GRAPHICS IN QBASIC

REVIEW OF QBASIC

QBasic is a programming language developed by Microsoft in 1985. It is the successor of earlier forms of BASIC (Beginners All-Purpose Symbolic Instructions Code). QBasic is an ideal programming language for beginners because of its simple commands, improved programming structures, better graphics and flexibility. It is an integrated development environment (IDE) to write, edit, debug and execute basic programs.

CONSTANTS AND VARIABLES

CONSTANTS

Constants are the values that do not change during the execution of a program. Constants are of two types:

Numeric Constants: They are further divided into two types: **integer** and **real** constants. Integer constants are positive or negative numbers, without any fractional part. E.g., 12, 234, -786 etc. Real constants are the numbers with a fractional part. E.g., 75.5, 46.25 etc.

Character Constants: They are also of two types: **Single Character Constant** and **String (Multi-character) Constant**. A 'single character constant' contains only a single character enclosed in double quotes. E.g., "D", "F" etc. A 'string constant' is a sequence of characters enclosed within double quotes. E.g., "Kips", "Welcome to Basic Programming" etc.

VARIABLE

A **variable** is a meaningful name of data storage location in computer memory. It holds the value and continues until another value is assigned to it. A variable has a type, which is defined by the kind of value it holds.

Numeric Variable: It can hold only numbers that can be of integer, floating decimal, or long integer type. It is represented by an alphabet. A numeric variable should not contain any space or symbols like ^, ?, \, /, @, :, ; etc.

Example: **A = 5** means the value 5 is assigned to the variable **A**.

String Variable: A string variable contains values, symbols or text within double quotes (" "). It is represented by an alphabet followed by the dollar sign. E.g., A$ = "KIPS".

Example

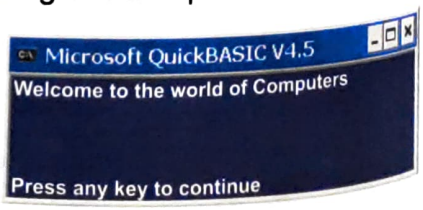
Another way to think of a variable is to imagine a small bucket with a name on it. Put "Name$" on it. This is the bucket's (variable's) name. Now take a piece of paper and write your name on it and drop it into the imaginary bucket. Now the variable 'Name$' has your name in it. Same way, computer variables can only hold one piece of information (one value) at a time.

GETTING STARTED

Double-click on the QBasic icon on the desktop to start QBasic. The QBasic window will appear with the Welcome dialog box. Press Esc key to hide the Welcome dialog box. Type the following program and press **F5** to run the program.

Example

```
CLS
PRINT "Welcome to the world of Computers"
END
```



QBASIC STATEMENTS

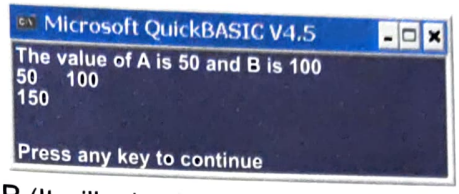
CLS stands for "Clear Screen" and is used to clear the screen.

PRINT statement is used to display any message or the output of a program. The message has to be enclosed within quotes but the constants, variables and expressions can be expressed as such. E.g.:

Syntax: [Line No] PRINT <Constant> or <Variable> or <Expression> or <"Message">or <list separated by commas or semi-colons>

Example

```
CLS
A = 50
B = 100
PRINT "The value of A is "; A; " and B is "; B
PRINT A, B
PRINT A + B
END
```



(It will print the values just after the statement.)

(It will print the values after leaving some spaces.)

(It will print the sum of the values.)

Let's Know More

The data, variable or expression should be of the same type on either side of the statement. If variable is numeric, the data assigned to it should also be numeric and so is the case with string variables. E.g.: A = 42, A$ = "God is Great"

Fact File

John G. Kemeny(R) and **Thomas E. Kurtz(L)** designed Basic language in 1964.

INPUT command is used to enter the values (text or number) while the program is being executed. This command waits for the user to enter the information and then assigns the values accordingly.

