# Dasar Pemrograman dengan Javascript

## Apa itu pemrograman?

Secara Bahasa :

Program adalah sebuah kumpulan atau serangkaian instruksi

Secara Istilah Dunia IT

Program adalah urutan instruksi terstruktur yang dirancang agar komputer dapat berperilaku sesuai dengan ketentuan dan tujuan untuk menyelesaikan suatu pekerjaan

Dan pemrograman disini maksudnya adalah memberikan **instruksi sederhana** kepada komputer melalui Bahasa pemrograman contoh intruksi sederhana itu seperti melakukan operasi matematika contoh lainnya yaitu menampilkan tulisan ke layar

Perintah perintah yang kita tulis dalam bentuk Bahasa pemrograman itu disebut dengan **statement** dan statement yang kita buat haruslah sederhana terstruktur dan masuk akal karena apapun instruksi yang diberikan selama statementnya dimengerti oleh komputer maka akan dieksekusi jadi harus memberikan instruksi yang benar

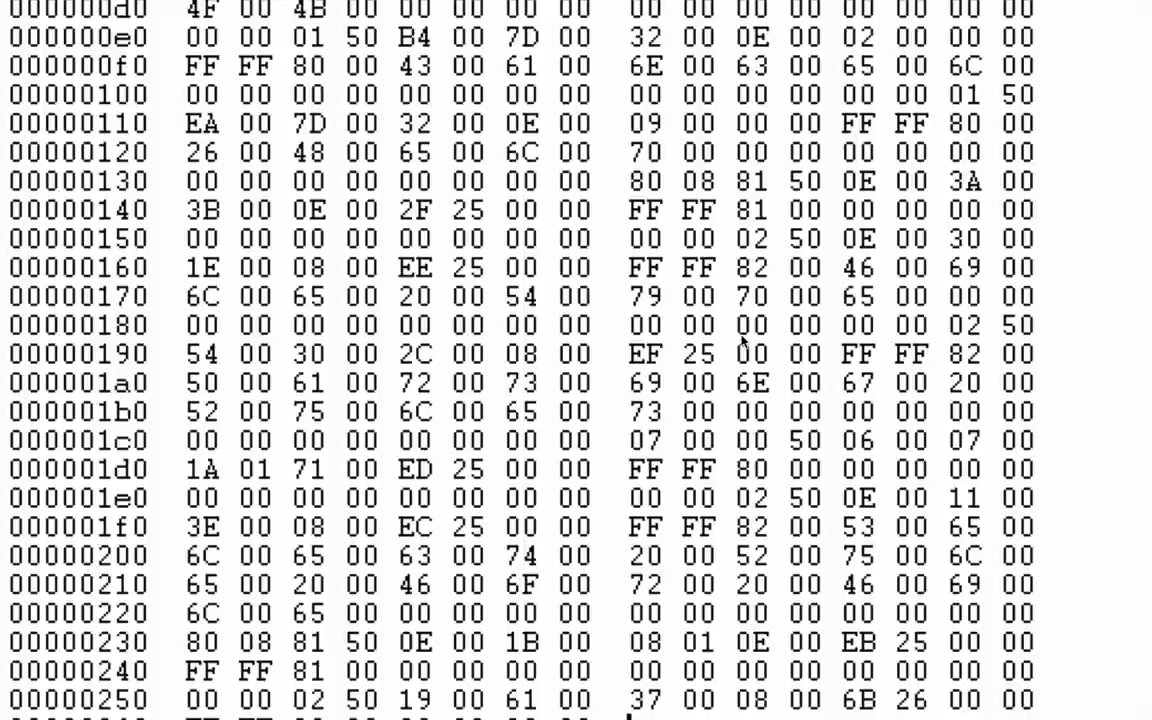
Singkatnya pemrograman itu adalah cara kita untuk memberikan instruksi/perintah sederhana dan terstruktur kepada komputer dan instruksi yang kita berikan haruslah sederhana, benar dan masuk akal dan juga statement yang kita buat harus mengikuti aturan pada Bahasa pemrograman tertentu

## Apa itu Bahasa Pemrograman?

Bahasa pemrograman merupakan Pustaka kumpulan instruksi instruksi untuk membuat perintah terhadap komputer

Bahasa pemrograman dibuat sejak komputer dibuat dan ada banyak sekali Bahasa pemrograman dan terus bertambah seiring berkembangnya teknologi, mengapa ada banyak karena biasanya Bahasa pemrograman tertentu dibuat untuk tujuan tertentu juga jadi tidak semua Bahasa pemrograman bisa melakukan hal yang sama

Tapi sebetulnya perangkat yang kita gunakan sehari hari seperti komputer tidak mengerti Bahasa pemrograman yang banyak tersebut .komputer sebenarnya hanya mengerti satu Bahasa yang disebut dengan Bahasa mesin yang biasanya instruksi intruksinya itu ditulis dengan binner,hexa dll



Dan karena Bahasa mesin cenderung sulit ditulis dan dibaca oleh manusia maka dibuatlah Bahasa pemrograman yang lebih mudah ditulis dan dibaca , Bahasa pemrograman juga memiliki tingkatan dengan sebutan High Level Language dan Low Level Language

