

Tugas Jurnal Pertemuan 2

1. Soal1.js

```
1  const readline = require("readline");
2
3  const rl = readline.createInterface({
4    input: process.stdin,
5    output: process.stdout,
6  });
7
8  rl.question("Masukkan nama Anda: ", (nama) => {
9    console.log(`Selamat datang, ${nama}!`);
10   rl.close();
11 });
12
```

```
.js
Masukkan nama Anda: ahmad uffi
Selamat datang, ahmad uffi!
```

Kode di atas merupakan program Node.js yang menggunakan modul readline untuk membaca input dari pengguna melalui terminal. Pertama, dibuat antarmuka input/output dengan `readline.createInterface()` yang memungkinkan interaksi pengguna lewat `stdin` dan `stdout`. Kemudian, fungsi `rl.question()` digunakan untuk menampilkan pertanyaan "Masukkan nama Anda: " dan menunggu input dari pengguna. Setelah pengguna memasukkan namanya, program akan menampilkan pesan sapaan "Selamat datang, [nama]!" dengan nama yang dimasukkan, lalu menutup antarmuka input menggunakan `rl.close()` agar program berhenti dengan benar.

2. Soal2.js

```
1  const bilanganGenap = [2, 4, 6, 8, 10];
2
3  bilanganGenap.forEach((angka, index) => {
4    console.log(`Angka genap ${index + 1} : ${angka}`);
5  });
6
```

```

PS E:\ITTP\Modul Ittp\Semester
PS E:\ITTP\Modul Ittp\Semester
.js
0 #$$$
1
2 ##
3 $$
4 ##
5
6 #$$$
7
8 ##
9 $$
10 ##
11
12 #$$$
13
14 ##
15 $$
16 ##
17
18 #$$$
19
20 ##

```

Kode di atas terdiri dari dua kelas dalam JavaScript, yaitu Penjumlahan dan SimpleDatabase. Kelas Penjumlahan memiliki satu metode penjumlahanTigaAngka(a, b, c) yang menjumlahkan tiga angka, namun sebelum menjumlahkan, dicek terlebih dahulu apakah semua parameter bertipe number; jika tidak, maka akan melempar error. Contoh pemanggilan metode ini dilakukan dengan penjumlahan.penjumlahanTigaAngka(23, 11, 10) yang menghasilkan output 44. Sedangkan kelas SimpleDatabase berfungsi menyimpan data-data yang ditambahkan melalui metode addData(data), sekaligus mencatat tanggal dan waktu saat data tersebut dimasukkan menggunakan new Date().toUTCString(). Metode printAllDatas() digunakan untuk menampilkan seluruh data beserta tanggal inputnya. Di akhir kode, objek database menambahkan tiga data dan mencetak semuanya ke konsol.

3. Soal3.js

```
1  const readline = require("readline");
2
3  const rl = readline.createInterface({
4    input: process.stdin,
5    output: process.stdout,
6  });
7
8  function isPrime(n) {
9    if (n < 2) return false;
10   for (let i = 2; i <= Math.sqrt(n); i++) {
11     if (n % i === 0) return false;
12   }
13   return true;
14 }
15
16 rl.question("Masukkan sebuah angka (1 - 10000): ", (nilaiString) => {
17   let nilaiInt = parseInt(nilaiString);
18
19   if (isNaN(nilaiInt) || nilaiInt < 1 || nilaiInt > 10000) {
20     console.log(
21       "Input tidak valid! Harap masukkan angka antara 1 hingga 10000."
22     );
23   } else {
24     if (isPrime(nilaiInt)) {
25       console.log(`Angka ${nilaiInt} merupakan bilangan prima`);
26     } else {
27       console.log(`Angka ${nilaiInt} bukan merupakan bilangan prima`);
28     }
29   }
30
31   rl.close();
32 });
33
```

```
PS E:\ITTP\Modul Ittp\Semester4\Praktikum
Masukkan sebuah angka (1 - 10000): 34
Angka 34 bukan merupakan bilangan prima
PS E:\ITTP\Modul Ittp\Semester4\Praktikum
```

Kode di atas adalah program Node.js yang digunakan untuk mengecek apakah sebuah angka termasuk bilangan prima atau tidak, dengan input dari pengguna melalui terminal. Pertama, modul readline digunakan untuk membuat antarmuka input/output. Kemudian didefinisikan fungsi `isPrime(n)` yang mengembalikan `true` jika angka tersebut adalah bilangan prima dan `false` jika tidak, dengan logika pemeriksaan hingga akar kuadrat dari `n`. Setelah pengguna memasukkan angka melalui `rl.question`, input tersebut dikonversi ke bilangan bulat menggunakan `parseInt`. Program akan memvalidasi bahwa input merupakan angka antara 1 hingga 10000; jika tidak valid, ditampilkan pesan kesalahan. Jika valid, program akan mengecek apakah angka tersebut bilangan prima dan mencetak hasilnya, lalu menutup interface readline dengan `rl.close()`.