Tokens

The elements of Java program are known as tokens.

1) Keywords :- These are the (~~names provided by the~~) predefined words who's meaning Cannot be changed by the developers.

→ All Keywords must be in (Keywords) lowercase.

Ex :- Class, extends, implements, through, throws, byte, Short, int, float, double, if, else, Switch, etc. Public, Private, Product, static, void, boolean, try, catch, final, finally, break, continue, interface, char,

2) Identifiers :- These are the names provided by the developpers to identify a class (or) method (or) Variables, interface, etc.

→ While providing the identifiers we must be aware of

i) Rules ii) Conventions.

(i) Rules :- Rules are mandatory otherwis syntax become wrong.

ⓐ Keywords cannot be used as identifiers.

ⓑ It Should not start with the numbers.

ⓒ $, _ (doller, underscore) Special characters Can be used in the identifiers.

(ii) Conventions :-

these are the Standard practicess followed to recognise the Identifiers.

1) Each words in the Java program name must Start with Upper case letters.

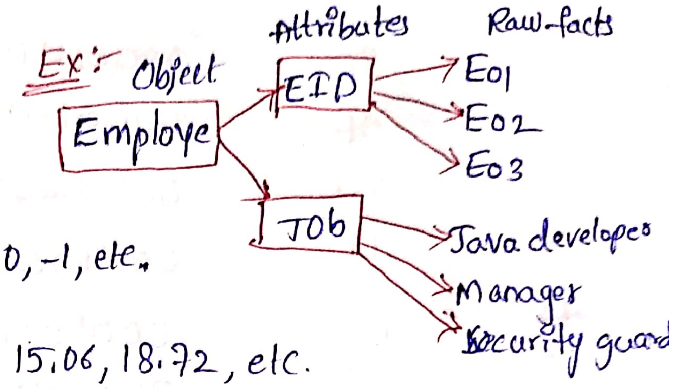
2) For methods name first word must be in lower care but remaining words must start with upper Case letters.

3) The class name must be Similar to Java file name.
(Java Programme Name)

3) Literals / Data / Values :-

These are the raw-facts (or) Actual-facts which describes on attribute (Common feature) amongst particular time of object).

We have 4 types of data



1) Numbers → Integer No: 1, 10, 0, -1, etc.
→ Floating no: 35.56, 15.06, 18.72, etc.

2) Single Character → 'a', '1', '$', ' ', '5', '~~10~~'

3) String → " ", "Rita", "Rajkumar", "1".

4) Boolean → true, false.
(1) (0)

ASCII — American standard Code for Information Interchange

ASCII Table

Dec	Hex	Binary	char.
0	00	00000000	Null
1	01	00000001	SOH
2	02	0000 0010	STX
3	03	0000 0011	ETX
4	04	0000100	EOT
5	05	0000 0101	ENT
6	06	0000 0110	ACK
7	07	0000 0011	BEL
8	08	00001000	BS
9	09	0000 1001	HT
10	0A	0000 1010	LF

11	0B	00001011	VT
12	0C	00001100	FF
13	0D	00001101	CR
14	0E	00001110	SO
15	0F	00001111	SI
16	10	00010000	DLE
17	11	00010001	DC1
18	12	00010010	DC2
19	13	00010011	DC3
20	14	00010100	DC4
21	15	00010101	NAK
22	16	00010110	SYN
23	17	00010111	ETB
24	18	00011000	CAN
25	19	00011001	EM
26	1A	00011010	SUB
27	1B	00011011	ESC
28	1C	00011100	FS
29	1D	00011101	GS
30	1E	00011110	RS
31	1F	00011111	US
32	20	00100000	Space
33	21	00100001	!
34	22	00100010	"
35	23	00100011	#
36	24	00100100	$
37	25	00100101	%
38	26	00100110	&
39	27	00100111	'
40	28	00101000	(

41	29	00101001	)
42	2A	00101010	*
43	2B	00101011	+
44	2C	00101100	,
45	2D	00101101	—
46	2E	00101110	.
47	2F	00101111	/
48	30	00110000	0
49	31	00110001	1
50	32	00110010	2
51	33	00110011	3
52	34	00110100	4
53	35	00110101	5
54	36	00110110	6
55	37	00110111	7
56	38	00111000	8
57	39	00111001	9
58	3A	00111010	:
59	3B	00111011	;
60	3C	00111100	<
61	3D	00111101	=
62	3E	00111110	>
63	3F	00111111	?
64	40	01000000	@
65	41	0100001	A
66	42	01000010	B
67	43	01000011	C
68	44	01000100	D
69	45	01000101	E
70	46	01000110	F
71	47	01000111	G
72	48	01001000	H
73	49	01001001	I
74	4A	01001010	J
75	4B	01001011	K

76	4C	01001100	L
77	4D	01001101	M
78	4E	01001110	N
79	4F	01001111	O
80	50	01010000	P
81	51	01010001	Q
82	52	01010010	R
83	53	01010011	S
84	54	01010100	T
85	55	01010101	U
86	56	01010110	V
87	57	01010111	W
88	58	01011000	X
89.	59	01011001	Y
90	5A	01011010	Z
91	5B	01011011	[
92	5C	01011100	\
93	5D	01011101	]
94	5E	01011110	^ (caret / circumflex)
95	5F	01011111	_
96	60	01100000	`
97	61	01100001	a
98	62	01100010	b
99	63	01100011	c
100	64	01100100	d
101	65	01100101	e
102	66	01100110	f
103	67	01100111	g
104	68	01101000	h
105	69	01101001	i
106	6A	01101010	j
107	6B	01101011	k
108	6C	01101100	l
109	6D	01101101	m

110	6E	01101110	n
111	6F	01101111	o
112	70	01110000	p
113	71	01110001	q
114	72	01110010	r
115	73	01110011	s
116	74	01110100	t
117	75	01110101	u
118	76	01110110	v
119	77	01110111	w
120	78	01111000	x
121	79	01111001	y
122	7A	01111010	z
123	7B	01111011	{
124	7C	01111100	|
125	7D	01111101	}
126	7E	01111110	~
127	7F	01111111	DEL