

# Characters & Strings

هو رمز واحد فقط — يمكن أن يكون حرفًا واحدًا، أو رقمًا واحدًا، أو علامة ترقيم، أو مسافة واحدة.

أمثلة:

!

Punctuation

—

Space

7

Digit

b

Letter  
(lowercase)

A

Letter  
(uppercase)

## Declaration syntax:

```
Var C : character;
```

## ملاحظة:

يُكتب هذا الرمز بين علامتي اقتباس مفردتين:

'A'      'z'      '3'      '!'

# Character Comparison

## ASCII printable characters

32	space	64	@	96	`
33	!	65	A	97	a
34	"	66	B	98	b
35	#	67	C	99	c
36	\$	68	D	100	d
37	%	69	E	101	e
38	&	70	F	102	f
39	'	71	G	103	g
40	(	72	H	104	h
41	)	73	I	105	i
42	*	74	J	106	j
43	+	75	K	107	k
44	,	76	L	108	l
45	-	77	M	109	m
46	.	78	N	110	n
47	/	79	O	111	o
48	0	80	P	112	p
49	1	81	Q	113	q
50	2	82	R	114	r
51	3	83	S	115	s
52	4	84	T	116	t
53	5	85	U	117	u
54	6	86	V	118	v
55	7	87	W	119	w
56	8	88	X	120	x
57	9	89	Y	121	y
58	:	90	Z	122	z
59	;	91	[	123	{
60	<	92	\	124	
61	=	93	]	125	}
62	>	94	^	126	~
63	?	95	_		

كل character له رقم خاص به (يسمى Code ASCII)  
تتم مقارنة ال characters باستخدام هذه الأرقام، تمامًا كما نقارن  
بين الأعداد الصحيحة

و بالتالي يمكننا القول:

- '0' < '9'
- 'A' < 'Z'
- 'a' > 'A'
- 'a' < 'b'
- 'Z' < 'a'

**A<sub>B</sub>C** Is a letter?

```
If ('a' <= C) And (C <= 'z')  
Or ('A' <= C) And (C <= 'Z')  
Then ...
```

**12  
34** Is a digit?

```
If ('0' <= C) And (C <= '9') Then ...
```

**↑** Is uppercase?

```
If ('A' <= C) And (C <= 'Z') Then ...
```

**↓** Is lowercase?

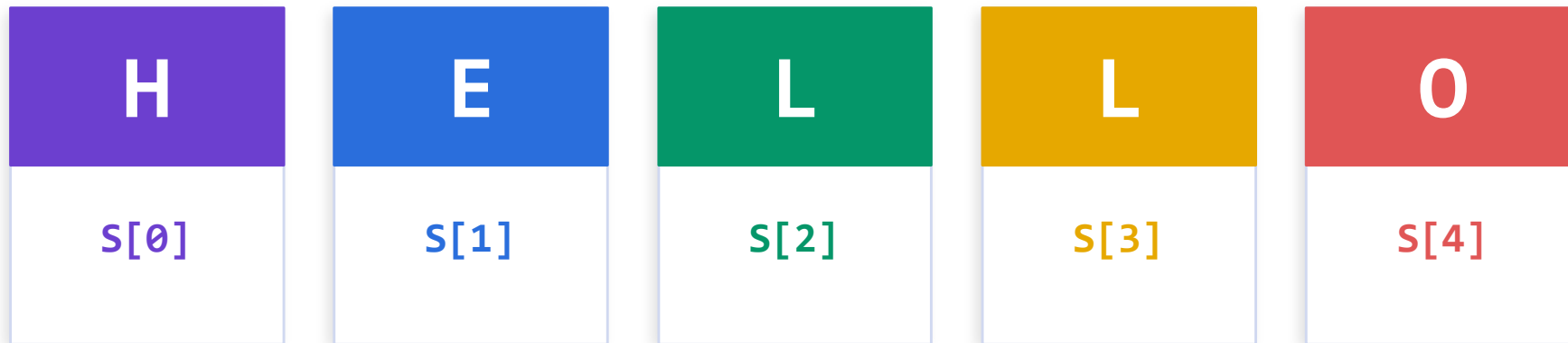
```
If ('a' <= C) And (C <= 'z') Then ...
```

```
Algorithm Classifier;  
Var C : character;  
Begin  
  write('Enter a character: ');  
  read(C);  
  If ('a'<=C) And (C<='z') Or ('A'<=C) And (C<='Z')  
  Then  
    write('It is a letter');  
  Else If ('0'<=C) And (C<='9') Then  
    write('It is a digit');  
  Else  
    write('Other character');  
  EndIf  
End.
```

اكتب خوارزمية تقرأ  
**character** واحدًا من  
المستخدم، ثم تحدد نوع هذا  
الحرف.

السلسلة النصية (String) هي تسلسل من الcharacters — قد تكون كلمة، أو جملة، أو أي نص يتم تخزينه في متغير واحد.

**مثال:** السلسلة النصية " HELLO " تتكوّن من 5 characters.



## Declaration syntax:

```
Var S : string;
```

## Getting the length with length(s):

```
Var S : string;  
    n : integer;  
n := length(S);
```

**length(S)**

تعطي عدد ال characters الموجودة في السلسلة النصية S

**S[i]**

تتيح الوصول إلى ال character الموجود في الموقع i

- S[0] هو ال character الأول.
- S[length(S)-1] هو ال character الأخير

```
Algorithm Traversing;  
Var S : string;  
    i,n : integer;  
Begin  
    write("Enter a string");  
    read(S);  
    n := length(S);  
    For i:=0 to (n-1) do  
        write(S[i]);  
    EndFor  
End
```

اكتب خوارزمية لزيارة كل  
character في السلسلة  
النصية واحدًا تلو الآخر  
باستخدام حلقة For

```
Algorithm CountChar;  
Var S : string; C : character;  
    i,n,count : integer;  
Begin  
    read(S); read(C);  
    n := length(S);  
    count := 0;  
    i := 0;  
    While i < n do  
        If S[i] = C Then  
            count := count + 1  
        EndIf  
        i := i + 1  
    EndWhile  
    write(C, ' appears ', count, ' times');  
End.
```

اكتب خوارزمية تعد عدد مرات  
ظهور character محدد داخل  
سلسلة نصية.