**LAPORAN WORKSHOP WEB DASAR**

**JAVASCRIPT BERCABANG**

****

PENULIS : MUHAMMAD FAJAR

NIM : 2023573010093

KELAS : TI-1A

**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI DAN KOMPUTER**

**POLITEKNIK NEGERI LHOKSEUMAWE**

**2024**

# **LEMBAR PENGESAHAN**

JUDUL PRAKTIKUM : Javascript BerCabang

NAMA : Muhammad Fajar

NIM : 2023573010093

KELAS : TI-1A

JURUSAN : Teknologi Informasi dan komputer

PRODI : Teknik Informatika

TANGGAL PRAKTIKUM : 24 APRIL 2024

TANGGAL PENYERAHAN : 30 APRIL 2024

NILAI :

KETERANGAN :

Buket Rata,30 April 2024

Dosen Pembimbing

**Amirullah S.S.T.M.Kom.**

Nip.19890828 2018 031 001

# **DAFTAR ISI**

[**LEMBAR PENGESAHAN** ii](#_Toc165405201)

[**DAFTAR ISI** iii](#_Toc165405202)

[**PENDAHULUAN** 1](#_Toc165405203)

[1. Percabangan if (percabangan\_if.html) 2](#_Toc165405204)

[2. Percabangan if/else (percabangan\_if\_else.html) 3](#_Toc165405205)

[3. Percabangan if/else/if (percabangan\_if\_else\_if.html) 4](#_Toc165405206)

[4. Percabangan switch/case (percabangan\_switch\_case.html) 5](#_Toc165405207)

[5. Percabangan dengan operator ternary (operator\_ternary.html) 6](#_Toc165405208)

[6. Percabangan bersarang (nested.html) 7](#_Toc165405209)

[7. Ternary dalam ternary (nested\_ternary.html) 8](#_Toc165405210)

[**KESIMPULAN** 9](#_Toc165405211)

