

!!Catatan: Diharapkan seluruh pengerjaan Assignment tidak sepenuhnya mengandalkan penggunaan AI!!

*"Proses belajar ibarat menanam pohon. Jika hanya mengandalkan AI tanpa memahami esensinya, yang berkembang bukan kompetensimu, melainkan ketergantungan yang melemahkan."
- Learning Design Dibimbing*

Assignment Guidance: Javascript Control Flow

Full Stack Web Development Bootcamp

Periode Pembelajaran

JavaScript Fundamentals

JavaScript Control Flow

Objectives

1. Student mampu memahami penggunaan pseudocode dan flowchart dalam menyusun algoritma pemrograman.
2. Student mampu memahami dan mengimplementasikan conditional statement (if-else dan switch case) dalam bentuk kode dan flowchart.
3. Student mampu memahami konsep dan fungsi dari looping statement (for loop, while loop, do while loop) beserta perbedaannya.
4. Student mampu mengimplementasikan looping untuk manipulasi data bertipe string dan range angka.
5. Student mampu mengimplementasikan nested loop serta mengombinasikan antara conditional dan looping statement dalam menyelesaikan permasalahan.

Deskripsi Assignment

Pada assignment ini, student diminta untuk mengimplementasikan control flow dalam JavaScript untuk menyelesaikan beberapa studi kasus logika sederhana. Studi kasus tersebut meliputi analisis bilangan dalam range tertentu, perhitungan rata-rata nilai siswa, serta menampilkan hasil perpangkatan angka menggunakan nested looping.

Hasil akhir dari assignment ini berupa kode program JavaScript yang dapat mencetak output sesuai dengan aturan dan format yang telah ditentukan. Assignment ini dirancang untuk mengasah pemahaman konsep dasar control flow seperti perulangan dan percabangan logika dalam konteks dunia nyata.

Detail Assignment

Student harus menyelesaikan semua soal berikut sesuai instruksi:

Soal 1: Analisis Bilangan 1 – 20

- Lakukan looping dari angka 1 hingga 20.
- Untuk setiap angka, berikan kondisi dan cetak output jika angka tersebut:
 - Blok **IF, ELSE, ELSE IF pertama**:
 - Bilangan ganjil: tampilkan "{bilangan} merupakan bilangan Ganjil".
 - Bilangan genap: tampilkan "{bilangan} merupakan bilangan Genap".
 - Blok **IF, ELSE, ELSE IF kedua**:
 - Habis dibagi 4: tampilkan "{bilangan} habis dibagi dengan 4".
 - Habis dibagi 6: tampilkan "{bilangan} habis dibagi dengan 6".
 - Habis dibagi 4 dan 6: tampilkan "{bilangan} habis dibagi dengan angka 4 dan 6".
 - Habis dibagi 3 atau 5: tampilkan "{bilangan} habis dibagi dengan angka 3 atau 5".
 - Jika bilangan tidak habis dibagi 4, 6, 3, dan 5 maka tampilkan bilangan nya saja

Soal 2: Menghitung Nilai Rata-rata dan Grade

- Diketahui nilai siswa (Gabriel) untuk 5 semester: 85, 95.20, 67.75, 78, 75.
- Hitung rata-rata nilai.
- Tentukan grade akhir berdasarkan kriteria berikut:
 - A: 90–100
 - B: 80–89.99
 - C: 75–79.99
 - D: 60–74.99
 - E: < 60
- Cetak nilai rata-rata dan grade akhir siswa.
- **Note:** Gunakan array untuk menyimpan seluruh nilai dari 5 semester kedalam 1 variabel

Soal 3: Perpangkatan Menggunakan Nested Loop

- Buat dua looping: outer loop untuk angka 1–5, dan inner loop untuk pangkat 1–3.
- Tampilkan hasil perpangkatan dengan format "{nilai} pangkat {pangkat} = {hasil}".
- Pisahkan setiap set hasil angka dengan `console.log("=====")`.

Ekspektasi Output:

1. Output mencerminkan pemahaman terhadap conditional dan looping.
2. Format output sesuai instruksi.
3. Tidak ada hardcode hasil, student diharapkan memahami logika dan implementasi program.

