



Today's agenda

↳ Intro

↳ output

↳ operators

↳ Data types



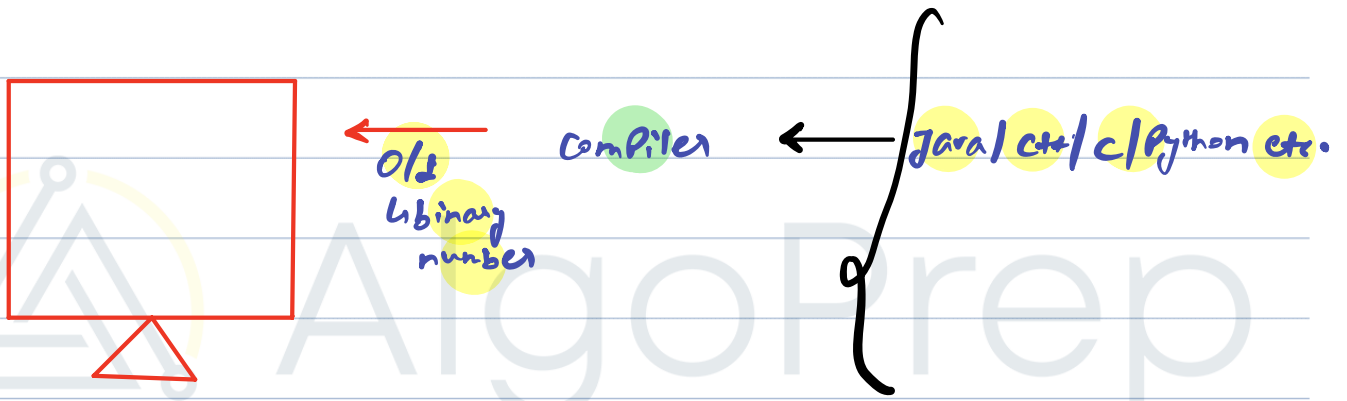
AlgoPrep



* Computer is ! → dumb

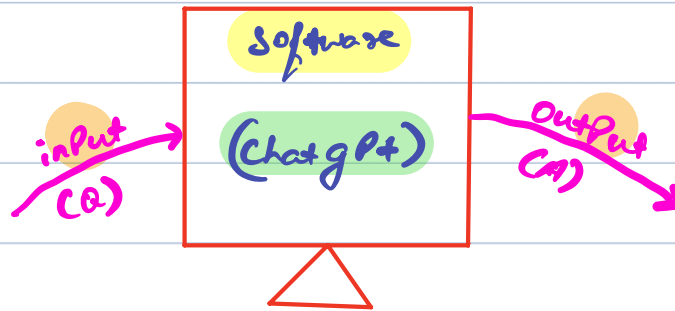
→ you use english for instructions in day to day life.

↓
Java/C++/C/Python etc.



→ english → rules → grammar

→ Java → we follow rules
↳ syntax



IDE

↳ Integrated development environment
↳ Eclipse / vscode / IntelliJ etc.] online ide / editor

* Rule 1:

↳ output

↳ `System.out.println(10);`
↳ next line

`System.out.println(7);`

console

10

7



↳ `System.out.print(10);`

`System.out.println(7);`

→ `System.out.println(8);`

console

107

8





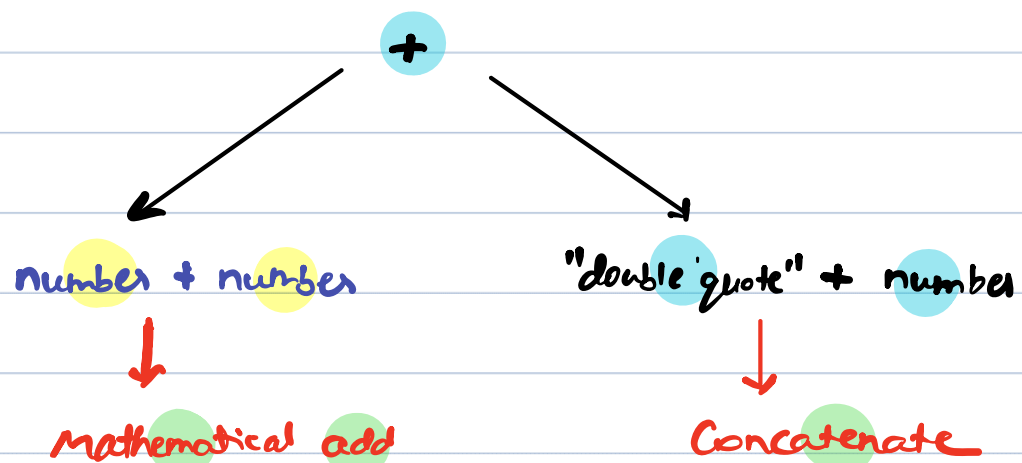
`System.out.print(10*9);` → 90

↳ Double quote Printing

`System.out.println("AlgoPrep");` → AlgoPrep

↳ Plus sign in output

`System.out.println(7+10 + "Hello")` → 70 Hello





operators

$\hookrightarrow + - * / () \rightarrow \text{BODMAS}$



$$2 + 2 \div 2 = 3$$

Rank 1: $()$

Rank 2: \div

Rank 3: $*$

Rank 2: Divide / multiply

\hookrightarrow left to right (Computer Process order)

Rank 4: $+$

Rank 5: $-$

Rank 3: add / subtract

\hookrightarrow left to right (Computer Process order)

Break time 10:34 PM



* Data types

↳ Numbers, Characters, boolean etc.

Integers

decimal

true

false

add 2 numbers
↓
input



let the first no. be x ;

let the second no. be y ;

System.out.println(x+y);

→ $\text{int } x;$
 $x = 30;$

type name
int x ;
 $x = 30;$
↑
equivalent
value

= ← assignment
↳ assign right side to left side.



↓
LHS should always be a variable.

Ex1:

```
int temp;
```

```
temp = 20;
```

```
temp = 100;
```

```
→ System.out.println(temp); → 100
```

100
20
temp

Ex2:

```
int temp;
```

```
temp = 20;
```

```
System.out.println(temp); → 20
```

```
temp = 100;
```

```
→ System.out.println(temp); → 100
```

100
20
temp

Ex3:

```
int temp;
```

```
temp = 100
```

```
↳ int temp; → error
```

```
temp = 20;
```

```
System.out.println(temp);
```

100
temp

temp

↳ Same variable name can't be created more than once.



Creating variables

↖ declare variable
`int temp;`

↑ assign the value/
initialization
`temp = 20;`

`int temp = 20;`

↳ declare as well as
initialize.



AlgoPrep



// Input

↳ Scanner ^{name}sch = new Scanner ^{once in a Code}(System.in);

int temp = sch.nextInt();

↓
integer

No such element exception → Not given the input.



AlgoPrep