

LOGBOOK SUPLEMEN PRAKTIKUM — DASAR PEMROGRAMAN JAVASCRIPT

Mata Kuliah: Aplikasi Web dan Mobile | Program Studi Teknik Industri | Universitas Negeri Yogyakarta
Persiapan Pertemuan 4 | Semester Genap 2025/2026

SUPLEMEN PRAKTIKUM · TEKNIK INDUSTRI UNY

Logbook Dasar Pemrograman JavaScript

Identitas Mahasiswa

NAMA LENGKAP

Nanik Wijayanti

NIM

23051430006

KELAS / ROMBEL

Teknik Industri-A 2023

TANGGAL Pengerjaan

27/02/2026



DOSEN PENGAMPU

Dr. Eng. Ir. Aji Ery Burhandenny, S.T., M.AIT.

BAGIAN A

Checklist Materi Suplemen

Centang setiap topik setelah Anda membaca, memahami, dan mencoba contoh kodenya di Browser Console.



Bagian 1 & 2 — Mengapa JavaScript & cara kerja di browser

Konsep

Saya memahami peran JS dalam tiga pilar web dan dapat membuka Browser Console (F12)



Bagian 3 — Variabel: let, const, dan aturan penamaan

Praktik

Saya sudah mencoba mendeklarasikan variabel dan memahami kapan memakai let vs const



Bagian 4 — Tipe Data: Number, String, Boolean

Praktik

Saya memahami jebakan String + Number dan cara konversi dengan Number()



Bagian 5 — Operator: Aritmatika, Perbandingan (===), Logika (&&, ||)

Praktik

Saya sudah mencoba minimal satu perhitungan dan satu ekspresi perbandingan



**Bagian 6 — Control Flow: if, else if, switch, ternary**

Praktik

Saya memahami urutan else if dan kenapa break diperlukan di switch**Bagian 7 — Studi Kasus Kalkulator OEE (membaca dan memahami seluruh kode)**

Analisis

Saya dapat menjelaskan setiap variabel dan setiap blok if dalam kode OEE tersebut**Bagian 9 — Membaca seluruh daftar Kesalahan Umum Pemula**

Review

Saya sudah mengidentifikasi minimal dua kesalahan yang pernah atau mungkin saya lakukan

BAGIAN A2

Bukti Pengerjaan Latihan Mandiri

Untuk setiap latihan: (1) centang setelah selesai, (2) jawab pertanyaan uji pemahaman singkat, dan (3) unggah screenshot output Console sebagai bukti pengerjaan yang akan tercetak di PDF.

Level 1 — Latihan Dasar

**Latihan 1.1 — Variabel Profil Mesin**

Level 1

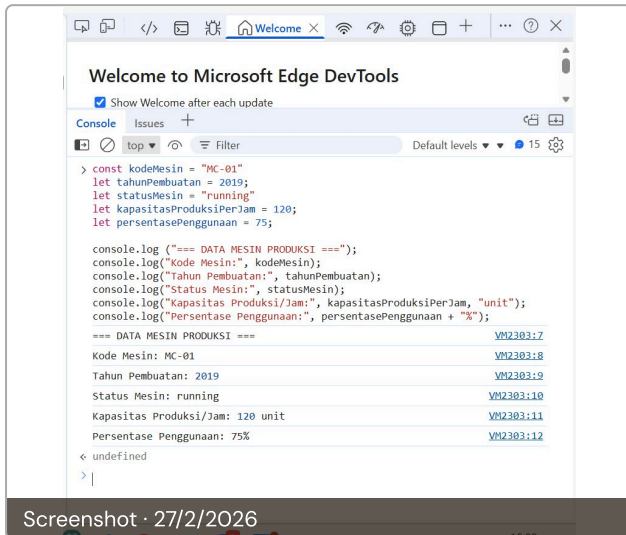
Membuat variabel dengan tipe tepat (let/const) dan menampilkan semua dengan console.log()

UJI PEMAHAMAN SINGKAT

Mengapa variabel "kode mesin" sebaiknya dideklarasikan dengan `const` bukan `let`? Jelaskan singkat.