# 

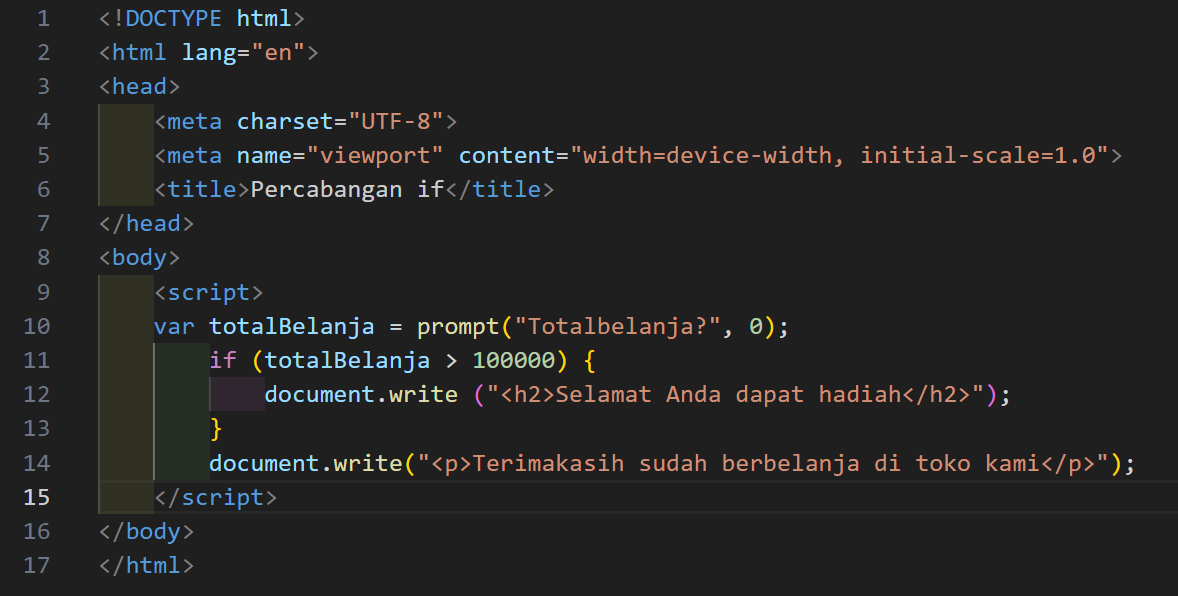
# **PENDAHULUAN**

Percabangan merupakan salah satu konsep dasar dalam pemrograman jacascript yang memungkinkan kita untuk membuat Keputusan berdasarkan kondisi tertentu. Dengan menggunakan percabangan, kita dapat mengontrol alur program sesuai dengan kondisi yang diberikan, sehingga memungkinkan aplikasi untuk melakukan Tindakan yang berbeda tergantung pada situasi yang ada. Dalam javascript, terdapat beberapa cara untuk melakukan percabangan, seperti penggunaan statement if-else, switch-case, dan operator ternary. Memahami konsep percabangan sangat penting bagi seorang pengembang web karena memungkinkan mereka untuk menulis kode yang lebih fleksibel dan responsive.

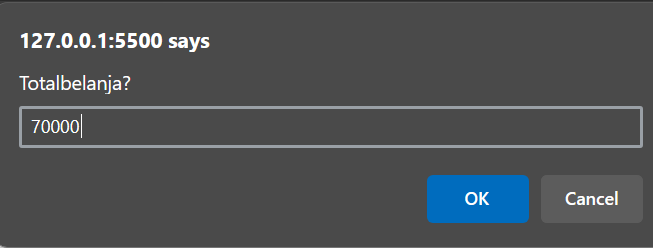
Salah satu jenis percabangan yang umum digunakan adalah statement if-else. Dengan menggunakan if-else, kita dapat mengevaluasi suatu kondisi, dan jika kondisi tersebut benar (true), maka blok kode yang terkait dengan if akan dieksekusi. Namun, jika kondisi tersebut salah (false), maka blok kode dalam else akan dieksekusi. Hal ini memungkinkan aplikasi untuk merespons secara dinamis terhadap masukan pengguna atau perubahan dalam lingkungan eksternal.

Selain statement if-else, JavaScript juga mendukung percabangan menggunakan switch-case. Struktur ini berguna ketika kita memiliki banyak kondisi yang mungkin dan ingin mengeksekusi blok kode tertentu berdasarkan nilai variabel tertentu. Dengan switch-case, kita dapat menentukan serangkaian kasus (case) yang masing-masing berkaitan dengan nilai yang berbeda dari suatu variabel. Jika nilai variabel cocok dengan salah satu kasus yang ditentukan, blok kode terkait akan dieksekusi. Jika tidak ada kasus yang cocok, kita dapat menentukan blok default yang akan dieksekusi. Ini memungkinkan pengembang untuk mengorganisir kode dengan lebih baik, terutama dalam skenario di mana terdapat banyak kemungkinan nilai yang harus dipertimbangkan.

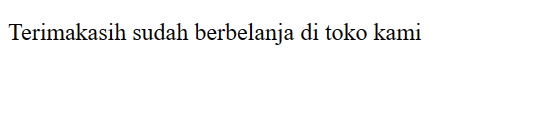
1. Percabangan if (percabangan\_if.html)