Syntax: [LINE NO] INPUT <Numeric or String Variable Name>

INPUT A

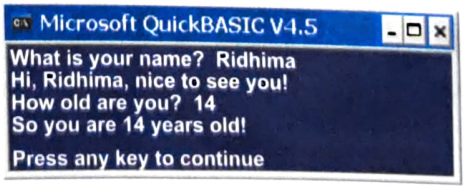
We can also have a list of variables in an INPUT statement but they should be separated by commas.

Syntax: [LINE NO] INPUT <Variable1>, <Variable2>, <Variable3>

Example

Type the following program and press F5 to execute it.

```
PRINT "What is your name?";
INPUT name$
PRINT "Hi, "; name$; ", nice to see you!"
PRINT "How old are you?";
INPUT age
PRINT "So you are "; age; " years old!"
END
```



The **END** command tells QBasic that the program ends here.

We can also use INPUT command to ask the user to enter the information.
E.g., **INPUT "What is your name"; name$**. It will also give the same result.

GOTO

Sometimes you want the program to jump to a particular line. GOTO statement is used to transfer the control from one statement to another.

Syntax: [Line No.] GOTO <Line No.>

Let's Discuss

Constant vs Variable

IF ... THEN STATEMENT

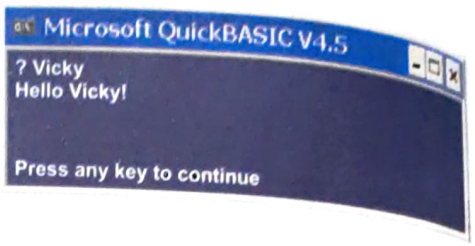
It is used for making decisions based on the result of comparisons. The IF command has to be followed by THEN command and lastly END IF command.

Syntax: IF <CONDITION> THEN <STATEMENT>

If the condition is TRUE, then the instruction(s) specified after "THEN" is executed. If the condition is FALSE, the control shifts to the ENDIF, ignoring everything after the THEN statement on that line.

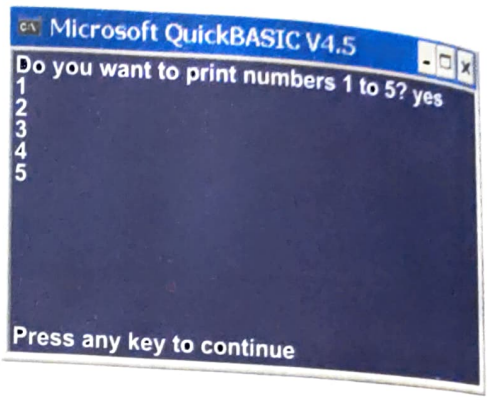

Example

```
INPUT NAME$
IF NAME$ = "Vicky" THEN
PRINT "Hello Vicky!"
END IF
```




Example

```
INPUT "Do you want to print numbers 1 to 5" ; Y$
IF Y$ = "yes" THEN
PRINT "1"
PRINT "2"
PRINT "3"
PRINT "4"
PRINT "5"
END IF
```



Here, the program asks you a question. If you enter yes, then it prints 1 to 5 on the screen.

IF... THEN ...ELSE

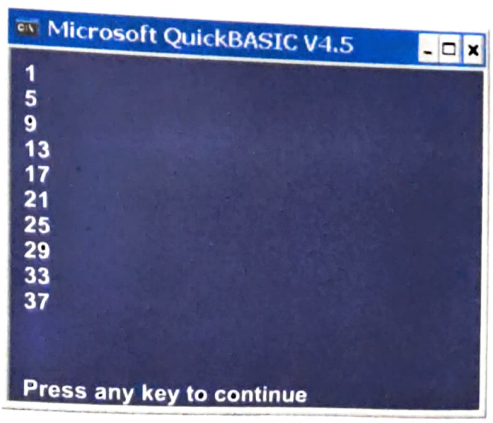
It is a conditional decision making statement. If the condition given after IF is true, statement(s) specified after THEN is executed. But if the condition is False, the statement(s) specified after ELSE will be executed.

