BHAILANG

A toy programming language written in Typescript

`npm i -g bhailang`

Playground   **View Source**

Made by **@aniketsingh0104** and **@ristri**

Playground   Run   **Clear**

```
1
2    hi bhai
3      bol bhai "Hello World";
4
5        bhai ye hai a = 3;
6        bhai ye hai b = 0;
```

```
 7
 8          jab tak bhai (b < 5) {
 9            bol bhai b;
10
11            agar bhai (b == a) {
12              bol bhai "b is equal to a";
13            } nahi to bhai (b == 0) {
14              bol bhai "b is equal to zero";
15            }
16
17            b += 1;
18          }
19
20        bye bhai
21
```

Documentation

Bhailang is dynamically typed toy programming language, based on an inside joke, written in Typescript.

General

`hi bhai` is the entrypoint for the program and all program must end with `bye bhai`. Anything outside of it will be ignored.

Variables

Variables can be declared using `bhai ye hai`.

```
This will be ignored

hi bhai
    // Write code here
bye bhai

This too
```

```
hi bhai
    bhai ye hai a = 10;
    bhai ye hai b = "two";
    bhai ye hai c = 15;
    a = a + 1;
    b = 21;
    c *= 2;
bye bhai
```

Types

Numbers and strings are like other languages. Null values can be denoted using `nalla`. `sahi` and `galat` are the boolean values.

```
hi bhai
    bhai ye hai a = 10;
    bhai ye hai b = 10 +
(15*20);
    bhai ye hai c = "two";
    bhai ye hai d = 'ok';
    bhai ye hai e = nalla;
    bhai ye hai f = sahi;
    bhai ye hai g = galat;
bye bhai
```

Built-ins

Use `bol bhai` to print anything to console.

```
hi bhai
    bol bhai "Hello Wor"
    bhai ye hai a = 10;
    {
        bhai ye hai b = 20;
        bol bhai a + b;
    }
    bol bhai 5, 'ok', nalla ,
sahi , galat;
bye bhai
```

Conditionals

Loops

Bhailang supports if-else-if ladder construct , `agar bhai` block will execute if condition is `sahi`, otherwise one of the subsequently added `nahi to bhai` blocks will execute if their respective condition is `sahi`, and the `warna bhai` block will eventually execute if all of the above conditions are `galat`.

Statements inside `jab tak bhai` blocks are executed as long as a specified condition evaluates to `sahi`. If the condition becomes `galat`, statement within the loop stops executing and control passes to the statement following the loop. Use `bas kar bhai` to break the loop and `agla dekh bhai` to continue within loop.

```
hi bhai
    bhai ye hai a = 10;
    agar bhai (a < 20) {
      bol bhai "a is less
than 20";
    } nahi to bhai ( a < 25 )
{
      bol bhai "a is less
than 25";
    } warna bhai {
      bol bhai "a is greater
than or equal to 25";
    }
bye bhai
```

```
hi bhai
    bhai ye hai a = 0;
    jab tak bhai (a < 10) {
      a += 1;
      agar bhai (a == 5) {
        bol bhai "andar se
bol bhai ", a;
        agla dekh bhai;
      }
      agar bhai (a == 6) {
        bas kar bhai;
      }
      bol bhai a;
    }
    bol bhai "done";
bye bhai
```