Tools

Visual Studio Code, Node.js

Pengumpulan Assignment

Deadline :

Maksimal H+7 Kelas (Pukul 23.30 WIB)

Details :

Dikumpulkan dalam bentuk zip, secara INDIVIDU, di LMS

Indikator Penilaian

No	Aspek Penilaian	Parameter	Bobot Maksimal
1	Conditional Statement	Student mampu membuat kondisi yang benar sesuai dengan rules untuk menampilkan angka genap dan ganjil (kombinasi dengan arithmetic operators)	15
2		Student mampu mengimplementasikan conditional statement untuk membuat kondisi yang benar untuk menampilkan angka yang dapat dibagi dengan 4, 6, 4 DAN 6, serta 3 ATAU 5 (kombinasi dengan logical & comparison operators)	20
3		Student mampu mengimplementasikan arithmetic operators untuk mendapatkan nilai rata-rata, lalu membuat kondisi yang benar untuk menampilkan Grade akhir dari siswa	20
4	Looping Statement	Student mampu mengimplementasikan looping	20

		(baik dengan For Loop / While Loop) untuk melakukan iterasi pada range angka 20 pada soal 1	
5		Student mampu mengimplementasikan nested looping untuk menampilkan perpangkatan	25

Referensi/Template

1. [Modul Pembelajaran Full Stack Web Development](#), Algoritma Dasar Pemrograman Javascript 2-4 halaman 19 - 32
2. <https://www.geeksforgeeks.org/what-is-pseudocode-a-complete-tutorial/>
3. <https://www.geeksforgeeks.org/what-is-a-flowchart-and-its-types/>
4. https://www.w3schools.com/js/js_if_else.asp
5. https://www.w3schools.com/js/js_loop_for.asp
6. <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/while>
7. <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/do...while>
8. <https://www.petanikode.com/javascript-perulangan/>
9. <https://www.petanikode.com/javascript-percabangan/>

Sanksi Penggunaan AI:

Apabila student terdeteksi 100% menggunakan AI, maka hasil assignment akan diberikan skor 0.

Ketentuan Pencapaian Nilai:

Nilai minimum Lulus Penyaluran Kerja: 75

Nilai minimum Lulus Bootcamp: 65

Ketentuan Penilaian:

Mengumpulkan Assignment tepat waktu: Sesuai dengan nilai yang diberikan mentor

Mengumpulkan Assignment 12 jam setelah deadline: - 3 dari nilai yang diberikan mentor

Mengumpulkan Assignment 1 x 24 Jam setelah deadline: - 6 dari nilai yang diberikan mentor

Mengumpulkan Assignment 2 x 24 Jam setelah deadline: - 12 dari nilai yang diberikan mentor

Mengumpulkan Assignment 3 x 24 Jam setelah deadline: - 18 dari nilai yang diberikan mentor

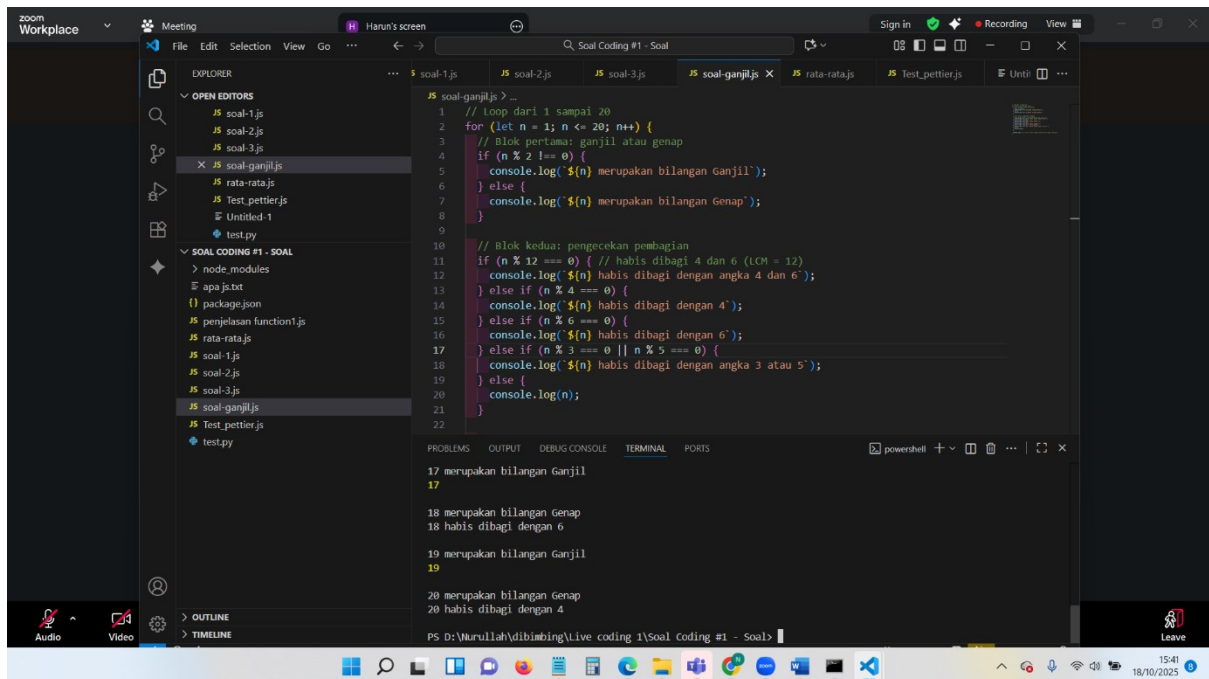
Mengumpulkan Assignment 4 x 24 Jam setelah deadline: - 24 dari nilai yang diberikan mentor