Kode mesin dideklarasikan dengan `const` karena itu merupakan nilai tetap, sedangkan `let` adalah nilai yang berubah. Sehingga kode mesin sendiri itu merupakan nilai yang tetap dan tidak berubah.

SCREENSHOT OUTPUT CONSOLE



Latihan 1.2 — Kalkulator Biaya Material

Level 1

Menghitung biaya per unit, total biaya, dan berat total material; tampilkan di console

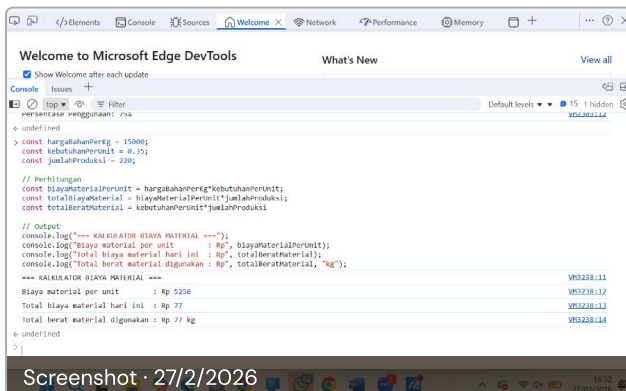


UJI PEMAHAMAN SINGKAT

Jika jumlahProduksi diubah menjadi 0, apa yang terjadi pada biayaPerUnit? Apa masalah matematis yang muncul?

Jika jumlahProduksi diubah menjadi 0, maka biayaPerUnit tidak dapat dihitung. Biasanya biayaPerUnit dihitung dengan rumus = total biaya/jumlah produksi, jika jumlah produksi 0, maka hasil pembagian tersebut menjadi tidak terdefinisi (undefined). Hal ini dapat menyebabkan pemrograman error.

SCREENSHOT OUTPUT CONSOLE



Level 2 — Latihan Menengah



Latihan 2.1 — Sistem Klasifikasi Reject

Level 2

Logika if/else if/else menentukan kategori (Excellent/Acceptable/Warning/Critical) dari reject rate



UJI PEMAHAMAN SINGKAT

Jika Anda membalik urutan kondisi (cek ≥ 5 dulu, baru ≥ 3), apakah hasilnya berbeda? Jelaskan mengapa urutan kondisi `else if` sangat penting.

Jika saya membalik urutan kondisi nya maka hasilnya akan salah. Jika urutannya dibalik

cek ≥ 3

baru cek ≥ 5

5 ≥ 3 (benar)

Program nanti akan berhenti, hasil kategorinya 3, padahal nilainya adalah 5. Hal ini menjadi masalah karena nilai yang lebih besar juga masuk ke syarat yang lebih kecil. Jadi kalau yg kecil dicek dulu, yang besar akan ketutup.

Oleh karena itu urutan kondisi `else if` perlu diperhatikan dan sangat penting. Program akan membaca barisan yang paling atas dulu, jika kondisi tersebut sudah benar, program akan berhenti dan kondisi dibawahnya tidak akan dicek. Akibatnya kalau kondisi yang umum/kecil taruh diatas maka kondisi besar/khusus bisa tidak dipakai dan hasil program bisa jadi keliru /salah.

SCREENSHOT OUTPUT CONSOLE — UJI MINIMAL 2 SKENARIO

```

Welcome to Microsoft Edge DevTools

> const totalProduksi = 1000;
const jumlahReject = 30;
const rejectRate = (jumlahReject / totalProduksi) * 100;
let kategori;
let tindakan;

if (rejectRate < 3) {
  kategori = "Excellent";
  tindakan = "Tidak ada tindakan";
}
else if (rejectRate >= 3 && rejectRate < 5) {
  kategori = "Acceptable";
  tindakan = "Perbaikan standar";
}
else if (rejectRate >= 5 && rejectRate < 8) {
  kategori = "Warning";
  tindakan = "Investigasi Proses";
}
else {
  kategori = "Critical";
  tindakan = "Hentikan Produksi, Lakukan RCA";
}

console.log("===== SKENARIO 1 USE EXCELLENT & ACCEPTABLE =====");
console.log("Reject Rate:", rejectRate.toFixed(2) + "%");
console.log("Kategori:", kategori);
console.log("Tindakan:", tindakan);

===== SKENARIO 1 USE EXCELLENT & ACCEPTABLE =====
Reject Rate: 3
Kategori: Excellent
Tindakan: Tidak ada tindakan
  