* Input



* Output

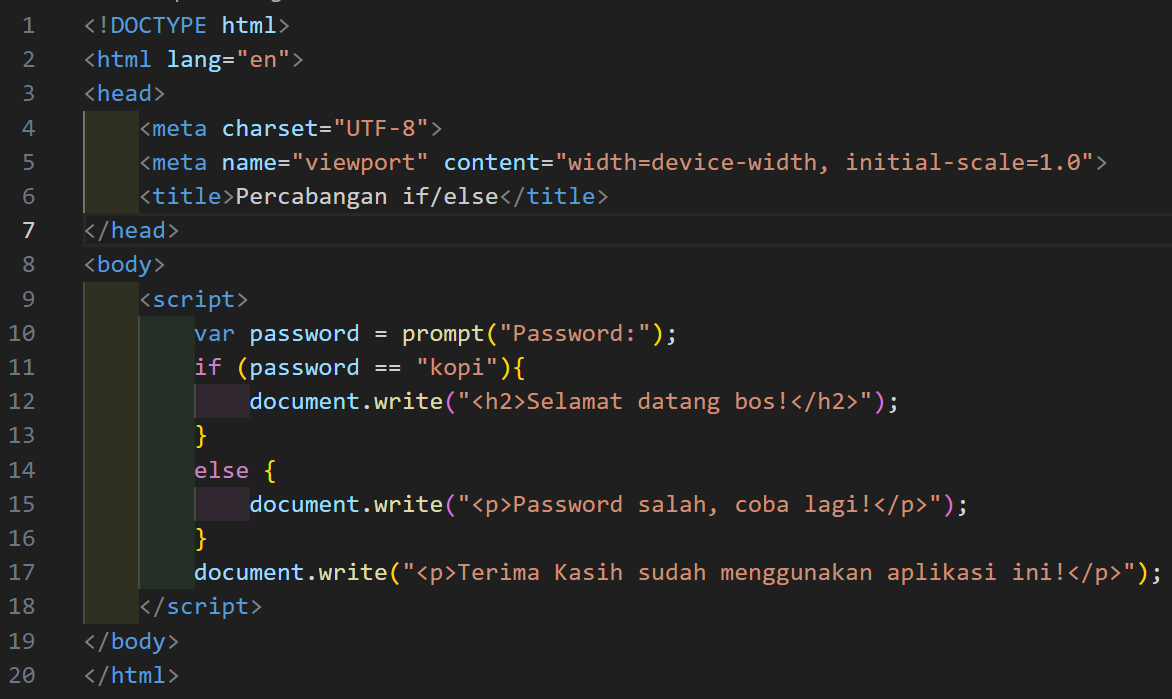


* Analisis

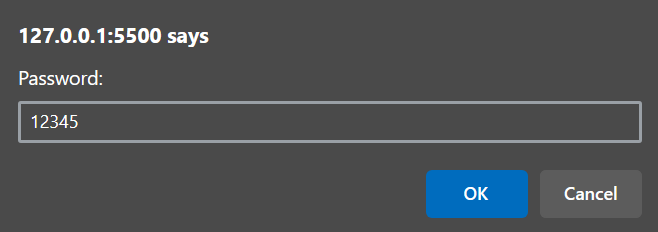
Program ini menggunakan percabangan if .

* **Var totalBelanja :** sebuah variable bernama “totalBelanja”
* **If (totalBelanja > 100000) :** jika variable totalBelanja bernilai lebih dari 10000, maka program akan mencetak kalimat “Selamat Anda dapat hadiah!” dan “Terima kasih sudah berbelanja di toko kami”,  **namun** jika nilai dari variable totalBelanja tidak lebih dari 100000, maka hanya mencetak Kalimat “Terima kasih sudah berbelanja di toko kami”

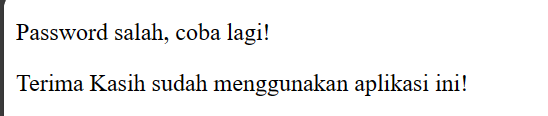
1. Percabangan if/else (percabangan\_if\_else.html)