Syntax: IF <condition> THEN <statement1> ELSE <statement2>


Example

To print series with a gap of 4 between 1 and 40

```
10  N = 1
20  IF N < 41 THEN GOTO 30 ELSE GOTO 60
30  PRINT N
40  N = N + 4
50  GOTO 20
60  END
```



USING GRAPHICS

In QBasic, it's very easy and interesting to create and display graphics on the screen. We use pixels (picture elements) to draw lines, figures and patterns of different shapes. Before we proceed with making beautiful graphics, let us first understand the display characteristics of our monitor. We can have two common modes of display:

MEDIUM RESOLUTION

The computer screen is divided into 320(horizontal) by 200 (vertical) pixels.

HIGH RESOLUTION

The computer screen is divided into 640(horizontal) by 200 (vertical) pixels.

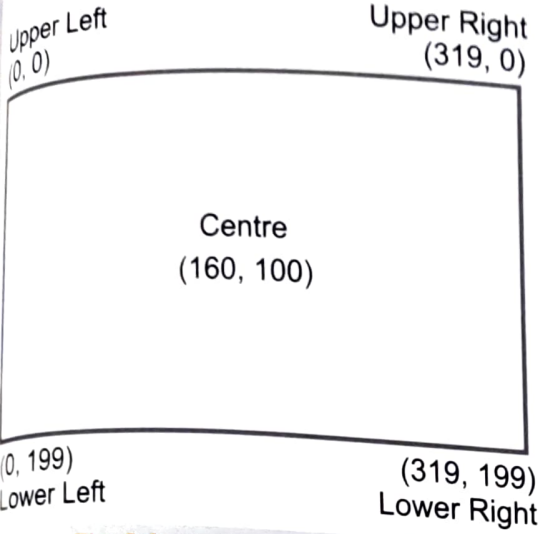


Fig. 8.1: Medium Resolution

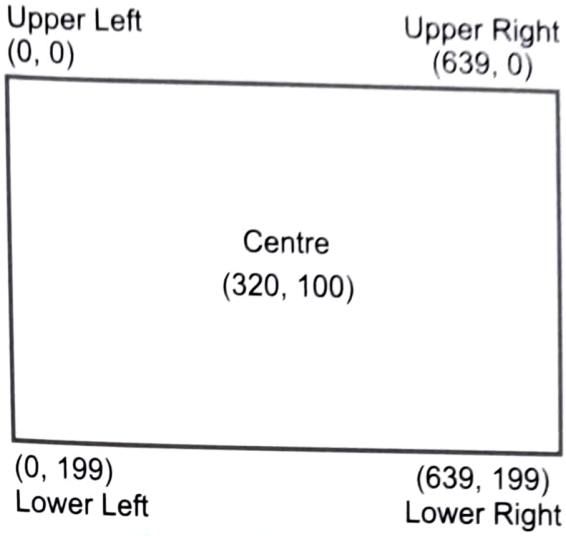


Fig. 8.2: High Resolution

Let's Know More

The amount of pixels per unit area on the screen is called its resolution.

SCREEN STATEMENT

This statement is used to set the screen resolution. Its syntax is as follows:

[Line No] SCREEN mode parameter

There are many screen modes that can be used in QBasic. Every mode has a different resolution and supports different number of colours. The following table displays the various screen modes with their resolution and the number of colours they support.

Screen Mode	Resolution	No. of Colours Supported
Screen 1	320 X 200	4
Screen 2	640 X 200	2
Screen 7	320 X 200	16
Screen 8	640 X 200	16
Screen 9	640 X 350	16
Screen 10	640 X 350	2
Screen 11	640 X 480	2
Screen 12	640 X 480	16
Screen 13	320 X 200	256

NOTE

Screen 0: This mode can be used only for text. No graphics can be created in this mode. Except mode 0, all the modes can be used to create graphics. The default mode of the screen is mode 0.

Let's Know More

Locate statement moves the cursor to the specified position on the screen.

Locate [row], [column]

Locate 5, 5

Know the Fact

To set a point from where a particular drawing should start on the screen, PSET statement is used **PSET (X , Y) , C**.

X refers to column and **Y** refers to row co-ordinate and **C** refers to colour code.

Let's Know More

We can use $, ! and # signs in a variable name.

