

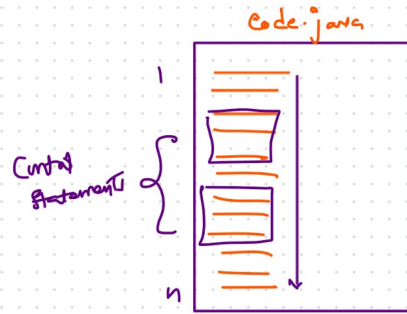
Basic Structure

Control Statements

— continuation

① if (condition) {
 ↓ true
 one condition
 } else {
 ↓
 }
 }

② if-else ladder
 if (condition) {
 ↓ true
 } else if (condition) {
 ↓
 } else {
 ↓
 }



1-100
 Umpire
 25
 4
 20
 5
 55
 win / lost
 re-play

Nested if

if (expression - out) {
 ↓ true
 if (expression in) {
 ↓
 }
 }
 }

Loops

repeated tasks to be executed again & again then we use loops

Structure

for (initialization; condition; updation) {
 ↓ true → it keeps iterating/executing body
 ↓ false → it comes out of loop
 body
 }
 1, 2, 3, 4, 5
 updation logic

S
 S.O.P ("Hi, how are you")
 S.O.P ("Hi, how are you")
 S.O.P ("Hi, how are you")
 S.O.P ("Hi, how are you")
 S.O.P ("Hi, how are you")

⑤

sum = a + b

sum = 2

① sum > sum / 2

sum = a + b

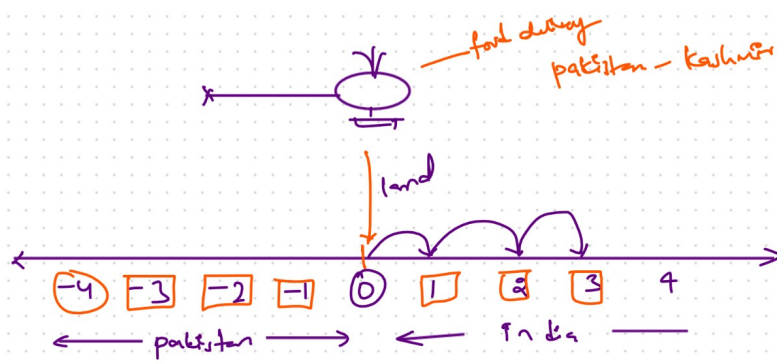
② sum = 2

sum > sum / 2

yield

→ bug

= maintenance



- 1) where to land (initialization)
- 2) check where region/village belong to india (condition)
- 3) if ford driver find pakistans { body }
- 4) we'll decide to go forward/backward (update)

$i = 0$
 $i--$
 $i = i-1$

$0 > -4$
 $-1 > -4$
 $-2 > -4$
 $-3 > -4$
 $-4 > -4$

increment →
 ← decrement

② while loop :

initialization, update

Structure :

while (true/false condition) {

 _____ body

 }

③ do while loop

Structure

do {

at least once

} while (condition);

while (condition) {

 }

once 1, 2, 3

Note :

- 1) for loops updates automatically on each iteration
- 2) where as while loop used to update on manual check