* Input



* Output



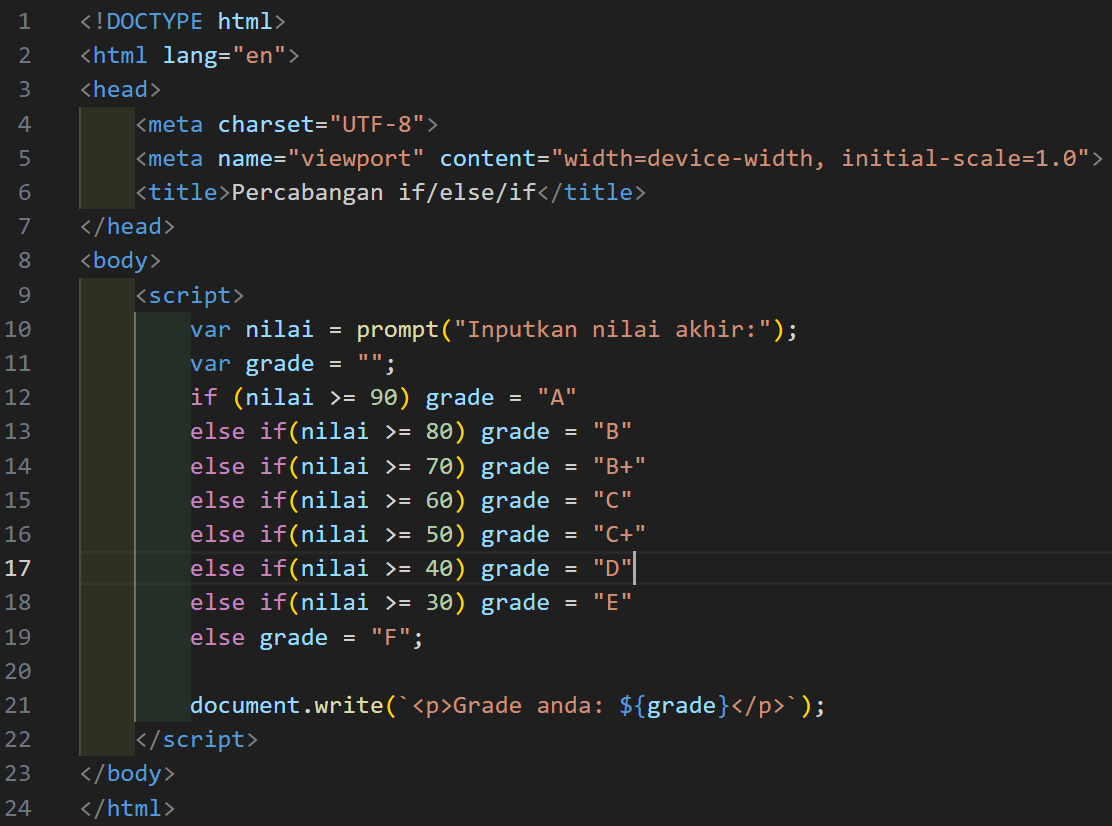
* Analisis

Program ini menggunakan percabangan if / else. Yang Dimana jika true maka program akan menjalankan kondisi if, jika false, maka program akan menjalankan kondisi false.

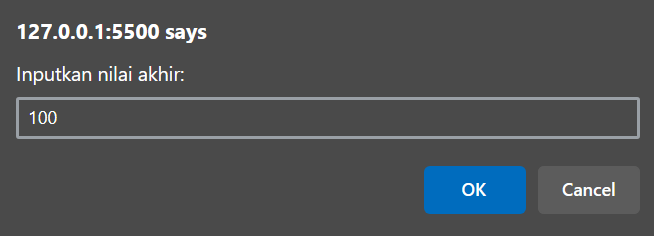
* If (password == “kopi”) : jika nilai dari variable kopi itu adalah kata “kopi” , maka akan menampilkan kalimat “Selamat datang bos!” dan kalimat “Terima kasih sudah menggunakan aplikasi ini!”, namun jika tidak, maka akan menjalankan kondisi else, yang akan menampilkan kalimat “Password salah, coba lagi” dan kalimat “Terima Kasih sudah menggunakan aplikasi ini”

Program ini akan menampilkan kalimat “Terima kasih sudah menggunakan aplikasi ini” walaupun programnya true atau false.