```

Screenshot · 27/2/2026

```

Welcome to Microsoft Edge DevTools

> const totalProduksi = 1000;
const jumlahReject = 50;
const rejectRate = (jumlahReject / totalProduksi) * 100;
let kategori;
let tindakan;

if (rejectRate < 3) {
  kategori = "Excellent";
  tindakan = "Tidak ada tindakan";
}
else if (rejectRate >= 3 && rejectRate < 5) {
  kategori = "Acceptable";
  tindakan = "Perbaikan standar";
}
else if (rejectRate >= 5 && rejectRate < 8) {
  kategori = "Warning";
  tindakan = "Investigasi Proses";
}
else {
  kategori = "Critical";
  tindakan = "Hentikan Produksi, Lakukan RCA";
}

console.log("===== SKENARIO 2 USE WARNING & CRITICAL =====");
console.log("Reject Rate:", rejectRate.toFixed(2) + "%");
console.log("Kategori:", kategori);
console.log("Tindakan:", tindakan);

===== SKENARIO 2 USE WARNING & CRITICAL =====
Reject Rate: 5
Kategori: Warning
Tindakan: Investigasi Proses
  
```

Screenshot · 27/2/2026



Latihan 2.2 — Kalkulator Lembur

Level 2

Menghitung total upah lembur berdasarkan jam lembur, dengan tarif 1.5x dan 2x



UJI PEMAHAMAN SINGKAT

Berapa total lembur (Rp) untuk operator dengan gaji pokok Rp 3.500.000 yang lembur 5 jam? Tulis perhitungan manual Anda.

upah per jam = $3.500.000/173$
= Rp 20.231,21

jam 1-3 (tarif 1,5x)

tarif = $1,5 \times 20.231,21 = \text{Rp } 30.346,82/\text{jam}$
 $3 \times 30.346,82 = \text{Rp } 91.040,46$

jam 4-5 (tarif 2x)

tarif = $2 \times 20.231,21 = \text{Rp } 40.462,42/\text{jam}$
 $2 \times 40.462,42 = \text{Rp } 80.924,84$

Total upah lembur = $\text{Rp } 91.040,46 + \text{Rp } 80.924,84$
= $\text{Rp } 171.965,30$

SCREENSHOT OUTPUT CONSOLE — UJI MINIMAL 2 SKENARIO

```

Welcome to Microsoft Edge DevTools
Show Welcome after each update
What's New
View all
What's New in DevTools 115 -
Reject Rate: 4.88% VM6328:27
Kategori: Warning VM6328:28
Status Mesin: Investigasi Proses VM6328:29
< undefined
> const gajiPokok = 3500000;
const jamNormal = 173;
const upahPerJam = gajiPokok / jamNormal;
// INPUT 3JAM LEMBUR (UBAH NILAI INI)
const N = 3;
let totalLembur;
if (N <= 3) {
  // Tarif 1,5x saja
  totalLembur = N * 1.5 * upahPerJam;
  console.log("Tarif lembur: 1,5x");
} else {
  // Tarif campuran
  const jam1x = 3;
  const jam2x = N - 3;
  totalLembur = (jam1x * 1.5 * upahPerJam) +
    (jam2x * 2 * upahPerJam);
  console.log("Tarif lembur: 1,5x + 2x");
}
console.log("Jam lembur:", N, "jam");
console.log("Total upah lembur: Rp", totalLembur.toFixed(2));
Tarif lembur: 1,5x VM6328:13
VM6328:13
VM6328:21
Screenshot · 27/2/2026