COLOR STATEMENT

This statement is used to set attractive screen colours for displaying the text and the graphics. Its syntax is as follows:

[Line No] COLOR parameter f

[Line No] COLOR parameter f, parameter b

Where parameter **f** specifies the **foreground** colour and parameter **b** specifies the **background** colour. There are 16 colour options for background and foreground colours, specified by the values 0 to 15.

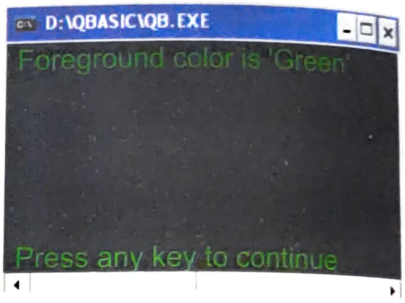
BACKGROUND COLOUR CHART

0. Black	4. Red	8. Dark Gray	12. Light Red
1. Blue	5. Magenta	9. Light Blue	13. Light Magenta
2. Green	6. Brown	10. Light Green	14. Light Yellow
3. Cyan	7. Gray	11. Light Cyan	15. White

Let us create programs to demonstrate the use of Color statement.

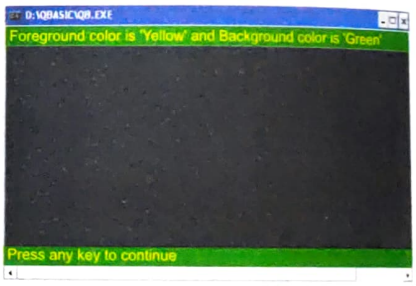
Example

```
CLS
COLOR 2
PRINT "Foreground color is 'Green'"
```



Example

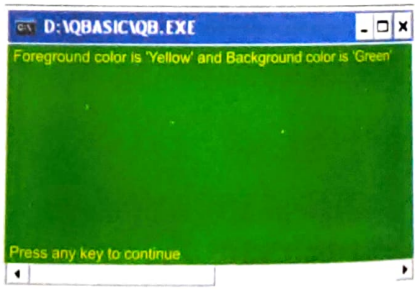
```
CLS
COLOR 14, 2
PRINT "Foreground color is 'Yellow' and Background color is 'Green'"
```



To change the background colour of the whole screen, write the CLS statement after Color statement.

Example

```
Color 14, 2
CLS
Print "Foreground color is 'Yellow' and Background color is 'Green'"
```



LINE STATEMENT

The LINE statement is used to draw a straight line between two points on the screen.

For better output and proper colours, we should choose the screen mode that supports all the 16 colours given in t chart.

yntax: LINE (X1, Y1) - (X2, Y2), COLOR

/here (X1, Y1) and (X2, Y2) represent the start and end coordinates of a line drawn in e specified colour.

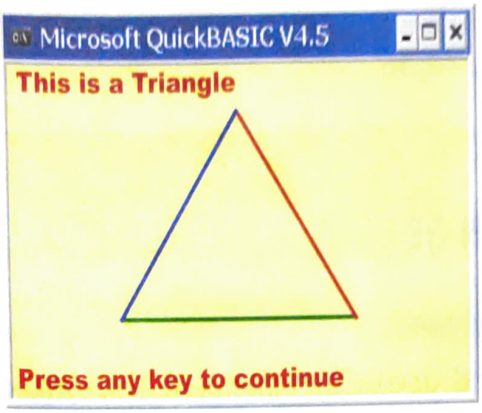
LINE (16, 20) - (44, 20), 1

The program will draw a blue line from the points 16, 20 all the way to the points 44, 20

Let's draw an isosceles triangle with horizontal base.

ample

```
SCREEN 7
COLOR 4,14
CLS
PRINT " This is a Triangle"
LINE (160, 30) - (250, 180), 4
LINE (250, 180) - (70,180), 2
LINE (70, 180) - (160, 30), 1
END
```



Know the Fact

To draw vertical line, x-axis co-ordinates will remain the same for both starting and ending points of the line. Similarly, for drawing a horizontal line, only the x-axis co-ordinates will change and y-axis co-ordinates will remain the same. To draw diagonal line on the screen, both 'x' and 'y' co-ordinates will change.