1. Percabangan if/else/if (percabangan\_if\_else\_if.html)



* Input



* Output

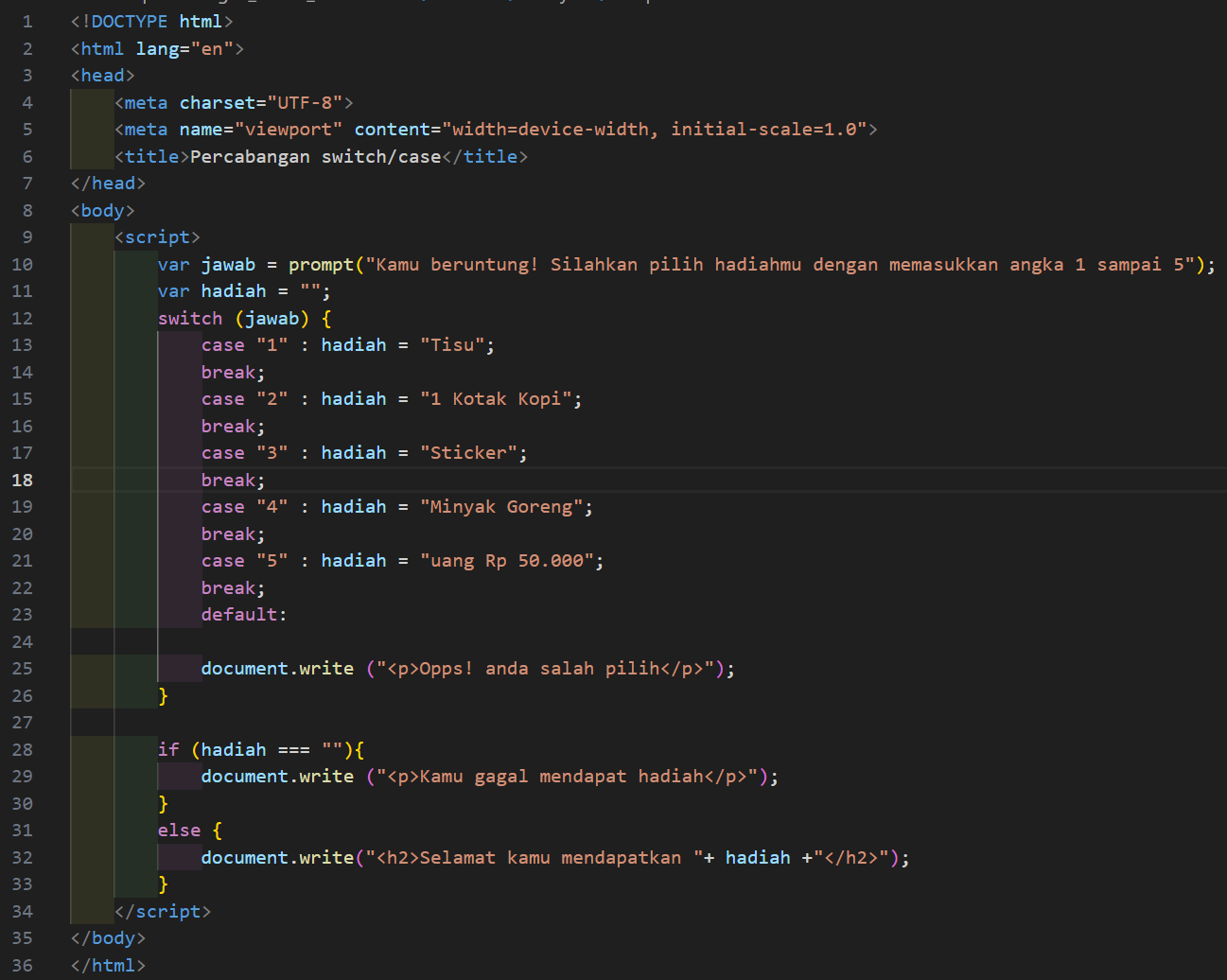


* Analisis

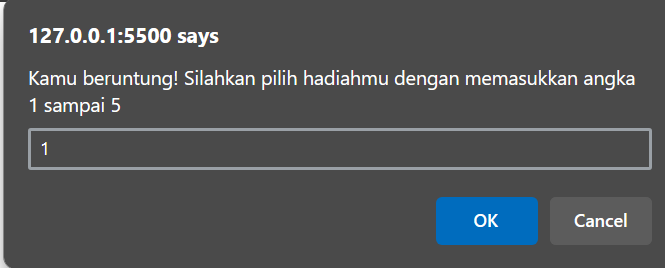
Program ini menggunakan percabangan if / else / if

* **If (nilai >= 90) :** jika variable nilai bernilai 90, maka akan mencetak huruf A, jika nilai dibawah 90, maka akan diperiksa lagi apakah nilai >= 80, jika salah, maka akan diperiksa lagi sampai ke else if (nilai >= 30) yang akan mencetak huruf E, maka jika nilai dibawah 30, maka akan menjalankan kondisi terakhir yaitu kondisi **else** yang akan mencetak huruf F.