```

```

Welcome to Microsoft Edge DevTools
Show Welcome after each update
What's New
View all
What's New in DevTools 115 -
< undefined
> const gajiPokok = 3500000;
const jamNormal = 173;
const upahPerJam = gajiPokok / jamNormal;
const N = 5;
let totalLembur;
if (N > 3) {
  const jam1x = 3;
  const jam2x = N - 3;
  totalLembur = (jam1x * 1.5 * upahPerJam) +
    (jam2x * 2 * upahPerJam);
}
console.log("=== SKENARIO 2: LEMBUR > 3 JAM ===");
console.log("Jam lembur:", N, "jam");
console.log("Tarif lembur: 1,5x + 2x");
console.log("Upah per jam: Rp", upahPerJam.toFixed(2));
console.log("Jam 1-3 (1,5x):", 3, "jam");
console.log("Jam 4-5 (2x):", N - 3, "jam");
console.log("Total upah lembur: Rp", totalLembur.toFixed(2));
=== SKENARIO 2: LEMBUR > 3 JAM === VM6375:17
Jam lembur: 5 jam VM6375:18
Tarif lembur: 1,5x + 2x VM6375:19
Upah per jam: Rp 20231.21 VM6375:20
Jam 1-3 (1,5x): 3 jam VM6375:21
VM6375:22
VM6375:23
Screenshot · 27/2/2026

```

BAGIAN B

Uji Pemahaman Kode

Prediksi output kode berikut *tanpa menjalankannya* terlebih dahulu, lalu klik "Periksa".

SOAL B-1 · OPERATOR & TIPE DATA

```
let a = 10;
let b = "5";
let c = a + Number(b);
let d = a + b;
console.log(c);           // Jawaban 1
console.log(d);           // Jawaban 2
console.log(typeof c);    // Jawaban 3
```

JAWABAN 1 — CONSOLE.LOG(C)

JAWABAN 2 — CONSOLE.LOG(D)

JAWABAN 3 — TYPEOF C

SOAL B-2 · CONTROL FLOW

```
let reject = 8;
let total  = 200;
let rate   = (reject / total) * 100;
if (rate < 1) {
  console.log("Excellent");
} else if (rate < 3) {
  console.log("Acceptable");
} else if (rate < 5) {
  console.log("Warning");
} else {
  console.log("Critical");
}
```

NILAI RATE (%)

OUTPUT DI CONSOLE

SOAL B-3 · SWITCH & LOGIKA

```
let shift = 2;
let isWeekend = true;
let bonus = 0;
switch (shift) {
  case 3: bonus = 50000; break;
  case 2: bonus = 25000; break;
  default: bonus = 0;
}
if (isWeekend && shift === 2) {
  bonus = bonus * 2;
}
console.log(bonus);
```

OUTPUT — CONSOLE.LOG(BONUS)

50000

BAGIAN C

Refleksi Per Topik

Tuliskan refleksi jujur untuk setiap topik. Minimal 40 karakter per jawaban.

C-1

Jelaskan dengan kata-kata Anda sendiri: apa perbedaan let dan const? Berikan satu contoh nyata dari konteks industri untuk masing-masing.

Petunjuk: pikirkan data apa yang berubah vs data apa yang tetap dalam sistem produksi.

Let bisa berubah nilainya sedangkan const nilainya tetap. Contohnya yang let yaitu jumlah produk cacat per shift

(let jumlahReject = 0;)

const contohnya yaitu kode mesin

(const mesin = MC-01;) karena kode mesin merupakan tetap dan tidak bisa berubah

C-2

Mengapa menggunakan `===` lebih aman daripada `==`? Tuliskan contoh kode singkat yang menunjukkan perbedaan perilaku keduanya.

Petunjuk: coba bandingkan angka 0 dengan boolean false menggunakan keduanya di Console.

menggunakan `===` lebih aman karena membandingkan nilai dan tipe data sekaligus, sedangkan `==` hanya membandingkan nilainya saja dan bisa saja ketika menggunakan `==` dapat mengubah tipe data secara otomatis. Hal ini dapat membuat bingung.

