print a to z

series:- 'a', 'b', 'c', ------, 'z'

```
for ( char i = 'a'; i <= 'z'; i++) {
        Syso ( i);
}
```

Note:- compiler understand only number's

```
public static void main(String[] args) {
    for (char i = 'a'; i <= 'z'; i++) {
        System.out.println(i);
    }
}
```

Print a, c, e... till the characters are less than z

```
public static void main(String[] args) {
    for (char i = 'a'; i <= 'z'; i += 2) {
        System.out.println(i);
    }
}
```

Print z, y, x.... till 26 characters

one liner :- from 'z' to 'a' by -1

```
public static void main(String[] args) {
    for (char i = 'z'; i >= 'a'; i--) {
        System.out.println(i);
    }
}
```

Print a, B, c, D, e, F, g..... 26 characters

series :- a , b , c , d , - - - - - - - , z

idx ⟶ 0 1 2 3 ... 25

modify the series so that every char. at odd indexes will become capital

```
public static void main(String[] args) {
    int idx = 0;
    for (char i = 'a'; i <= 'z'; i++) {
        if ( idx % 2 != 0 ) {
            System.out.println( (char)(i - 32) );
        } else {
            System.out.println(i);
        }
        idx++;
    }
}
```

print, aBcDeFg - - - - - - - so on.

dry run

idx = 0 1 2 3 4 5 6

i = 'a', (0%2 != 0) ✗
i = 'b', (1%2 != 0) ✓
i = 'c', (2%2 != 0) ✗
i = 'd', (3%2 != 0) ✓
i = 'e', (4%2 != 0) ✗
i = 'f', (5%2 != 0) ✓
i = 'g', (6%2 != 0) ✗

Note:- for(initialization ; cond ; upgradation)

Ex:-

```
for ( int i=0 ;     ; i++ ){
    print (i);
}
```

Ex:-

```
for (    ;    ;    ){
    print("Hi");
}
```

(also a valid loop because all 3 values are optional)

⇒ String str = "Rahul";
0 1 2 3 4

- str.length() // 5
- str.charAt(3) // 'u'
- str.charAt(5) // String Index Out Of Bound

Print Alternate Elements of a String

i/p) String str = "Geekster"; i=
0 1 2 3 4 5 6 7

o/p) ans = "Gese";

Code

Str = "abcd"
0 1 2 3

len = 4

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    String str = scn.nextLine();

    int n = str.length();
    for (int i = 0; i < n; i += 2) {
        char ch = str.charAt(i);
        System.out.print(ch);
    }
}
```

Reverse The String

str = "geekster"

ans = "retskeeg"

Note:-
- can't traverse in reverse
- can't use inbuilt fn

Approach

String ans = "";

ans = "G";

ans = "eG";

ans = "eeG";

ans = "keeG";

ans = "skeeG";

ans = "tskeeG";

ans = "etskeeG";

ans = "retskeeG";

str = "Geekster";

0 1 2 3 4 5 6 7

i

ans = "eeG"

ch = 'k'

ans = ch + ans;

code

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    String str = scn.nextLine();

    int n = str.length();
    String ans = "";
    for (int i = 0; i < n; i++) {
        char ch = str.charAt(i);
        ans = ch + ans;
    }
    System.out.println(ans);
}
```

nth power of 2

$n = 3$, $ans = 8 = 2^3$

$n = 4$, $ans = 16 = 2^4$

n $\quad ans = 2^n = \underbrace{2 * 2 * 2 * \cdots}_{n \text{ times}}$

Psudo code

1) take input n
2) create ans = 1
3) run the loop n no. of times
   3.1) ans = ans * 2
4) print ans

Code

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int ans = 1;
    for (int i = 0; i < n; i++) { // n times
        ans = ans * 2;
    }
    System.out.println(ans);
}
```

dry run

n = 5

ans = 1

i=0, (0<5)✓ ans = 1*2 = 2

i=1, (1<5)✓ ans = 2*2 = 4

i=2, (2<5)✓ ans = 4*2 = 8

i=3, (3<5)✓ ans = 8*2 = 16

i=4, (4<5)✓ ans = 16*2 = 32

i=5, (5<5)✗