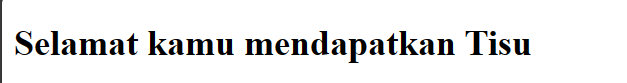
1. Percabangan switch/case (percabangan\_switch\_case.html)



* Input



* Output



* Analisis

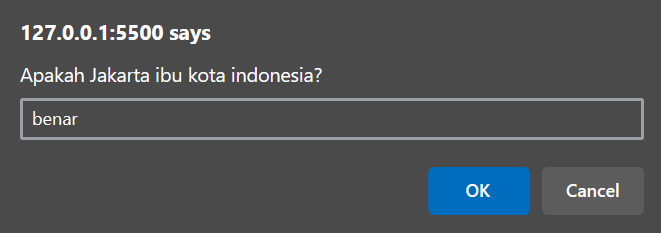
Program ini menggunakan **percabangan switch case**

**Switch (jawab) :** nilai yang dimasukkan user di variable jawab, akan menjalankan beberapa case case di programnya(case 1 - 5), jika variable hadiah kosong **(true)**maka akan menampilkan kalimat “Kamu gagal mendapat hadiah”, jika variable hadiah tidak kosong **(false)** maka akan menjalankan kondisi else, yang akan mencetak kalimat (Selamat kamu mendapatkan (hadiah) ).

