

WEEK 1 - DAY 1

⇒ Concept 1 :- IO Operations:

- Here. Output. Operation. is very simple in java.

Syntax :-

`System.out.print("Welcome to BOOTCAMP");`

- ↓
• This is used to print the code in same line.

- For printing the code in next

line :-
↓

`System.out.println("IO Operations");`

Input Operations :-

- For input operations we have to create scanner class in java.

- For this we have to import the scanner file from the import java.util.Scanner; command.

• Syntax :-

`{ Scanner variable1 = new Scanner(System.in);
 int variable2 = variable1.nextInt();`

Example:-

```
Scanner sc = new Scanner(System.in);  
int n = sc.nextInt();
```

2). Some of the scanner methods for.

Primitive data types are:-

- int → nextInt();
- float → nextFloat();
- double → nextDouble();
- long → nextLong();
- short → nextShort();
- byte → nextByte();
- boolean → nextBoolean();

Note:- Only for the word use use only next().

i.e. word → next();

2). Coding in Java Follows the Camel.

Casing Concept.

Ex :- 1) pradhviGopal.

2) studentName.

06/06/24.

Week 1 - Day 2

Concept 2 :- Arithmetic Operations along with Type Casting.

(i) What is meant by type casting?

A] • Type Casting means Data Type Conversion.
• It also follows the order:-

Byte, short, int, long, float, double.
Conversion Possible.

Conversion Not Possible.

⇒ If we assign `int = char` → This lead to compilation error.

Ex. :-

float x = 12.24;

int a;

a = x;

Error

↓

if we want to run this we have to force the Java compiler by using

Syntax:

a = (int) x;

a = (data-type) b;

Operators :-

* We have 7 types of Operators in Java.

1. Arithmetic Operators :-

+ \rightarrow Addition

- \rightarrow Subtraction

* \rightarrow Multiplication

/ \rightarrow Division

% \rightarrow Modulo Division gives remainder.

2. Unary Operators :-

++ \rightarrow Increment $\left\{ \begin{array}{l} \rightarrow ++a \rightarrow \text{Pre Increment} \\ \rightarrow a++ \rightarrow \text{Post Increment} \end{array} \right.$

-- \rightarrow Decrement $\left\{ \begin{array}{l} \rightarrow --a \rightarrow \text{Pre Decrement} \\ \rightarrow a-- \rightarrow \text{Post Decrement} \end{array} \right.$

3. Assignment Operators :-

= \rightarrow Equal to $\rightarrow a = a = b$

+ = $\rightarrow a + = b \rightarrow a = a + b;$

- = $\rightarrow a - = b \rightarrow a = a - b;$

* = $\rightarrow a * = b \rightarrow a = a * b;$

/ = $\rightarrow a / = b \rightarrow a = a / b;$

} \rightarrow Operation to be Performed.

4] Relational Operators:-

$\begin{matrix} = \\ > \\ < \\ >= \\ <= \\ != \end{matrix}$ } \rightarrow Used to perform operations based on Relations. } \rightarrow Prints true or false.

5] Logical Operations:-

$\begin{matrix} \text{OR} \rightarrow \vee \\ \text{AND} \rightarrow \&\& \\ \text{NOT} \rightarrow ! \end{matrix}$ } \rightarrow Performs logical operations based on their truth tables of the operators.

6] Ternary Operations:-

`int a = 7;`
`int b = 10;`

`int max = (a > b) ? a : b;`
if condition satisfies, max value = a; } else, max value = b;

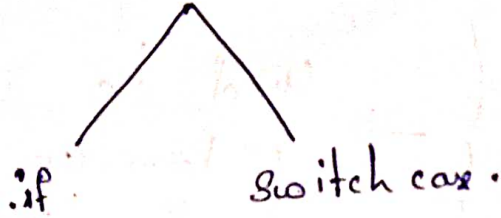
7] Bitwise Operators:-

$\&$ \rightarrow Single bit perform AND operation.
 $|$ \rightarrow Single bit perform OR operation..
 $>>$ \rightarrow Right shift operation.
 $<<$ \rightarrow Left shift operation.

