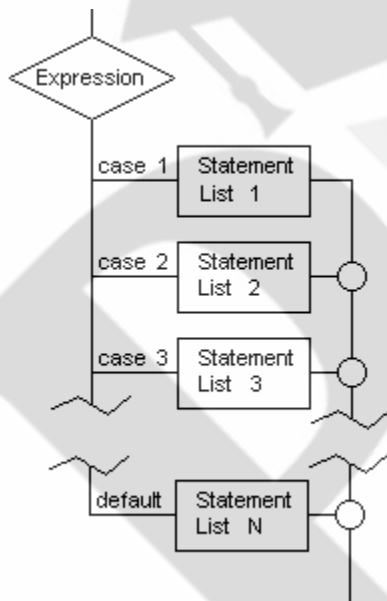


Fig:-switch statement flowchart

## (2):-Loop statements (लूप स्टेटमेंट्स):-

loops के द्वारा हम किसी एक स्टेटमेंट या बहुत सारें स्टेटमेंट्स को एक से ज्यादा बार execute कर सकते हैं जब तक कि condition को प्राप्त नहीं कर लिया जाता.

loops स्टेटमेंट निम्नलिखित प्रकार के होते हैं:-

- 1:-while loop
- 2:-do-while loop
- 3:-for loop



## 1:-while loop:-

while loop एक entry-controlled लूप है. इसमें स्टेटमेंट तब तक लगातार execute होते रहता है जब तक कि एक condition सत्य(true) नहीं हो जाती है. यह सबसे पहले condition की जांच करती है और बाद में स्टेटमेंट को execute करती है.

### **syntax:-**

```
while(condition)
{
    statement1;
    statement2;
    _____
}
```

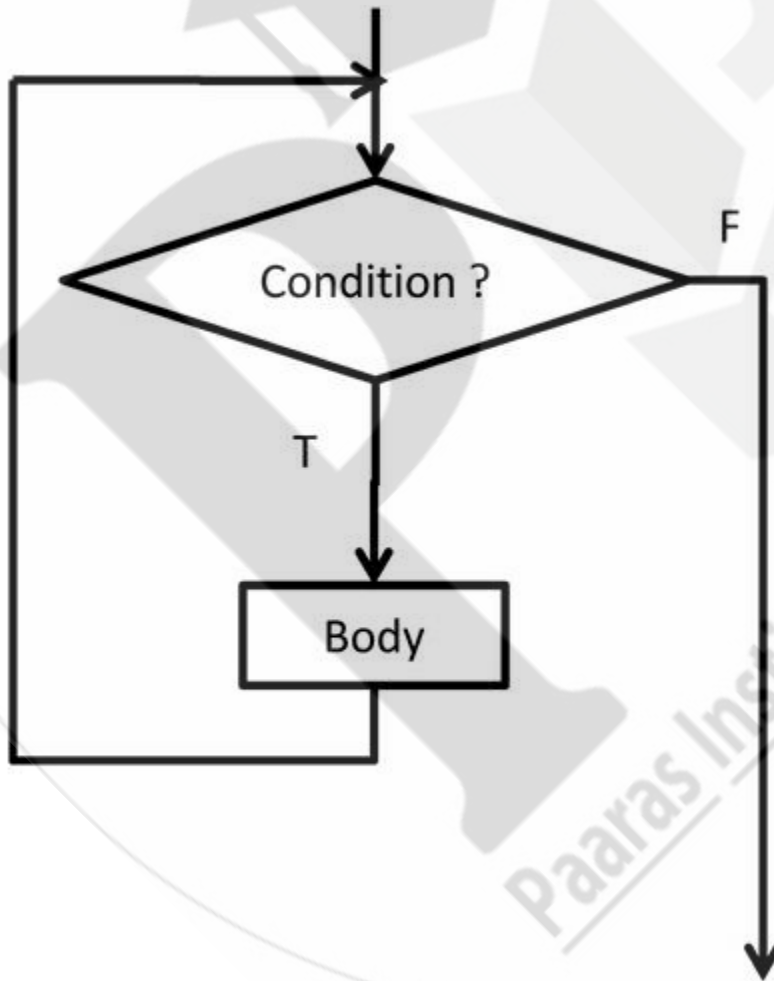


Fig:-while loop

## 2:-do-while loop:-

यह loop एक exit-controlled लूप है. यह लूप भी while लूप की तरह ही है परन्तु यह पहले स्टेटमेंट को execute करता है तथा बाद में condition की जांच करता है.

