n=3

d= 1
2
3

digit → 123

reverse → 321

1	2	3
x100	x10	x1

number = ~~0~~ ~~1~~ ~~12~~ 123

digit = s.nextInt

$0 \times 10 + 1 = 1$

$1 \times 10 + 2 = 12$

$12 \times 10 + 3 = 123$

number x10 + digit

① digit form

② reverse

n = 4

i	digit
1	5
2	2
3	1
4	6

// number form

reverse number

3
2
1

n = 123

rev. = 321

reversed = 0

n = 123

while(n != 0) F

int digit = n%10 → 3

12%10 → 2

1%10 → 1

rev. = rev ×10 + digit → 0 ×10 + 3 = 3

3×10 + 2 = 32

32 ×10 + 1 = 321

n/10 → 123/10 → 12

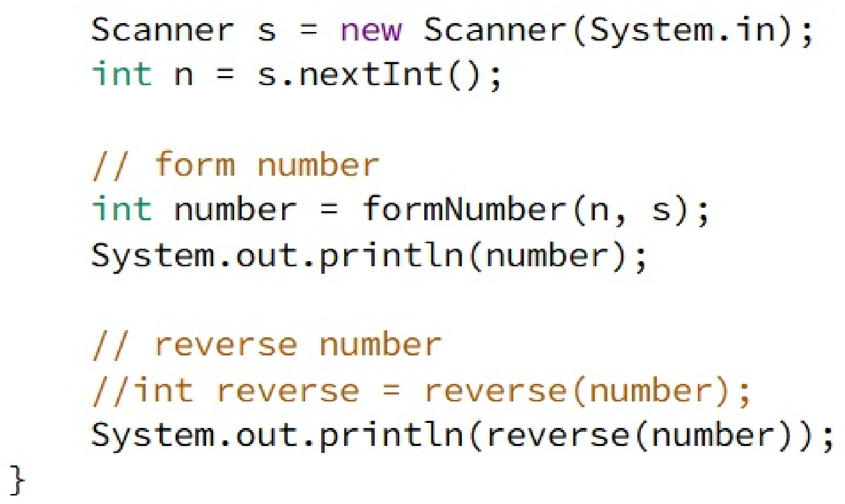
12/10 → 1

1/10 → 0

321

return

① Main

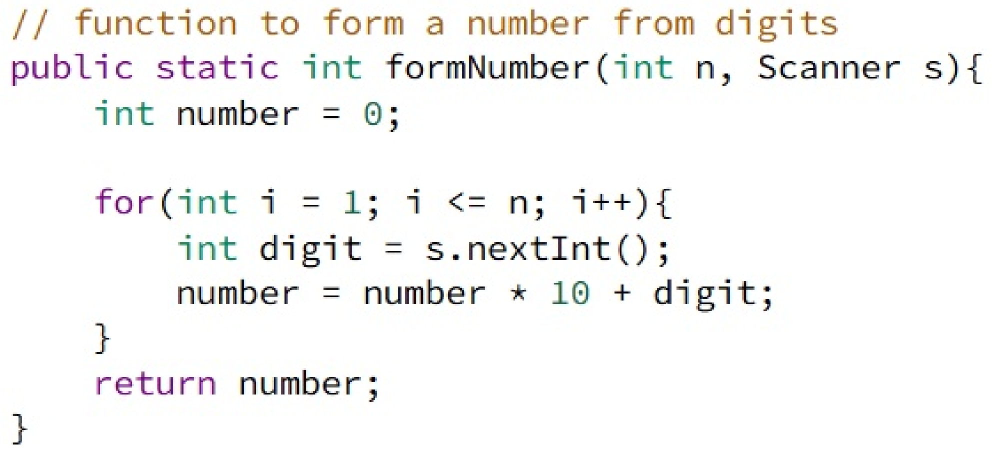


```java
        Scanner s = new Scanner(System.in);
        int n = s.nextInt();

        // form number
        int number = formNumber(n, s);
        System.out.println(number);

        // reverse number
        //int reverse = reverse(number);
        System.out.println(reverse(number));
    }
```

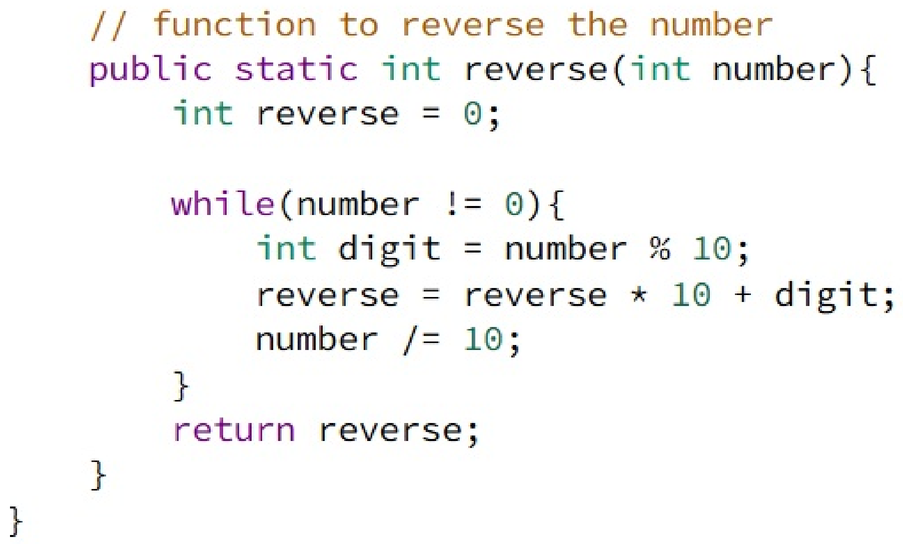
②



```java
    // function to form a number from digits
    public static int formNumber(int n, Scanner s){
        int number = 0;

        for(int i = 1; i <= n; i++){
            int digit = s.nextInt();
            number = number * 10 + digit;
        }
        return number;
    }
```

③



```java
    // function to reverse the number
    public static int reverse(int number){
        int reverse = 0;

        while(number != 0){
            int digit = number % 10;
            reverse = reverse * 10 + digit;
            number /= 10;
        }
        return reverse;
    }
}
```

Armstrong number

int n = 153 ✓   digit = 3

ans = $(1)^3 + (5)^3 + (3)^3$

= 1 + 125 + 27

= 153 ✓

Yes, this is armstrong

n = 1634   digits = 4

$\Rightarrow (1)^4 + (6)^4 + (3)^4 + (4)^4$

= 1 + 196 + 81 + 256

= 1634   Yes.

n = 121   d = 3

$\Rightarrow (1)^3 + (2)^3 + (1)^3$

= 1 + 8 + 1

= 10   NO, not armstrong.

$n = 153$

① int rem = n%10

rem = $\underline{3}$

rem = $\underline{5}$

rem = 1

② int ans = 0

ans = $(3)^3 = \underline{3}\times3\times3$

ans = $(3)^3 + (5)^3 = \underline{5}\times5\times5$

ans = $(3)^3 + (5)^3 + (1)^3$

③ n = n/10

n = 15 (n>0)

n = 1 (n>0)

n = 0 (n>0) ✗ F

$n = 153$

ans = $(3)^3 + (5)^3 + (1)^3$

if (n == ans) return true

else return false

ans = ans + (rem × rem × rem) ✓

$0 + 3\times3\times3 = 27 + (5\times5\times5)$

```
boolean ans = func name ( n ) ;          ||boolean fun.
    Syso(ans)                                  calling .

Syso (fun name(n));
```