Mengumpulkan Assignment 5 x 24 Jam setelah deadline: - 30 dari nilai yang diberikan mentor

Mengumpulkan Assignment 6 x 24 Jam setelah deadline: - 36 dari nilai yang diberikan mentor

Mengumpulkan Assignment 7 x 24 Jam setelah deadline: - 42 dari nilai yang diberikan mentor

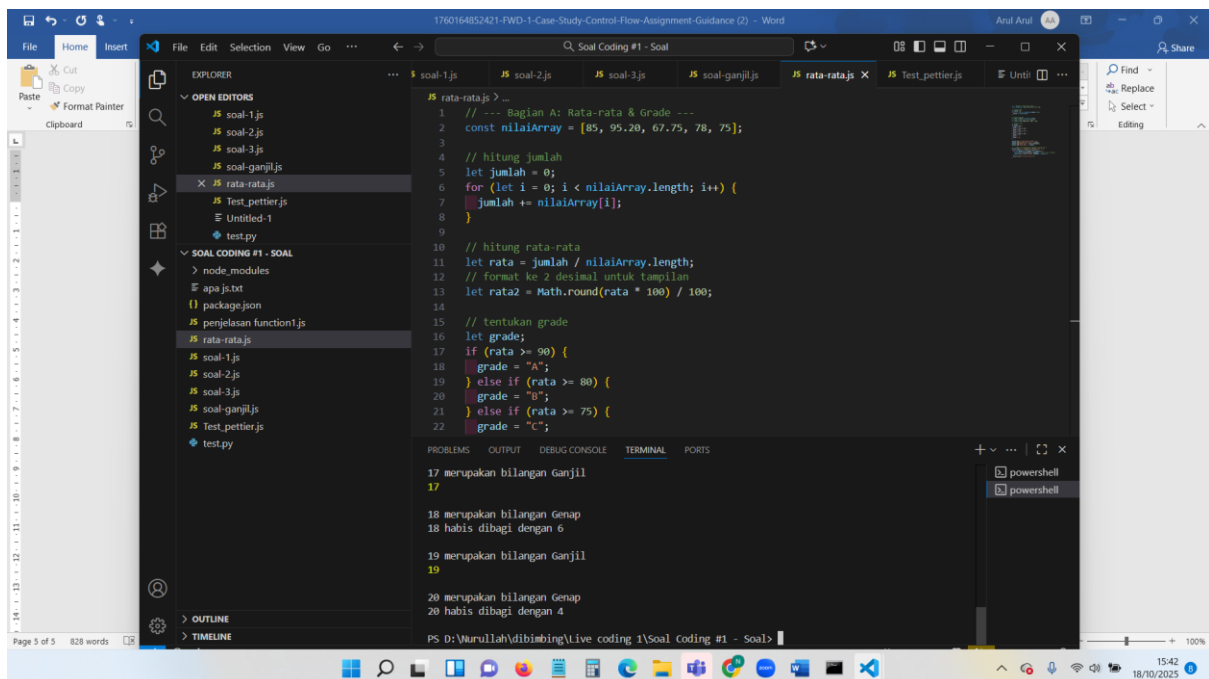
Jawaban Soal 1



```
1 // Loop dari 1 sampai 20
2 for (let n = 1; n <= 20; n++) {
3   // Blok pertama: ganjil atau genap
4   if (n % 2 !== 0) {
5     console.log(`${n} merupakan bilangan Ganjil`);
6   } else {
7     console.log(`${n} merupakan bilangan Genap`);
8   }
9
10  // Blok kedua: pengecekan pembagian
11  if (n % 12 === 0) { // habis dibagi 4 dan 6 (LCM = 12)
12    console.log(`${n} habis dibagi dengan angka 4 dan 6`);
13  } else if (n % 4 === 0) {
14    console.log(`${n} habis dibagi dengan 4`);
15  } else if (n % 6 === 0) {
16    console.log(`${n} habis dibagi dengan 6`);
17  } else if (n % 3 === 0 || n % 5 === 0) {
18    console.log(`${n} habis dibagi dengan angka 3 atau 5`);
19  } else {
20    console.log(n);
21  }
22 }
```