यह loop यह सुनिश्चित करता है कि प्रोग्राम को कम से कम एक बार execute किया गया है.

**syntax:-**

```
do
{
Statement1;
Statement2;
}
_____
}
```

# Visual Basic /VB.NET

PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

While (condition);

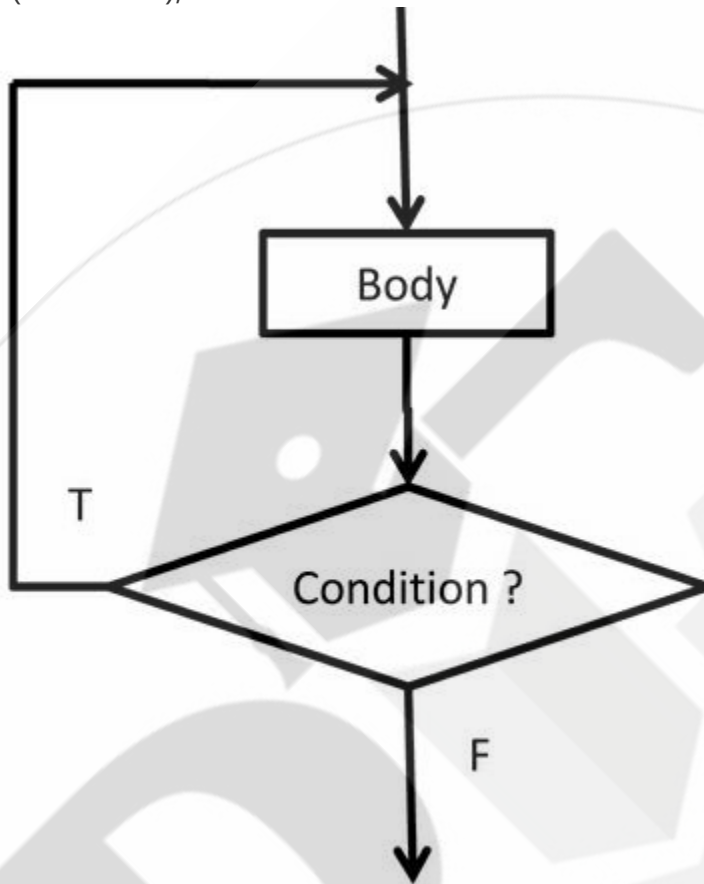


Fig:-while loop flowchart

## 3:- For loop:-

यह एक entry controlled लूप है. इसमें स्टेटमेंट तब तक execute होता है जब तक कि condition सत्य नहीं हो जाती है. यह loop तीन घटकों initialization स्टेटमेंट, boolean स्टेटमेंट तथा increment/decrement स्टेटमेंट से मिलकर बना होता है.

