

# Procedure! - الإجراء lecture

Procedures are subprograms that instead of returning a single value allow to obtain a group of results.

→ Procedure declaration! - الإعلـان

Procedure name (local variable)  
begin  
<Procedure body>  
end;

→ where the local variables defined in the body of procedure only.

→ Procedure body contain a collection of statements that define what the procedure does. it should always be enclosed between the reserved word (begin ~ end;).

# variables types

→ local variables :-

متغيرات محلية وتعرف داخل الاجراء فقط

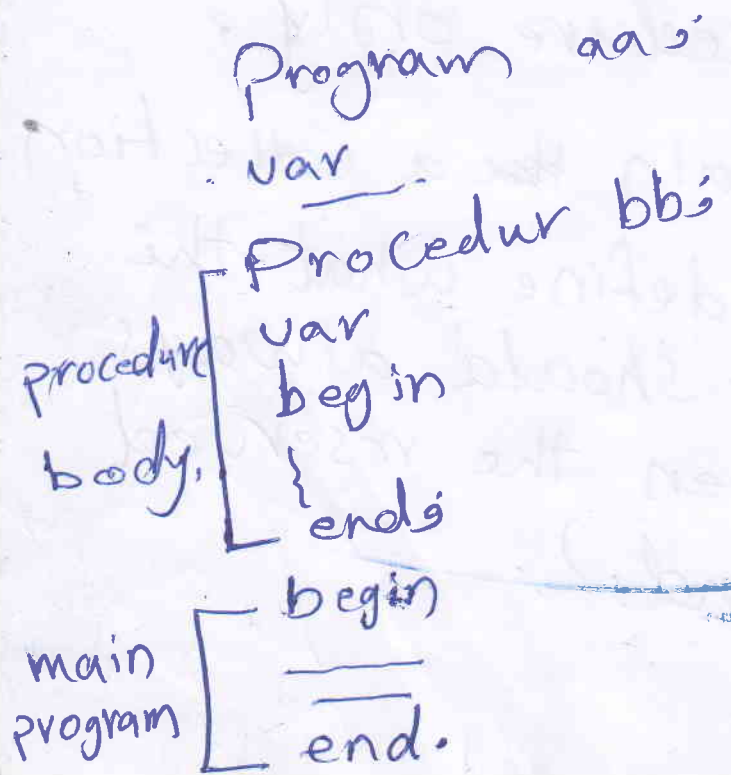
→ global variables :-

المتغيرات العالمية وتعرف داخل البرنامج الرئيسي main prog. والاقترانات Procedure, function

Q:- what is the differ between the local & global variables?

## note:

the Procedure must be declare above of the main program. like.





# calling procedure:-

while creating a procedure, you give a definition of what the procedure has to do. to use the procedure, you will have to call that procedure to perform a defined task. when a program calls a procedure, program control is transferred to the called procedure. a called procedure performs the defined task.

ex: Program example:  
procedure print;

↑ name of procedure.

```
procedure  
body {  
begin  
  writeln('press any key to continue');  
  readln;  
end;  
begin  
  writeln('my name is Ali');  
  print;  
  writeln('I like pascal language');  
  print;  
  writeln('good luck');  
  print;  
end;
```

Note:

if a local variable is declared  
inside a procedure with the

same name as the global variable

the global variable is overridden.  
عندما في الرتبة الأعلى

is Program ~~xxxxxx~~ without Proceed  
become

## Program examples

Begin

writeln('my name is Ali');

writeln('press any key to continue');

readln;

writeln('i like pascal language');

writeln('press any key to continue');

readln;

writeln('good luck');

writeln('press any key to continue');

readln;

end.

program aa;

ex:- Procedure class

var

a: integer;

Begin

a := 10;

writeln(a);

end;

Begin

class

a := 20;

end.

فوف يتم استدعاء الاجراء class  
من المتغير a كما في المثال  
مرفوع مع ان المتغير محلي  
خاص داخل الاجراء فقط.

فقط لأن المتغير مرفوع فقط  
خل الاجراء.



```

var
  B: integers;
  Procedure object;
var
  A: integer;
Begin
  A := 10;
  B := 64;
  writeln (a);
End;

```

قانوني شرعي، جائز  
 (legal)

```

Begin
  object;
  A := 20;
  B := 50;
End.

```

لانه متغير محلي لا يجوز المطالبة قيمته داخل البرنامج الرئيسي

غير قانوني  
 illegal  
 legal

مفادىء بأن المتغير العالمي يمكن تغييره داخل الاجراء والبرنامج، اما المتغير المحلي فلا يمكن ذلك الا من خلال الاجراء فقط.  
 كما انهم الامتدادات عن المتغير المحلي والعالمى بنفس الاسم، فأت العالمى سيستجاوز على المحلي كما في المثال الاتي .

\* write a pascal procedure called welcome which print the text string "welcome to pascal"

ans! Procedure welcome  
begin  
writeln('welcome to pascal')  
end;

\* write a pascal procedure called multiply, which accept 2 integer number 1 and number 2, and print the result of multiplying the 2 integers together.

ans! procedure multiply  
var  
number1, number2 : integer;  
Result : integer;  
begin  
Result := number1 \* number2;  
writeln(Result);  
end;

## Exercises

write Pascal Statement to define an array called numbers, which is an integer array with element ranging from 1 to 20

answer

```
var numbers: array [1..20] of integer;
```

write Pascal Statement which sum the content of an integer array called mynumbers, which has 20 elements numbered 1 to 20.

answer!

```
total := 0;
```

```
for loop := 1 to 20 do
```

```
total := total + mynumbers[loop];
```

write a Pascal Statement to display the ASCII value of the letter 'A'

answer!

```
writeln(ord('A'));
```



Write pascal statement to display the character represented by the ASCII value 52.

answer writeln (chr (52)) ;

\* Write a pascal statement to display the character which follows 'F'.

ans ; writeln (succ ('F')) ;

\* Write pascal statement which display the character comes before 'Z'.

answer writeln (pred ('Z')) ;

## ion Functions:

- ABS

the ABSolute Function returns the absolute value of either an integer or real;

ABS(-21) returns 21

ABS(-3.5) returns 3.50000+5

- COS:

~~returns~~ the Cosine Function returns the Cosine value.

COS(0) returns (1.0).

- EXP: the exponential function calculates e raised to the power of number.

EXP(10) returns e to the Power of 10.

- SIN:- returns the sine of the value.

SIN(90/2) returns  $\frac{1}{\sqrt{2}}$

or: return the square.

SQR(2) return 4.

- SQRT: Function returns the square root of <sup>the</sup> value.

SQRT(4) returns 2.0000

- TRUNC: return the whole part (no decimal places) of a real number

TRUNC(4.87) return 4.

mis