1. Percabangan dengan operator ternary (operator\_ternary.html)



* Input



* Output

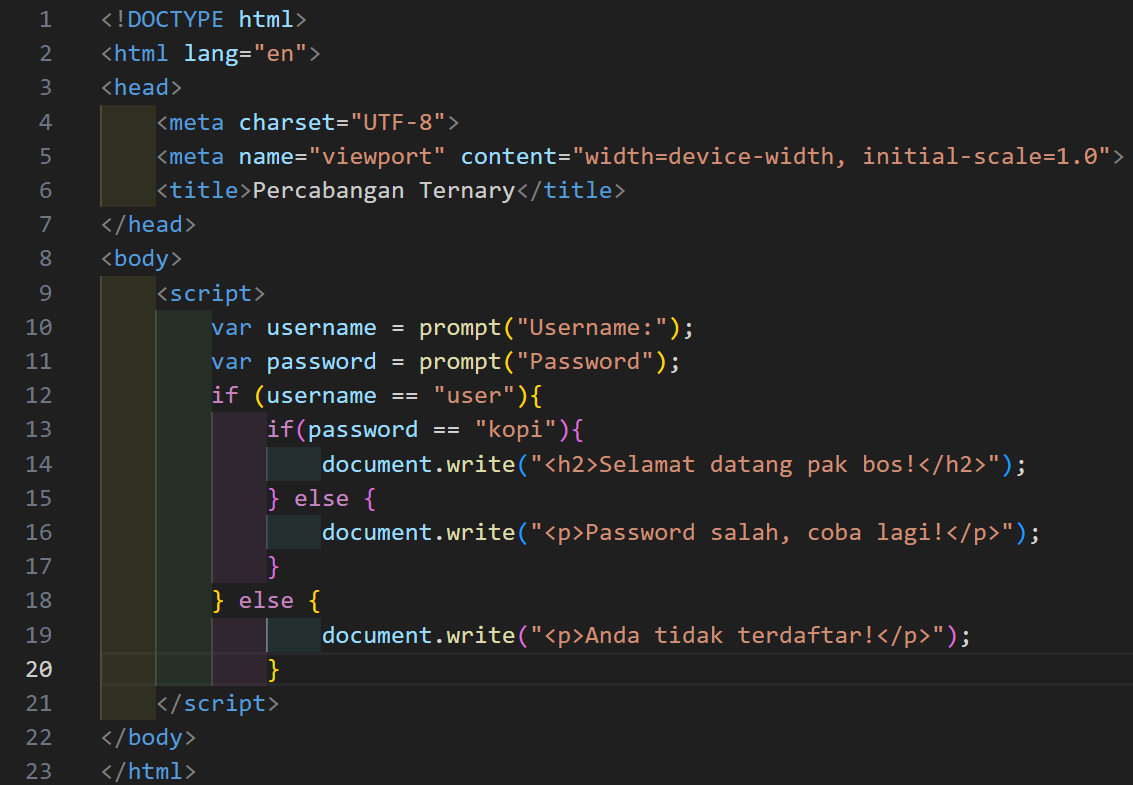


* Analisis

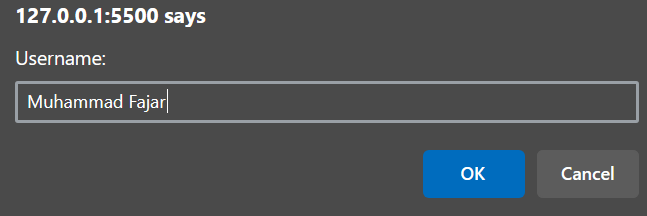
Program ini menggunakan percabangan ternary

* **Program ternary adalah** percabangan yang singkat dengan maksud tertentu yang bertujuan untuk memudahkan user dan menyingkatkan programnya
* (jwb.toUpperCase() == “IYA”) ? “benar” : “Salah” : nilai dari variable jwb diganti dengan huruf kapital semua karena ada toUpperCase, setelah itu apakah nilai tersebut sama dengan IYA, jika benar**(true)** maka akan terisi kalimat “benar” (sebelah kanan) divariable jawaban, dan jika salah **(false)** maka akan terisi kalimat “salah” (sebelah kiri) divariable jawaban
* Setelah itu, program akan mencetak kalimat “Jawaban anda : (jawaban)” pada variable jawaban akan terisi kalimat sesuai denga isi nya yang sudah ditentukan oleh perkondisian **ternary.**

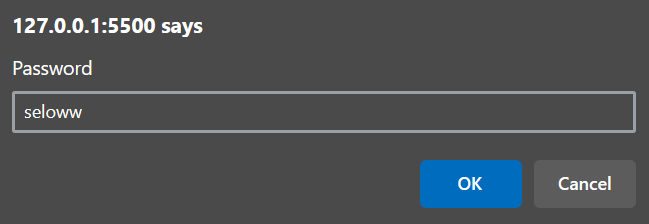
1. Percabangan bersarang (nested.html)



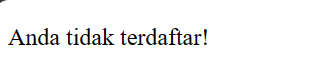
* Input (username)



* Input (password)



