



L5 - Java Practice Session (Intermediate)

RECAP

1. FizzBuzz Problem

 break

2. Break

a) Danger_number

b) FizzBuzz variation: stop as soon as sum of i's exceeds 3^*N

3. Continue

a) FizzBuzz variation: Do nothing if i is a multiple of 4.

 continue

4. Practice:

a) Find number of digits of a given number

b) Find sum of digits of a given number

c) Print sum of digits for all the numbers in a given range [L, R]

L = 5055

R = 5101

19, 20, 21, 22, 23, 24, 25, 26, 27

for (initialisation; condition; updation)
int i = 0; i <= 10 i + 2

Without initialization

int i = 0;

for (; i <= 10; i + 2) {

System.out.println(i);

3

Without condition.

```
int i = 0;
```

```
for( ; ; i += 2) {
```

```
    if (i > 10)
```

```
        break;
```

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

```
}
```

0, 2, 4, 6, 8, 10, 12

Without updation.

```
int i = 0;
```

```
for( ; ; ) {
```

```
    if (i > 10)
```

```
        break;
```

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

```
i += 2;
```

```
}
```

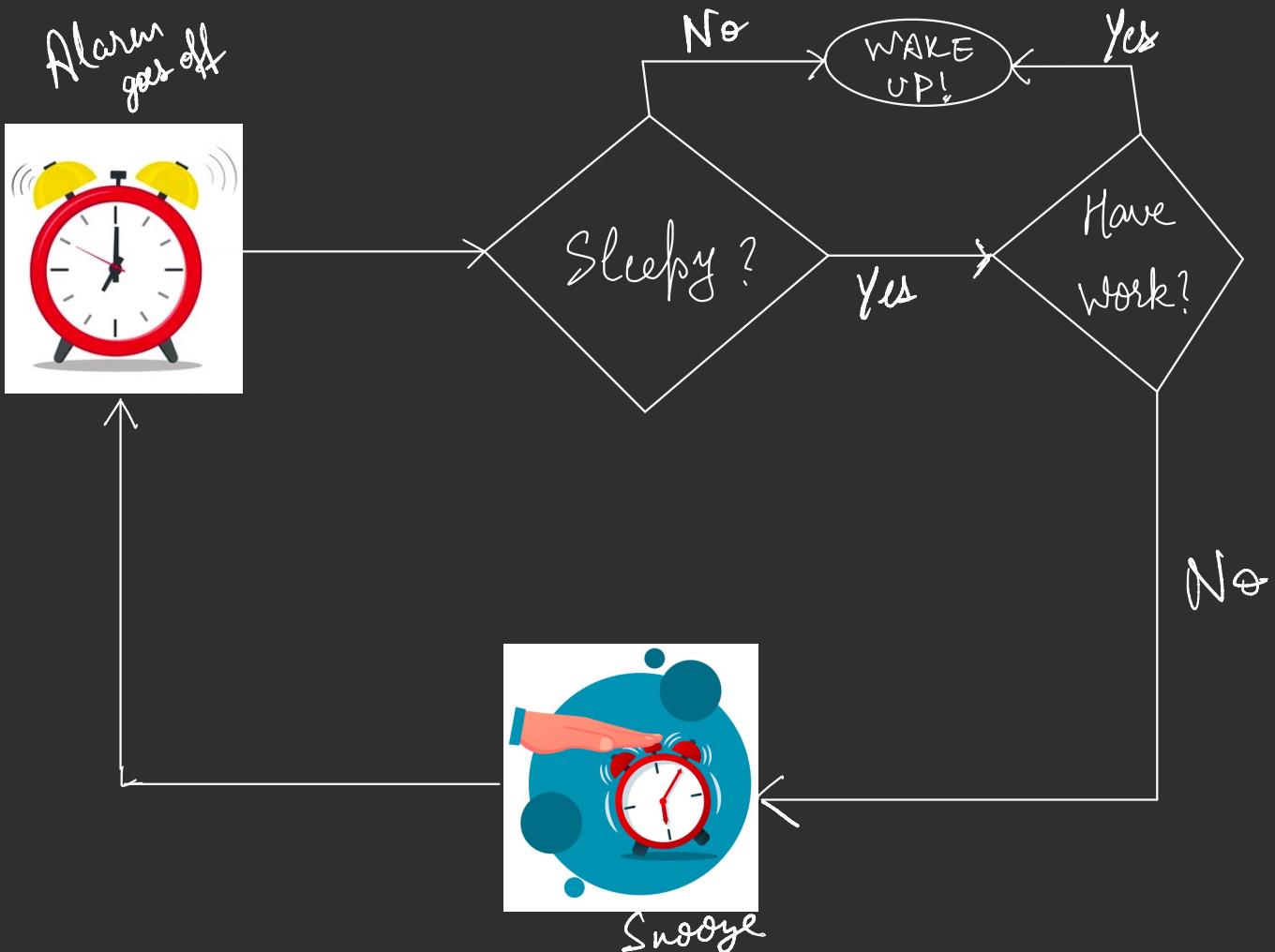
1	19	26
2	15	27
3	16	28
4	17	
5		29
6	18	30
7	19	
8	20	
9	21	
10	22	
11	23	
12	24	
13	25	

Given a value of N,
print only those numbers
from 1 to N that have
sum of digits = K

Eg. R ~ 30
K ~ 9
Output :-

9
18
27

What if updation is
not the same
everytime?



While Loop

```
while (... condition) {  
    // body  
}
```

```
int i = 1;  
  
while ( i <= 10 ) {  
    System.out.println(i);  
    i++;  
}
```

Condition → do → condition → do → condition → do → ~~condition~~

Do While Loop

most problems

do {

// Code

}

while(... condition ...);

NEVER

USED

END

do → condition → do → condition → do → ~~condition~~

$1 \rightarrow 1$

$2 \rightarrow 2$

\vdots

$10 \rightarrow 1$

\vdots

$14 \rightarrow 5$

\vdots

$65 \rightarrow 15$

\downarrow

$69 \rightarrow 15$ (1)

$78 \rightarrow 15$ (2)

Print first N numbers
that have their
sum of digits = K

$87 \rightarrow 15$ (3)

$96 \rightarrow 15$ (4)

$125 \rightarrow 15$

$155 \rightarrow 15$

$156 \rightarrow 12$

$157 \rightarrow 13$

$158 \rightarrow 14$

$159 \rightarrow 15$ (5*)

Input

$N = 5$

$K = 15$

Output:

69

78

87

96

159

Input:

target-count = 9

target-sum = 6

[]

6: 1
↓

Output:

6

105 \Rightarrow 8

15

114 \Rightarrow 9

24

33

42

51

60

Hunt those numbers
having digit sum = K

15: 2
↓

24: 3

More
Practice

A number ≥ 2 is prime if & only if 1 & the number itself are the only 2 factors of it.

Given a positive number, check if it's prime or not

✓ 2 → 1, 2 ✗ 8 → 1, 2, 4, 8

✓ 3 → 1, 3 ✗ 9 → 1, 3, 9

✗ 4 → 1, 2, 4 ✗ 10 → 1, 2, 5, 10

✓ 5 → 1, 5

✗ 6 → 1, 2, 3, 6

✓ 7 → 1, 7

2, 3, 4, 5, ..., N-1

boolean isPrime = true;
for (int i = 2; i <= N-1; ++i) {
 if (N % i == 0) { // N is a multiple of i
 isPrime = false;
 break;
 }
}

System.out.println(isPrime);

$N = 35$; is prime, true

$i=2 \rightarrow i=3 \rightarrow i=4 \rightarrow i=5$

Thank You!