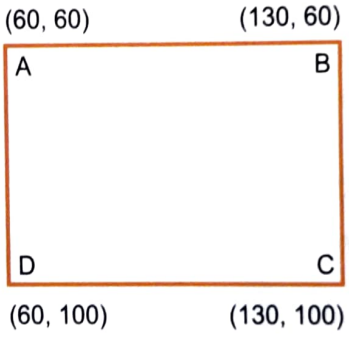
DRAWING BOXES

With Line statement, we can also draw boxes.

ample

```
SCREEN 7
COLOR 5,15
CLS
LINE (60,60)-(130,100),6,B
```

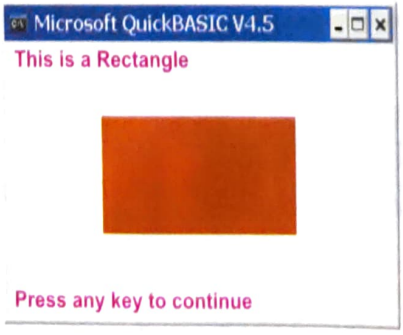
The letter B indicates the box option. In this statement, the coordinates (60, 60) and (130, 100) are the opposite corners of the rectangle.



To fill the box with the desired colour shade, add BF (box fill) option.

ample

```
SCREEN 7
COLOR 5,15
CLS
PRINT "This is a Rectangle"
LINE (60,60)-(130,100), 6, BF
```

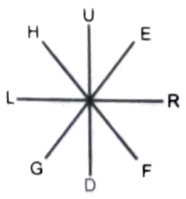


Let's Know More

With **DRAW** command, you can start a drawing from a fixed point which is set with PSET command and move around on the screen and draw straight lines in eight directions without specifying any pixel co-ordinates.

Syntax: DRAW "Direction string"

Direction string




```
CLS
PSET (80,80)
DRAW "R120 H60 G60"
```

R120 will draw a 120 pixels straight line in the right direction. H60 and G60 will draw lines of length 60 pixels in the left upward and the left downward directions respectively.

CIRCLE STATEMENT

e Circle statement is used to draw a circle, ellipse or an arc of a circle.

ntax: CIRCLE (x, y), r, p

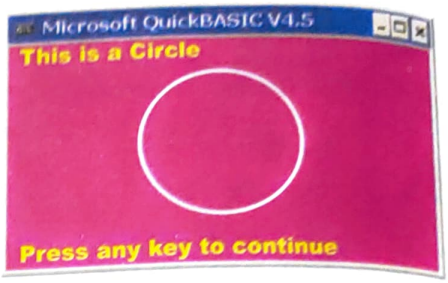
e coordinates x and y specify the centre location of a circle on the screen, 'r' gives the lue of the radius and 'p' specifies the colour code.

NOTE

PAINT statement fills an object with a colour.
Syntax: PAINT (x,y), color, bordercolor


Example

```
SCREEN 7
COLOR 14, 5
CLS
PRINT "This is a Circle"
CIRCLE (160, 100), 70, 15
END
```



CREATING SOUND IN QBASIC

BEEP STATEMENT

If you want to add the beep sound, use BEEP statement. The syntax of using the BEEP statement is given below:


Example

```
CLS
INPUT "Press Enter to hear a beep", A$
BEEP
```

PLAY

It is used to generate musical notes. The codes used to produce different musical notes are as follows:


Example

```
PLAY "e8 d8 c8 d8 e8 e8 e4"
PLAY "c8 d8 e8 f8 g8 a8 b8 c4"
END
```

- QBasic is a programming language developed by Microsoft in 1985.
- Constants are the values that do not change during the execution of a program. Constants are of two types: Numeric and Character.
- A variable is a meaningful name of data storage location in computer memory. It holds the value and continues until another value is assigned to it.
- GOTO statement is used to transfer the control from one statement to another.
- IF...THEN...ELSE is a conditional decision making statement.
- In QBasic, every mode has a different resolution and supports different number of colours.
- The Circle statement is used to draw a circle, ellipse or an arc of a circle.