JAVA (Tokens)

ATUL KUMAR (LINKEDIN)
@ ATUL.IN (INSTAGRAM).

? **What is Tokens? full explaination.**

Token is the smallest elements of a program that is identified by a compiler.

Every java statements and expressions are created by using tokens.

Example :-

```
  → {   Class A
  →  {   p.s.v.m (string a[] )

            int a = 10 , b = 20;
            int c = a + b;
            S.O.P ( c ) ;
  →   }
  → }
```
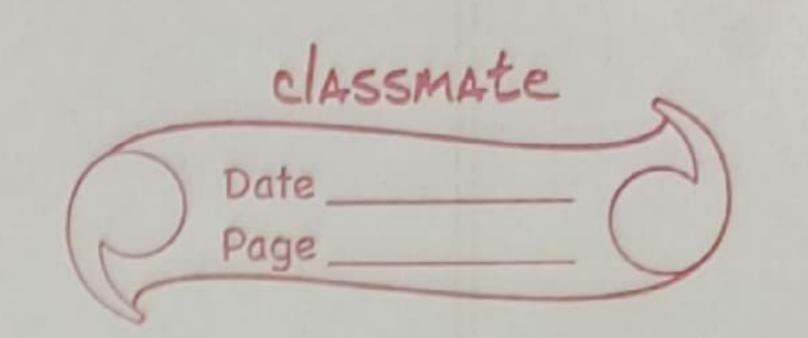
NOTES GALLERY (TELEGRAM).
@ NOTES GALLERY (INSTAGRAM)

Tokens :-

1). Keyword
2). Identifier
3). Operator
4). Separator
5). Literal

Ex :-   1). Keyword :- int, Static, Void

        11). Identifier :- Variable, Method, class-name
                    (It refers name)
                         only.

1.

JAVA (Tokens)

ATUL KUMAR (LINKEDIN)

? What is Tokens ? full explaination.

Token is the smallest elements of a program that is identified by a compiler.

Every java statements and expressions are created by using tokens.

Example :-

```
Class A
{
    p.s.v.m (string a[] )
    {
        int a = 10 , b = 20;
        int c = a + b ;
        s.o.p ( c ) ;
    }
}
```
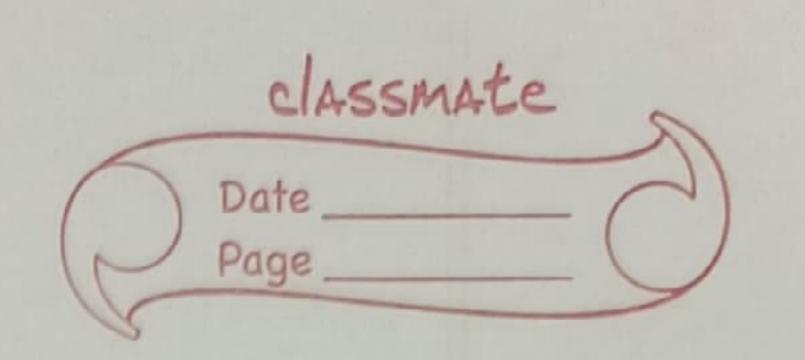
Tokens :-

1). | Keyword
2). | Identifier
3). | Operator
4). | Separator
5). | Literal

Ex :-   1). Keyword :- int, static, void

        11). Identifier :- Variable, Method, class-name
                    (It refers name)
                        only.

1.

III). Operator :- +, -, *, /, %, &&, !!, ++, --

IV). Separator :- ; , . : ( ) { }

Java Keyword

? What is Keyword? full explaination.

✓ Keyword's are the reserved word whose meaning is already defined in the compiler called Keywords.

Java Keywords :-

int	Continue	instance of	Super	boolean
short	break	interface	this	abstract
byte	default	native	Switch	case
long	double	new	Synchronized	catch
if	entends	package	throw	class
else	implements	Private	throws	Const *
for	final	Protected	try	goto *
while	finally	Public	transient	Strictfp **
do	float	return	void	enum ****
char	import	Static	volatile	assert ***

Note :- 1).
  * (Not used)
  ** (Added in 1.2 v)
  *** (Added in 1.4 v)
  **** (Added in 5.0 v)

2-

ii>. <u>true</u>, <u>false</u> and <u>null</u> just like keyword, but they are actually literals.

iii>. You can't use any of the following as identifiers in your program.

    i>. int double = 10;

    ii>. void int ( )
            {

            }

    iii>. A char = new ( );

<u>Java Identifiers</u>

? <u>What is Identifier ? full explaination.</u>

In Java, an identifier is the name of a variable, method, class, package or interface that is used for the purpose of identification.

Ex :-   Class A
        { int a = 10;
         p.s.v.m (String a [ ])
        {
            int a = 10;
        }

        void m1 ( )
         {

3.

S.O.P (a);
```
            }
        }
    }
```

Java identification

Note :-

I). Keyword can't be used as a identifier.

II). Identifier are case-sensitive.

III). We can't use whitespace in between identifier.

IV). Identifier always starts with letters, $ or ___

Java (Operator)

? **What is Operators ? With types.**

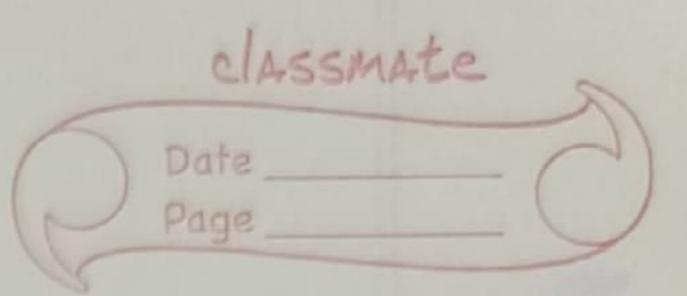☑ Operator is a symbol that is used to perform operations according to user requirement. (Variable & values)

Types

I). Arithmetic $(+, -, *, \div, \%)$

II). Relational $(<, >, >=, <=, !=, ==)$

41

iii). Logical (&& , ;; , !)

iv). incr / decr → Pre incr / Post incr (++ Exp, Exp++)
↘ pre decr / post decr (-- Exp, Exp --)

[unary.

v). Assignment → Simple (=)
↘ Compound (+=, -=, *= etc)

vi). ternary (?:)

vii). Bitwise (AND, OR, XOR, complement)

III). Logical (&& , ;; , !)

IV). incr / decr → Pre incr / Post incr (++ Exp, Exp++)
↘ pre decr / post decr (--Exp, Exp--)

[unary.

↳ v). Assignment → Simple (=)
↘ Compound (+= , -= , *= etc)

VI). ternary (?:)

VII). Bitwise (AND , OR , XOR , complement)