17 merupakan bilangan Ganjil
17
18 merupakan bilangan Genap
18 habis dibagi dengan 6
19 merupakan bilangan Ganjil
19
20 merupakan bilangan Genap
20 habis dibagi dengan 4

Soal no. 2



```
1 // --- Bagian A: Rata-rata & Grade ---
2 const nilaiArray = [85, 95, 20, 67.75, 78, 75];
3
4 // hitung jumlah
5 let jumlah = 0;
6 for (let i = 0; i < nilaiArray.length; i++) {
7   jumlah += nilaiArray[i];
8 }
9
10 // hitung rata-rata
11 let rata = jumlah / nilaiArray.length;
12 // format ke 2 desimal untuk tampilan
13 let rata2 = Math.round(rata * 100) / 100;
14
15 // tentukan grade
16 let grade;
17 if (rata >= 90) {
18   grade = "A";
19 } else if (rata >= 80) {
20   grade = "B";
21 } else if (rata >= 75) {
22   grade = "C";
23 }
```

17 merupakan bilangan Ganjil
17
18 merupakan bilangan Genap
18 habis dibagi dengan 6
19 merupakan bilangan Ganjil
19
20 merupakan bilangan Genap
20 habis dibagi dengan 4

Soal no 3.

V

The screenshot shows the Visual Studio Code interface with a dark theme. The Explorer sidebar on the left displays a project structure with files like `soal-1.js`, `soal-2.js`, `soal-3.js`, `soal-ganjil.js`, `rata-rata.js`, `pangkat.js`, `test_pettier.js`, `Untitled-1`, and `test.py`. The `SOAL CODING #1 - SOAL` folder is expanded, showing sub-files like `node_modules`, `apa.js.txt`, `package.json`, `pangkat.js`, `penjelasan function1.js`, `rata-rata.js`, `soal-1.js`, `soal-2.js`, `soal-3.js`, `soal-ganjil.js`, `Test_pettier.js`, and `test.py`.

The main editor window shows the content of `pangkat.js`:

```
1 pangkat.js > ...
2 for (let nilai = 1; nilai <= 5; nilai++) {
3   for (let pangkat = 1; pangkat <= 3; pangkat++) {
4     let hasil = nilai ** pangkat; // operator ** artinya pangkat
5     console.log(`${nilai} pangkat ${pangkat} = ${hasil}`);
6   }
7   console.log("=====");
8 }
```

The terminal at the bottom shows the output of the script:

```
3 pangkat 1 = 3
3 pangkat 2 = 9
3 pangkat 3 = 27
=====
4 pangkat 1 = 4
4 pangkat 2 = 16
4 pangkat 3 = 64
=====
5 pangkat 1 = 5
5 pangkat 2 = 25
5 pangkat 3 = 125
=====
```

The terminal prompt is `PS D:\Muhammad\dimbing\Live coding 1\Soal Coding #1 - Soal>`. The status bar at the bottom indicates the file is `pangkat.js` at line 8, column 1, with 4 spaces, UTF-8 encoding, CRLF line endings, and the JavaScript language mode. The Prettier extension is also active.