C-3

Dari seluruh materi suplemen, konsep mana yang paling sulit Anda pahami? Jelaskan apa yang membuat konsep tersebut sulit dan bagaimana Anda mencoba mengatasinya.

Sebenarnya saya masih cukup bingung dibagian if else nya. saya sudah mulai menangkap tetapi ketika program itu panjang dan harus dipahami. saya seketika malah bingung dan harus membaca berulang kali dan menggunakan bantuan AI untuk menjelaskan bagaimana logika if else itu berjalan

TINGKAT KESULITAN MATERI (PILIH SATU)



Mudah dipahami



Butuh usaha



Cukup menantang



Sangat sulit

C-4

Dari latihan mandiri Bagian 8, pilih satu soal yang sudah Anda kerjakan. Tulis ulang kode solusi Anda dan jelaskan logika yang Anda gunakan.

Petunjuk: salin kode dari VS Code / Console Anda ke sini, lalu jelaskan baris-baris kuncinya.

```
const totalProduksi = 1000;
const jumlahReject = 20;

const rejectRate = (jumlahReject / totalProduksi) * 100;

let kategori;
let tindakan;

if (rejectRate < 1) {
  kategori = "Excellent";
  tindakan = "Tidak Ada Tindakan";
}
else if (rejectRate >= 1 && rejectRate < 3) {
  kategori = "Acceptable";
  tindakan = "Monitor Lebih Ketat";
}
else if (rejectRate >= 3 && rejectRate <= 5) {
  kategori = "Warning";
  tindakan = "Investigasi Proses";
}
else {
  kategori = "Critical";
  tindakan = "Hentikan Produksi, Lakukan RCA";
}

console.log("=== SKENARIO 1 UJI EXCELLENT & ACCEPTABLE ===");
console.log("Reject Rate:", rejectRate.toFixed(2) + "%");
console.log("Kategori:", kategori);
console.log("Status Mesin:", tindakan);
```

Penjelasan:

1. Pertama-tama simpan total unit produksi dan jumlah unit yang reject dengan variabel const, karena nilai tersebut tetap jumlahnya. Ini itu menjadi bahan input ketika mau hitung reject rate nya.
2. Nah lalu yang kedua terdapat rumus menghitung reject ratenya, itu tinggal masukan rumus
3. Lalu selanjutnya, mendeklarasikan variabel untuk kategori dan tindakan. Keduanya saya deklarasikan dengan variabel let, karena hal tersebut bisa berubah nilainya.
4. Selanjutnya melogikakan if else nya, jika reject rate nya kurang dari satu maka termasuk kategori Excellent dan tindakannya tidak ada perbaikan. Jika yg diatas bukan termasuk itu maka turub ke bawahnya,

jika reject rate nya 1–3% maka kategorinya Acceptable dan tindakannya monitoring lebih ketat, dan turun selanjutnya ke bawah sampai menemukan yg jikanya tu mewakili hasil nyatanya.

5. Yang terakhir menampilkan hasil lalu menggunakan toFixed(2) karena untuk membulatkan angka menjadi 2 desimal

Bagian D — Refleksi Akhir & Rencana Belajar

Tulis secara jujur: apa yang paling berkesan dari suplemen ini, dan apa yang akan Anda lakukan sebelum Pertemuan 4 untuk memastikan diri Anda siap?

Saya akan terus belajar dan memahami mengenai semua latihan 8 yang telah dilakukan. Saya akan mendalami semuanya. Memainkan logika nya dengan baik dan memahami cara kepenulisan dengan benar mengenai Java Script ini.

Nanik Wijayanti

23051430006

Diperiksa oleh Dosen Pengampu

Dr. Eng. Ir. Aji Ery Burhandenny, S.T., M.AIT.

27 Februari 2026

Dokumen ini dicetak dari Logbook Digital Suplemen Praktikum — Aplikasi Web dan Mobile, Program Studi Teknik Industri, Universitas Negeri Yogyakarta