# while loops

```
sop ( "Hello" )
sop ( "Hello" )
sop ( "Hello" )
sop ( "Hello" )
sop ( "Hello" )
```

loops

```
                 True
                 condn
while  ( _ )  {
   |
}
```

```
int cnt=1;
while ( cnt <=5) {
|       sop( "Hello");
|       cnt = cnt+1  | cnt+=1; | cnt++;
}
```

## Syntax

```
1. initialization
   while ( condn) {
         // work
         updation
   }
```

```
int cnt=1;
while ( cnt <=5) {
|       sop( "Hello");
|       cnt = cnt+1 | cnt+=1; | cnt++;
}
```

Ques    Given a no', print its digits.

N= 6531        o/p→ 1
                    3
                    5
                    6

N=123    o/p 3
             2
             1

Extract last digit
↳ N%10

N= 6531

↓ /10        o/p→ 1

653 %10 = 3

↓ /10

65 %10 = 5

↓ /10

6

Incorrect ?

```
1) n= Scn.nextInt();      n = 65~~3~~1  65~~3~~  ~~6~~5  ~~6~~ ~~0~~ 0
2)  while(n>=0) {
3)     Sop(n%10);  →
4)     n= n/10
    }
```

O/P 1
3
5
6
0

```
if (n==0) { print (0) }
else {
1)    n= Scn. nextInt();
2)    while(n> 0) {
3)        Sop (n%10);
4)        n= n/10
      }
}
```

Ques) Given an Integer N, print sum of its digits.   $N \geq 0$

N= 6531   O/P → 15

N= 123   O/P → 6

```
n= Scn. nextInt();
int ans =0;
while(n> 0) {
    ans+= (n%10);
    n= n/10
}
Sop(ans);
```

Ques  Given a number N, Print ~~reverse~~ the no.

Print reverse

→ N = 6531,  O/P → 1356

→ N = 123,  O/P → 321

0 * 10 + Extracted

rev = ~~0~~ 1 10 + 3 ⇒ 130 + 5
↓
1350 + 6
↓
1356

N = 6531 ,  Extracted
↓  ①
653  3
↓
65
↓  5
6  6
↓
0

rev

6531  13
↓
65③

rev = rev × 10 + rem

rev = ~~0~~ ① → 1 × 10 + 3
↓
$$\begin{array}{r} 13 \times 10 \\ +\ 5 \\ \hline 135 \times 10 \\ +\ 6 \\ \hline 1356 \end{array}$$

6531
↓  1
653  3
↓
65  5
↓
6  6
↓
0

n = 123

```
n= Scn. nextInt(); 12~~3~~  1~~2~~  ~~1~~  0
 int rev = 0;  → ~~0~~ ~~3~~  ~~32~~ 321
 while(n> 0) {
        rev= (rev * 10)+ (n%10);
                      320 +      1
       n= n/10
  }
  Sop(ans);
```

```
n= Scn. nextInt();
 int rev = 0;
 while(n> 0) {
        int dig =  n%10 ;
       rev= (rev * 10) + dig
       n= n/10
  }
  Sop(ans);
```

https://www.scaler.com/topics/java/online-java-compiler/?snippet_slug=7d5e6a9cba209df4ed77