### **syntax:-**

```
for(initial condition, test condition; incrementor or decrementor)
{
statement1;
statement2;
}
```

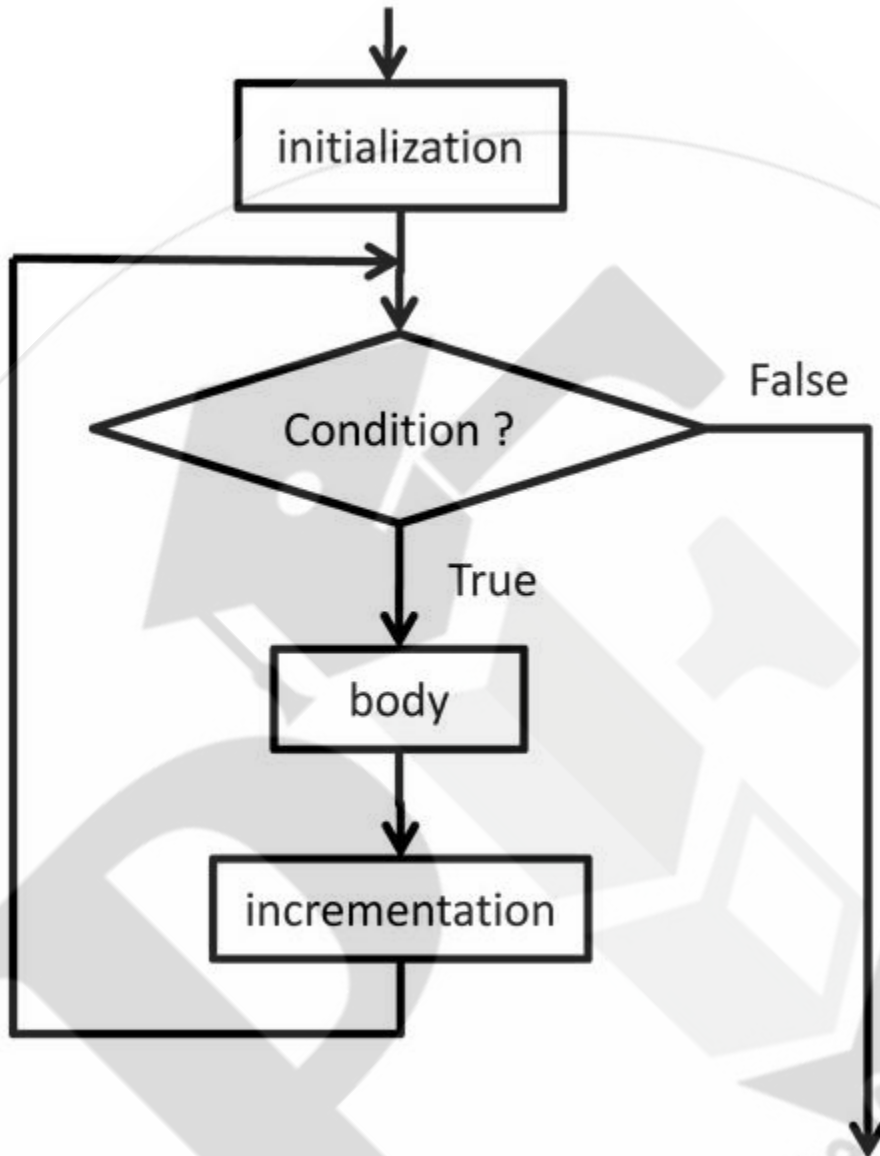


Fig:-for loop flowchart

**अन्य स्टेटमेंट्स:-** इनके अलावा कुछ और भी स्टेटमेंट्स है जो निम्न है:-

**1:-** Break statement

**2:-** Continue statement

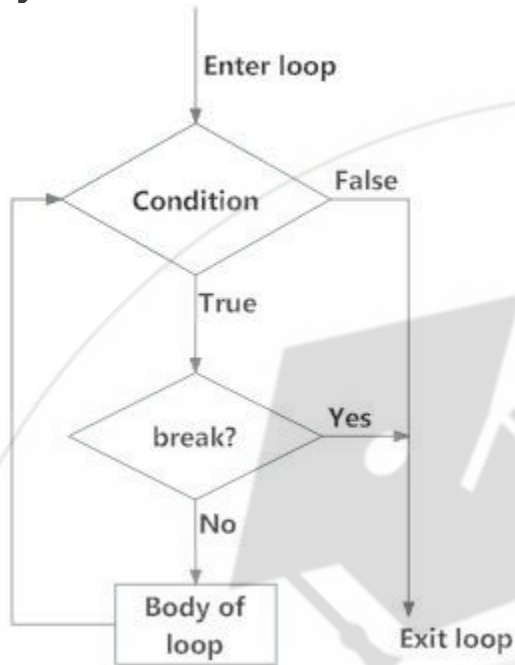
**1:-Break statement:-**

इस स्टेटमेंट का प्रयोग switch स्टेटमेंट में स्टेटमेंट के अनुक्रम(sequence) को समाप्त करने के लिए किया जाता है तथा किसी loop को तुरंत exit करने के लिए किया जाता है.

