



TUHOCC.CC

LIVE ((●))

# TỰ HỌC JAVA siêu tốc



LESSON  
**25**

**char - Ký tự trong Java**

```
char ch = 'x';
```

  
 @galailaptrinh

1

## Kiểu ký tự char

### 1. Khai báo :

✓ Trong java char được sử dụng để lưu trữ một ký tự, ký tự đặt trong nháy đơn

**char** <tên biến> = ' <ký tự> '

```
char ch = 'x'; // khởi tạo (có gán giá trị)
char ch2 = 65; // khởi tạo (dùng mã ASCII)
char ch3 ; // khai báo biến (0 gán giá trị)
System.out.println(ch2); // trả về A
```

Char	A	B	C	...	a	b
Value	65	66	67	...	97	98

ASCII table

✓ **Chú ý** khai báo char phải để trong **ngoặc đơn**, và **chỉ đc khai báo 1 ký tự** => nếu dùng nháy kép sẽ báo lỗi

```
char ch4 = "k";
```

Required type: char

Provided: String



1

## Kiểu ký tự char

# ASCII Table

Dec	Hex	Oct	Char	Dec	Hex	Oct	Char	Dec	Hex	Oct	Char	Dec	Hex	Oct	Char
0	0	0		32	20	40	[space]	64	40	100	@	96	60	140	`
1	1	1		33	21	41	!	65	41	101	A	97	61	141	a
2	2	2		34	22	42	"	66	42	102	B	98	62	142	b
3	3	3		35	23	43	#	67	43	103	C	99	63	143	c
4	4	4		36	24	44	\$	68	44	104	D	100	64	144	d
5	5	5		37	25	45	%	69	45	105	E	101	65	145	e
6	6	6		38	26	46	&	70	46	106	F	102	66	146	f
7	7	7		39	27	47	'	71	47	107	G	103	67	147	g
8	8	10		40	28	50	(	72	48	110	H	104	68	150	h
9	9	11		41	29	51	)	73	49	111	I	105	69	151	i
10	A	12		42	2A	52	*	74	4A	112	J	106	6A	152	j
11	B	13		43	2B	53	+	75	4B	113	K	107	6B	153	k
12	C	14		44	2C	54	,	76	4C	114	L	108	6C	154	l
13	D	15		45	2D	55	-	77	4D	115	M	109	6D	155	m
14	E	16		46	2E	56	.	78	4E	116	N	110	6E	156	n
15	F	17		47	2F	57	/	79	4F	117	O	111	6F	157	o
16	10	20		48	30	60	0	80	50	120	P	112	70	160	p
17	11	21		49	31	61	1	81	51	121	Q	113	71	161	q
18	12	22		50	32	62	2	82	52	122	R	114	72	162	r
19	13	23		51	33	63	3	83	53	123	S	115	73	163	s
20	14	24		52	34	64	4	84	54	124	T	116	74	164	t
21	15	25		53	35	65	5	85	55	125	U	117	75	165	u
22	16	26		54	36	66	6	86	56	126	V	118	76	166	v
23	17	27		55	37	67	7	87	57	127	W	119	77	167	w
24	18	30		56	38	70	8	88	58	130	X	120	78	170	x
25	19	31		57	39	71	9	89	59	131	Y	121	79	171	y
26	1A	32		58	3A	72	:	90	5A	132	Z	122	7A	172	z
27	1B	33		59	3B	73	;	91	5B	133	[	123	7B	173	{
28	1C	34		60	3C	74	<	92	5C	134	\	124	7C	174	
29	1D	35		61	3D	75	=	93	5D	135	]	125	7D	175	}
30	1E	36		62	3E	76	>	94	5E	136	^	126	7E	176	~
31	1F	37		63	3F	77	?	95	5F	137	_	127	7F	177	



1

## Kiểu ký tự char

### ❑ 2. Lấy char từ dữ liệu nhập vào :

```
Scanner sc = new Scanner(System.in);
System.out.println("Mời nhập vào 1 ký tự: ");
//đọc dữ liệu từ bàn phím
String input = sc.nextLine();
char ch4 = input.charAt(0); //lấy 1 ký tự đầu
System.out.println("Ký tự vừa nhập: " + ch4);
```

Mời nhập vào 1 ký tự:

a

Ký tự vừa nhập: a





2

## char method

❑ 3. **compareTo**: So sánh 2 ký tự => trả về 1 số nguyên ( dùng lớp **Character** ) :

```
//1. ký tự giống nhau trả về 0
System.out.println(Character.compareTo('a','a'));
//2. ký tự 1 trước 2 trong bảng ASII trả về số âm < 0
//A: 65 , a: 97 => 65-97 = -32
System.out.println(Character.compareTo('A','a'));
//3. ký tự 1 sau 2 trong bảng ASII trả về số dương >0
//a: 97, A: 65 => 97-65 = 32
System.out.println(Character.compareTo('a','A'));
```



0  
-32  
32

✓ **Tóm lại : (So sánh dựa vào mã ASCII)**

- ❖ 2 ký tự trùng khớp nhau thì **Compare =0**
- ❖ Ký tự 1 < ký tự 2 thì kết quả âm
- ❖ Ký tự 1 > Ký tự 2 thì kết quả dương



2

char method

STT	Cú pháp	Nội dung
1	<i>Character.isDigit()</i>	<i>True nếu ch truyền vào là chữ số (0,1,2....)</i>
2	<i>Character.isLetter()</i>	<i>True nếu ch truyền vào là chữ cái (a,b,c,A..)</i>
3	<i>Character.isWhitespace()</i>	<i>True nếu ch truyền vào là space</i>
4	<i>Character.isLowerCase()</i>	<i>True nếu ch truyền vào là ký tự thường</i>
5	<i>Character.isUpperCase()</i>	<i>True nếu ch truyền vào là ký tự viết hoa</i>

```
char ch5 = '1';
char ch6 = 'a';
char ch7 = 'A';
char ch8 = ' ';
```

```
System.out.println(Character.isDigit(ch5));
System.out.println(Character.isLetter(ch6));
System.out.println(Character.isLowerCase(ch6));
System.out.println(Character.isUpperCase(ch7));
System.out.println(Character.isWhitespace(ch8));
```