07/06/24

Week-1 Day 3

Concept 3 :- Conditional statements



⇒ IF Condition :-

This is conditional statement i.e. used to perform a specific task based on the condition. In if statement, if the condition exists then only executes the

Program.

- In if else condition if the condition 1 is true then the first statement block will ~~execute~~ executed otherwise else block will be executed.
- In nested if condition the if condition itself has a inbuilt if condition in.

its
Syntax :-

① if (condition) {
}
}

② if (Condition) {

```
    Block-1;  
}  
else  
{  
    Block-2;  
}
```

③ if (cond) {

```
    if (condi)  
        Block 1;  
    }  
    }  
else {  
    Block - 2;  
}
```

27. Switch Case :-

• Switch case is used for the to perform the multiple operations based on the given expression. The operation related to given expression will be performed.

Ex :-

Exp = a + b;

switch (Exp) {

case :

break;

case b :

break;

case c :

break;

default :

break.

}

4

08/06/24.

Week 1 - Day 4

Concept 4:- Taking character as an input & also concept of loops.

① Taking Character Input:-

- In java you can take an input as a word. is easy as same as using next() syntax.
- But taking a single letter as an input in a word makes a special case. For this you need to use character input.

• Syntax:-

`char ch;
ch = input.next().charAt(0);`

In this, H E L L O
 0 1 2 3 4

Represents the single character.

Ex:-

Q. Print:- 1, if input character b/w (A-Z)

-1, if input character b/w (a-z)

0, if input not a character.

⇒ Verify it in the VS Code provided in Github.

27. Loops:-

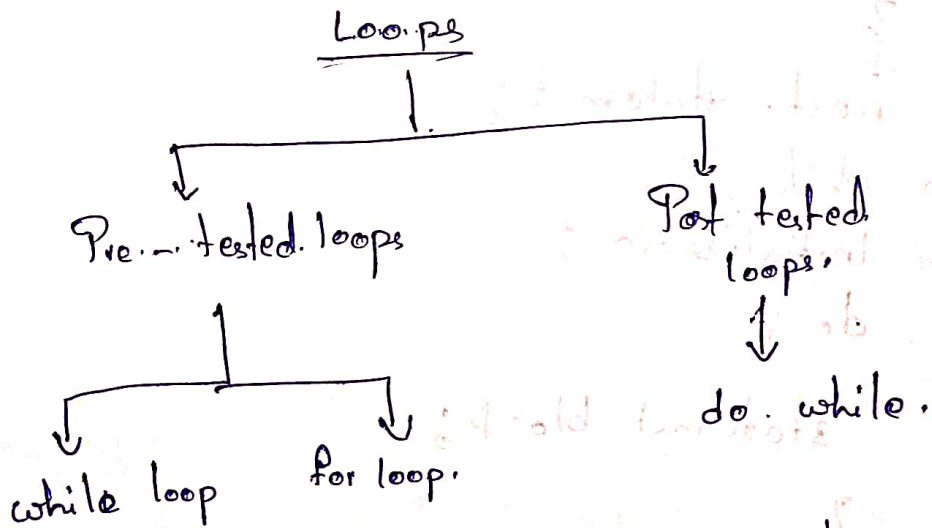
- A loop is thing that executes the statement block continuously until the condition satisfies.

- We have three parts in a loop:-

→ Initialization. → Says where to start.

→ Condition. → Says where to end.

→ Increment/Decrement. → Says how many ~~steps~~ ^{inc/dec} steps required to each iteration.



- In Pre-tested loops first the loop condition execute then if condition satisfies after that statement block executes. Otherwise it will terminate the loop.

⇒ while Block :- Initialisation ,
while (condition) ;

```
{  
    statement block ;  
    increment / Decrement ;  
}  
next statement ;
```

⇒ For loop Block :-

```
For (initialization ; condition ; inc./dec.) {  
    statement block ;  
}  
next statement ;
```

⇒ Do while :-
Initialisation ;
do {

```
    statement block ;
```

```
}
```

```
while (condition) ;
```

Ex :- Menu Drive Applications.