# Visual Basic /VB.NET

PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

**syntax:-** break;



## 2:-continue statement:-

continue स्टेटमेंट का प्रयोग तब किया जाता है जब हम loop को लगातार अगले iteration के साथ run करना चाहते हैं तथा वर्तमान iteration के लिए लूप के अन्य स्टेटमेंट्स को skip करना चाहते हैं.

**syntax:-** continue;

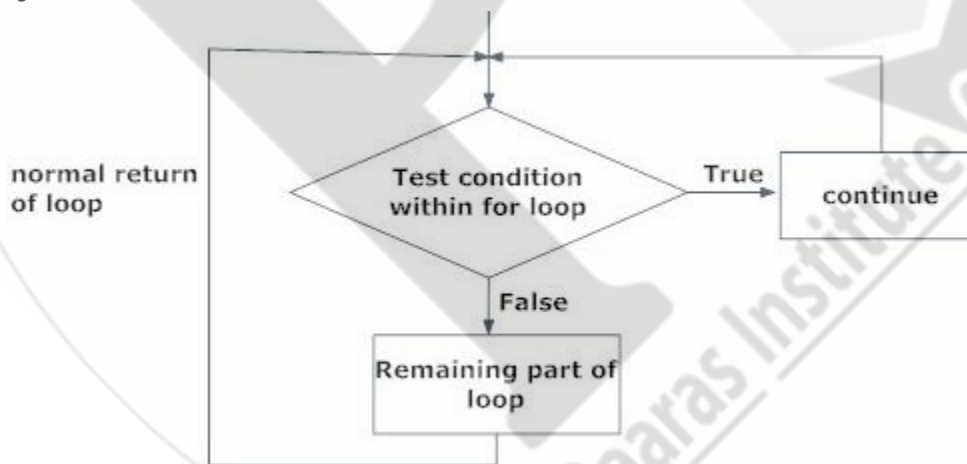


Fig:-continue flowchart

## MsgBox & Input Box-

### MsgBox -

**मैसेज बॉक्स ( Message box ) :-** मैसेज बॉक्स का प्रयोग सदैव प्रयोगकर्ता को सूचना प्रदान करने तथा उसे सूचना संदेश देने के लिए किया जाता है। इस डायलॉग बॉक्स का प्रयोग निम्नलिखित सूत्रानुसार ( Syntax ) किया जाता है।

*MsgBox ( Prompt [ , icons + buttons ] [ , title ] )*

*memory \_ variable = MsgBox ( prompt [ , icons buttons ] [ , title ] )*

**Prompt :** यह उस टेक्स्ट को दर्शाता है जिसे मैसेज बॉक्स में दिखाना है।

**Icons + Buttons :** यह सुनिश्चित करता है कि मैसेज बॉक्स पर किस प्रकार की पिक्चर अर्थात् आइकॉन प्रदर्शित हो तथा उस पर कौन – कौन से बटन लगे हों।

कुछ आइकॉन ( Icon ) के प्रकार निम्न हैं –

Constant	Value	Determination
vbOKOnly	0	मैसेज बॉक्स में केवल OK बटन प्रदर्शन होता है।
vbOKCancel	1	मैसेज बॉक्स में OK तथा Cancel दो बटन का प्रदर्शन होता है।
vbAbortRetryIgnore	2	मैसेज बॉक्स में Abort , Retry , Ignore तीन बटन का प्रदर्शन होता है।
vbYesNOCancel	3	मैसेज बॉक्स में केवल Yes , no तथा Cancel बटन प्रदर्शन होता है।