* Output

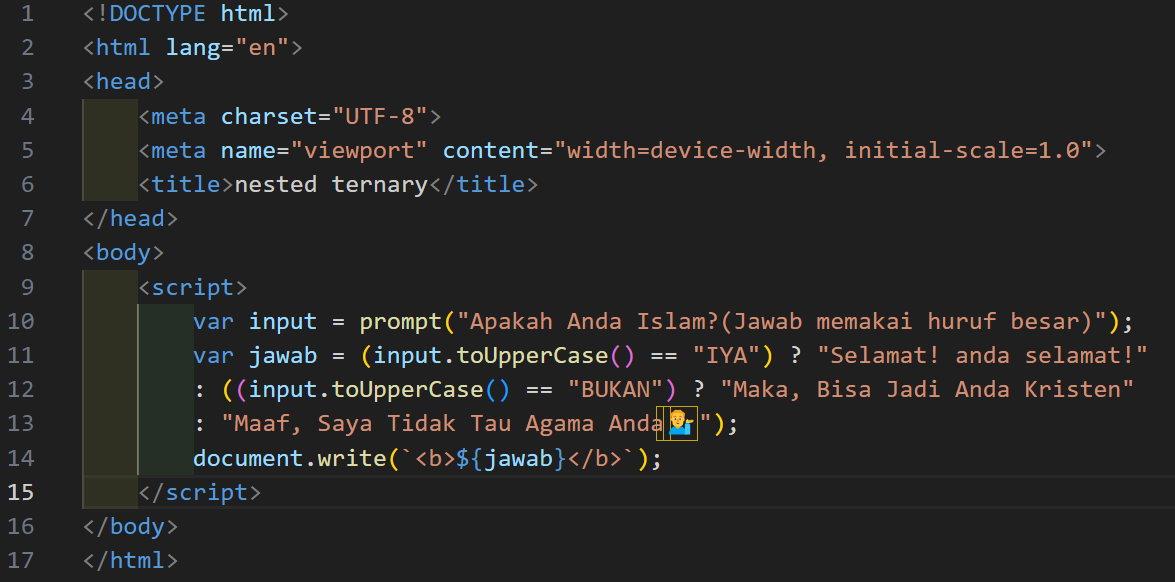


* Analisis

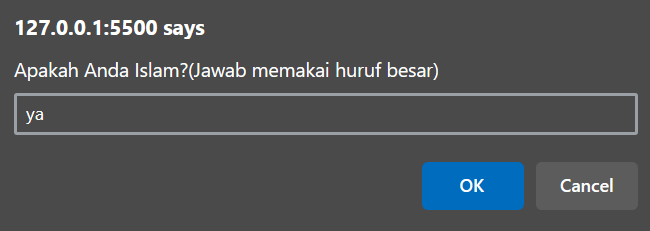
Program ini menggunakan kondisi if bersarang (nested if)

Jika variable username diinput kata “user” **(true)** maka akan dilakukan kondisi if lagi, jika variable password diinput kata “kopi” **(true)** maka akan mencetak kalimat “selamat datang bos!”jika tidak **(false)** maka akan mencetak kalimat “password salah, coba lagi”, jika variable username bukan “user” **(false)** maka dilakukan kondisi else yang akan mencetak kalimat “Anda tidak terdaftar!”

1. Ternary dalam ternary (nested\_ternary.html)



* Input



* Output



Program ini menggunakan percabangan ternary bersarang (ternary nested)

Mirip seperti if bersarang.

* **Var jawab = (input.toUpperCase() == “IYA”) ……… :** nilai input akan diubah menjadi huruf kapital semua, setelah itu diperiksa apakah yang input nya “IYA”, jika benar **(true)** maka akan menampilkan kalimat “Selamat! Anda selamat!” jika salah **(false)** maka akan diperiksa lagi apakah input nya “BUKAN”, jika benar **(true)** maka akan menampilkan kalimat “Maka, Bisa Jadi Anda Kristen”, jika salah **(false)** maka akan menampilkan kalimat (Maaf, Saya Tidak Tau Agama Anda💁‍♂️)
* Dan variable jawab akan ditampilkan dengan tag b (tulisan tebal).

# **KESIMPULAN**

Dengan memahami konsep percabangan dalam JavaScript, pengembang dapat menulis kode yang lebih fleksibel dan responsif, memungkinkan aplikasi untuk merespons secara dinamis terhadap berbagai situasi. Melalui penggunaan statement if-else, switch-case, dan operator ternary, mereka dapat membuat keputusan berdasarkan kondisi tertentu, menjalankan blok kode yang sesuai, dan mengontrol alur program dengan lebih efektif. Percabangan tidak hanya memungkinkan pengembang untuk menangani logika yang kompleks, tetapi juga membantu dalam mengatur kode dengan lebih terstruktur dan mudah dimengerti.

Selain itu, pemahaman yang baik tentang percabangan juga memungkinkan pengembang untuk menghindari duplikasi kode yang tidak perlu. Dengan menggunakan percabangan dengan bijak, mereka dapat menghindari menulis blok kode yang serupa untuk kondisi yang berbeda, sehingga meningkatkan efisiensi dan kemudahan pemeliharaan kode. Ini sangat penting dalam pengembangan perangkat lunak karena dapat mengurangi kerentanan dan memudahkan penambahan atau perubahan fitur di masa mendatang.

Secara keseluruhan, pemahaman yang kuat tentang konsep percabangan dalam JavaScript merupakan aspek kunci dari keterampilan pengembangan perangkat lunak yang efektif. Dengan kemampuan untuk membuat keputusan yang tepat dan mengontrol alur program, pengembang dapat membangun aplikasi yang lebih kuat, responsif, dan mudah dikelola. Oleh karena itu, pengembang perlu mengalokasikan waktu untuk mempelajari dan memahami berbagai teknik percabangan yang tersedia dalam bahasa pemrograman JavaScript.

Top of Form