 hacktiv8 / **phase-0-activities**

Watch40

Star70

Fork174

<> Code

Issues0

Pull requests0

Projects0

Insights

Join GitHub today

GitHub is home to over 28 million developers working together to host and review code, manage projects, and build software together.


Sign up

Dismiss

Branch: master

phase-0-activities / **modules** / **anchor-main-loop.md**

Find fileCopy path

 adhywiranata update w2 modules7b0ba53 on Jan 28

1 contributor

151 lines (128 sloc) | 4.14 KB

Raw

Blame

History

Looping

Soal looping ini memiliki 3 nomor.

1. Melakukan *Looping* Menggunakan **While**

Problem

Pada tugas ini kamu diminta untuk melakukan looping dalam JavaScript dengan menggunakan syntax while. Untuk membuat tantangan ini lebih menarik, kamu juga diminta untuk membuat suatu looping yang menghitung maju dan menghitung mundur. Jangan lupa tampilkan di console juga judul 'LOOPING PERTAMA' dan 'LOOPING KEDUA'.

Output

```
LOOPING PERTAMA
2 - I love coding
4 - I love coding
6 - I love coding
8 - I love coding
10 - I love coding
12 - I love coding
14 - I love coding
16 - I love coding
18 - I love coding
20 - I love coding
LOOPING KEDUA
20 - I will become fullstack developer
18 - I will become fullstack developer
16 - I will become fullstack developer
14 - I will become fullstack developer
12 - I will become fullstack developer
10 - I will become fullstack developer
8 - I will become fullstack developer
6 - I will become fullstack developer
4 - I will become fullstack developer
2 - I will become fullstack developer
```

2. Melakukan *Looping* Menggunakan **For**

Problem

Pada tugas ini kamu diminta untuk melakukan *looping* dalam JavaScript dengan menggunakan syntax `for` . Untuk membuat tantangan ini lebih menarik, kamu juga diminta untuk membuat suatu *looping* yang menghitung maju dan menghitung mundur. Jangan lupa tampilkan di console juga judul 'LOOPING PERTAMA' dan 'LOOPING KEDUA'.

Hints

Operator `++` disebut dengan operator *Increment* Operator `--` disebut dengan operator *Decrement*

Output

```
LOOPING PERTAMA
1 - I love coding
2 - I love coding
3 - I love coding
4 - I love coding
5 - I love coding
6 - I love coding
7 - I love coding
8 - I love coding
9 - I love coding
10 - I love coding
11 - I love coding
12 - I love coding
13 - I love coding
14 - I love coding
15 - I love coding
16 - I love coding
17 - I love coding
18 - I love coding
19 - I love coding
20 - I love coding
LOOPING KEDUA
20 - I will become fullstack developer
19 - I will become fullstack developer
18 - I will become fullstack developer
17 - I will become fullstack developer
16 - I will become fullstack developer
15 - I will become fullstack developer
14 - I will become fullstack developer
13 - I will become fullstack developer
12 - I will become fullstack developer
11 - I will become fullstack developer
10 - I will become fullstack developer
9 - I will become fullstack developer
8 - I will become fullstack developer
7 - I will become fullstack developer
6 - I will become fullstack developer
5 - I will become fullstack developer
4 - I will become fullstack developer
3 - I will become fullstack developer
2 - I will become fullstack developer
1 - I will become fullstack developer
```

3. Angka Ganjil dan Genap

Hint: kamu akan menggunakan kondisional juga di kasus ini.

Problem

- Buatlah sebuah perulangan 1 - 100 dengan penambahan *counter* sebanyak 1
- Di dalam perulangan, periksa setiap angka *counter*:
 - Apabila angka *counter* adalah angka genap, tuliskan **GENAP**
 - Apabila angka *counter* adalah angka ganjil, tuliskan **GANJIL**
- Buatlah 3 perulangan baru dari 1 - 100, dengan penambahan *counter* sebesar 2, 5, dan 9.
- Pada 3 perulangan baru ini periksa setiap angka *counter*:
 - Apabila bukan kelipatan yang ditentukan tidak perlu menuliskan apa-apa
 - Apabila angka *counter* adalah kelipatan 3 dengan penambahan 2, kelipatan 6 dengan penambahan 5, dan kelipatan 10 dengan penambahan 9, tuliskan:
 - "3 kelipatan 3" dan seterusnya.

output

```
//contoh - ganjil genap
//counter sekarang = 1,
//output
"GANJIL"
//counter sekarang = 2,
//output
"GENAP"
// dan seterusnya :)

//contoh - untuk penambahan counter 2
//counter sekarang = 1,
//output
""
//counter sekarang = 3,
//output
"3 KELIPATAN 3"
// dan seterusnya :)

//contoh - untuk penambahan counter 5
//counter sekarang = 1,
//output
""
//counter sekarang = 6,
//output
"6 KELIPATAN 6"
// dan seterusnya :)

//contoh - untuk penambahan counter 9
//counter sekarang = 1,
//output
""
//counter sekarang = 10,
//output
"10 KELIPATAN 10"
// dan seterusnya :)
```