# Visual Basic /VB.NET

## PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

vbYesNo	4	मैसेज बॉक्स में केवल Yes तथा No : दो बॉक्स को प्रदर्श होता है , इत्यादि ।
Question	32	Question icon के लिए
Exclamation	48	Exclamation icon के लिए
Information	64	Information icon के लिए
DefaultButton1	0	First button को default करने के लिए
DefaultButton2	256	Second button को default करने के लिए
DefaultButton3	512	Third button को default करने के लिए
ApplicationModal	0	इस mode में user को application में work करने के लिएmsgbox में click करना आवश्यक होता है।
SystemModal	4096	इसमें user के click करने तक सभी application suspendहो जाते हैं।
MsgBoxSetForeground	65536	Msgbox को foreground window बनाने के लिए
MsgBoxRight	524288	Text को right align करने के लिए
MsgBoxRtlReading	1048576	इसमें text Arabic system के अनुसार right to left हो जाते हैं।

### Icons + Buttons



# Visual Basic /VB.NET

PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

**Note :** विजुअल Basic में प्रत्येक बटन के लिए पहले से मान निर्धारित किया गया है जिसकी सूची निम्न हैं –

Constants	Value
OK	1
Cancel	2
Abort	3
Retry	4
Ignore	5
Yes	6
No	7

## MsgBox & Input Box

**Following is an example illustrates the use of message boxes**

\*Open a new Project and save the Form as messageboxdemo.frm and save the Project as messageboxdemo.vbp

\*Design the application as shown below

# Visual Basic /VB.NET

## PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

Object	Property	Settings
Form	Cation	MessageBoxDemo
	Name	frmMessageBoxDemo
Label	Caption	LblName
	Name	Name
TextBox	Name	TxtName
	Text	(Empty)
List Box	Name	LastName
Command Button	Caption	Add
	Name	cmdAdd

**Example illustrates the use of message boxes**

**Example:-**

```
Private sub command1_click()  
MsgBox "Hello World , How Are You", vbExclamation,"information"  
End Sub
```

## Input Dialog Box

**इनपुट डायलॉग बॉक्स ( Input dialog box ) :-** इनपुट box function user द्वारा इन्टर किया गया । String को वेरियबल में Return करता है । इनपुट बॉक्स में केवल दो बटन होते हैं । Ok और Cancel यदि यूजर के द्वारा OK बटन को क्लिक किया जाता है तो इन्टर किया हुआ मान वेरियबल को Return करता है ।

अगर यूजर द्वारा Cancel बटन को क्लिक करता है तो Zero स्ट्रिंग Return होती है ।

**Syntax ( सूत्र ) :-** `InputBox ( Message [ , Title ] [ , default ] [ , xpos ] [ . ypos ] [ , Helpfile ] [ . context ]`

**Messages** – टेक्स्ट मेसेज को प्रदर्शित करता है । मैसेज की अधिकतम लंबाई लगभग 1024 character हो सकती है । इनपुट बॉक्स में एक ज्यादा लाइन लिखने के लिये कैरिज Return ( 13 ) और chr ( 10 ) का उपयोग करते हैं ।

**Title :** Title बार में Title दिखाने के लिये title का उपयोग किया जाता है

**Default :** यह एक Optional स्ट्रीम Expression है , जिसका उपयोग TextBox में response को दिखाने के लिये किया जाता है ।

**xpos :** यह एक नम्बर है जिसके द्वारा left edge से horizontal distance को दिखाया जाता है ।

**ypos :** यह अपर edge का vertical distance है ।

### **Example:-**

```
Private Sub Command1_Click()  
Dim n1, n2 As Integer  
n1 = InputBox("Enter First Integer", "Input")  
n2 = InputBox("Enter Second Integer", "Input")  
n3 = Val(n1) + Val(n2)  
MsgBox "Addition of two numbers : " & n3  
End Sub
```

# Visual Basic /VB.NET

PAARAS INSTITUTE OF EDUCATION (KASHYAP SIR)

---