09/06/24

Week 1 - Day 5

Concept 5 :- Methods, Break & Continue :

- Method in java is nothing but the Functions Concept.
- Method consists of a Block of code. This Block is named with a name called Method name.
- If we want to have an operation to be performed multiple times that block will be written as a method.
- Then whenever it is necessary during that time we can call that method & use it.

27. Advantages :-

- No need of Code Repetition.
- Code will be clean & Understandable.
- Increased Readability.
- Space Reduction.

27. Syntax of Methods:-

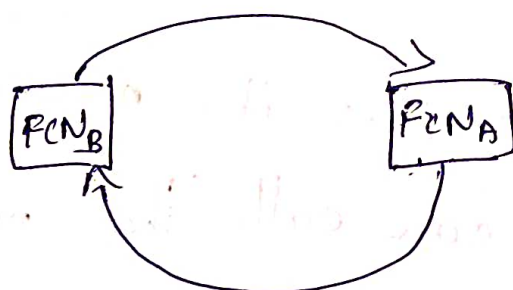
return type function Name (Parameters) {

}

Block of PCN Execution;

(I.e. we have any parameters that will be sent along with the code)

Function



return type

• Return type:- I.f. the function need to

be return value to the function caller then

we can ~~can~~ apply the return type based on that value.

27. Keywords Break & Continue:-

① Break:-

- Break statement will be used to break the loop & stop execution of remaining iterations & remaining block statements in the loop.

Ex:-

```
int i = 0;
while (i < 5) {
    if (i == 2) {
        break;
    }
    System.out.println(i);
    i++;
}
```

Output

0

1

② Continue:-

- Continue is used only when we want to skip a particular iteration in the loop.

Ex:-

```
int a = 0;
while (a < 5) {
    if (i == 2) {
        i++;
        continue;
    }
    System.out.println(i);
}
```

Output:-

0

1

3

4

10/06/24

Week 1. - Day 6

① Arrays:-

- Variables & characters are only used to store the single value information.
- Here came the problem when you want to store a large amount of data using the variables.
- So to avoid this problem the concept of arrays are introduced.
- An array is used to store the multiple elements of the same data type.

Syntax:-

- Arrays are used to create objects in java by using the key word new.
`datatype[] array name = new datatype[size];`

Ex:-

`int[] array1 = new int[5];`

- To access the elements
`array[index]` | `array1[1] = 2;`

- This single element insertion is not possible for all times when we want to insert multiple elements.

- In that case the following example is used:-

int n; \rightarrow For array size.

int n = sc.nextInt(); \rightarrow storing array size.

int a[n] \rightarrow Declaration of an array

for (int i = 0; i < n; i++) {

a[i] = sc.nextInt();

}

\rightarrow In this block, single single element stored in the array.

- The above to store multiple elements in the arrays at a time easily.

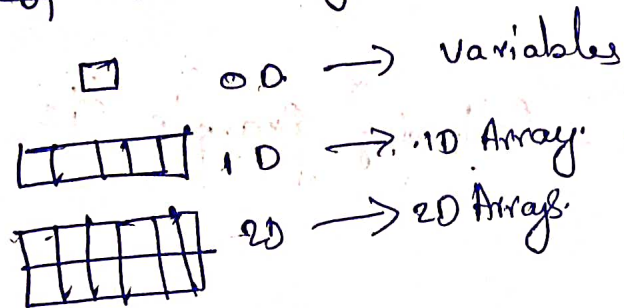
- What if you want to store elements in the both rows & columns?

11/06/24

Week 1 - Day 7

Concept: 7 2D Array

Types of Arrays :-



- The last one 2D → arrays.
- In this 2D array we can store the elements in such a way that in both rows & columns.

- This can be used in the Matrix.

Applications :

- This is also used in the 2 Dimensional shapes like the rectangle, square etc...

Syntax :-

datatype array name [] [] = new datatype [] []

81 = No of rows.

20

at the bottom