High Level Language cenderung lebih mudah dibaca dan ditulis oleh manusia sedangkan Low Level Language merupakan Bahasa yang cenderung lebih mudah dibaca oleh mesin, semakin tinggi level Bahasa pemrograman semakin mudah dipahami namun semakin berat untuk mesin karena harus melalui proses konversi yang disebut dengan compilasi atau interpretasi kedalam Bahasa mesin

Bahasa pemrograman ditulis dengan Plain Text atau hanya text biasa dan bisa ditulis menggunakan text editor biasa seperti notepad di window ataupun textedit di mac namun ada text editor yang memiliki fitur lebih untuk memudahkan kita dalam menulis Bahasa pemrograman yang disebut dengan Code Editor yang memiliki fitur seperti syntax highlighting debug tool dan auto complete seperti Sublime Text dan Visual Studio Code ada juga yang lebih canggih dari Code Editor yaitu IDE atau Integrated Development Environment yang memiliki fitur lebih lengkap untuk menulis dan menjalankan code contohnya seperti NetBeans, Visual Studio dan Intelij IDEA

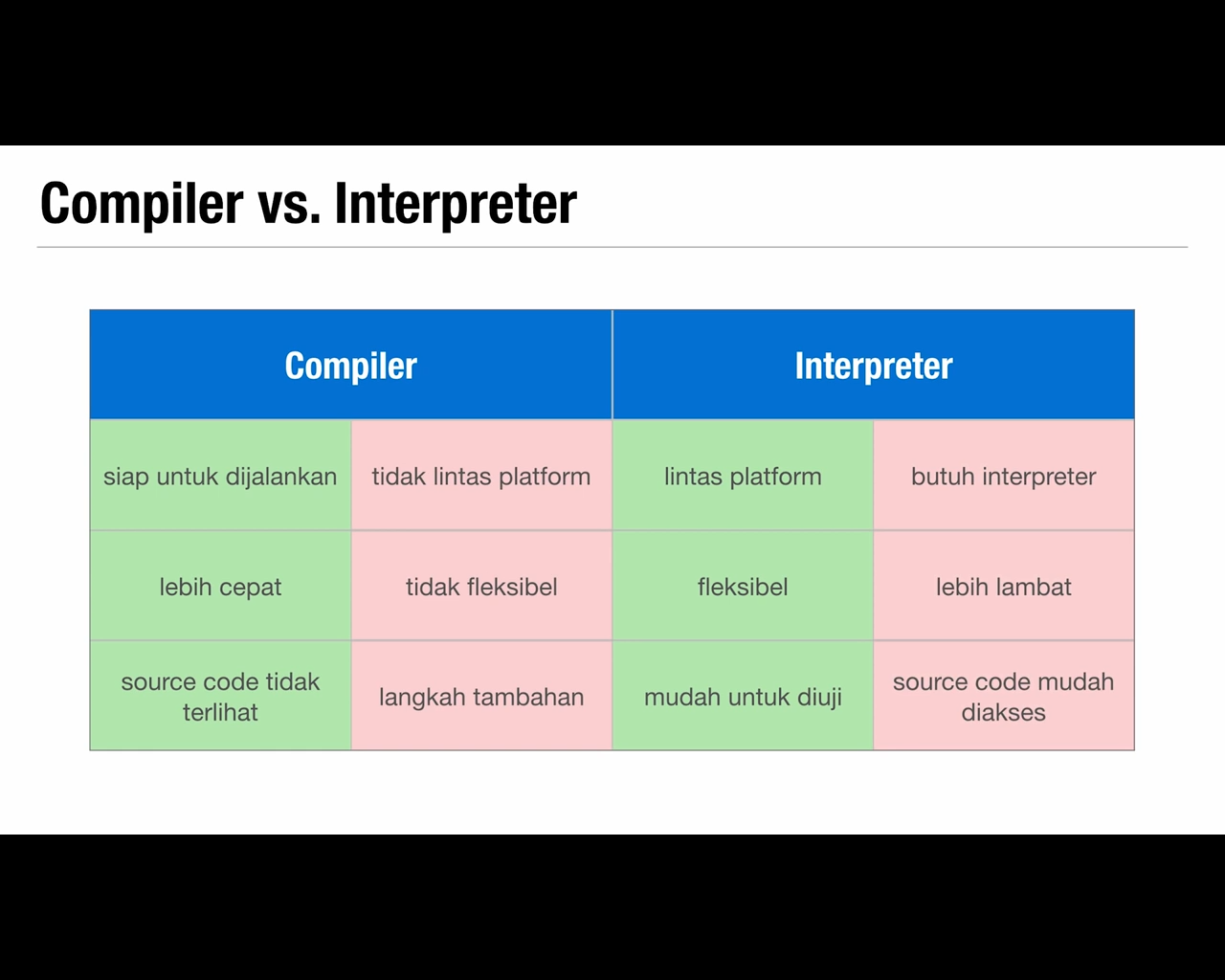
Kumpulan instruksi instruksi menggunakan Bahasa pemrograman disebut dengan Source Code

lantas bagaimana cara komputer bisa tetap memahami instruksi yang diberikan ? disini lah peran sebuah compiler dan interpreter sebagai penengah atau lebih tepatnya penerjemah dari Bahasa pemrograman ke Bahasa mesin

## Compiler & Interpreter

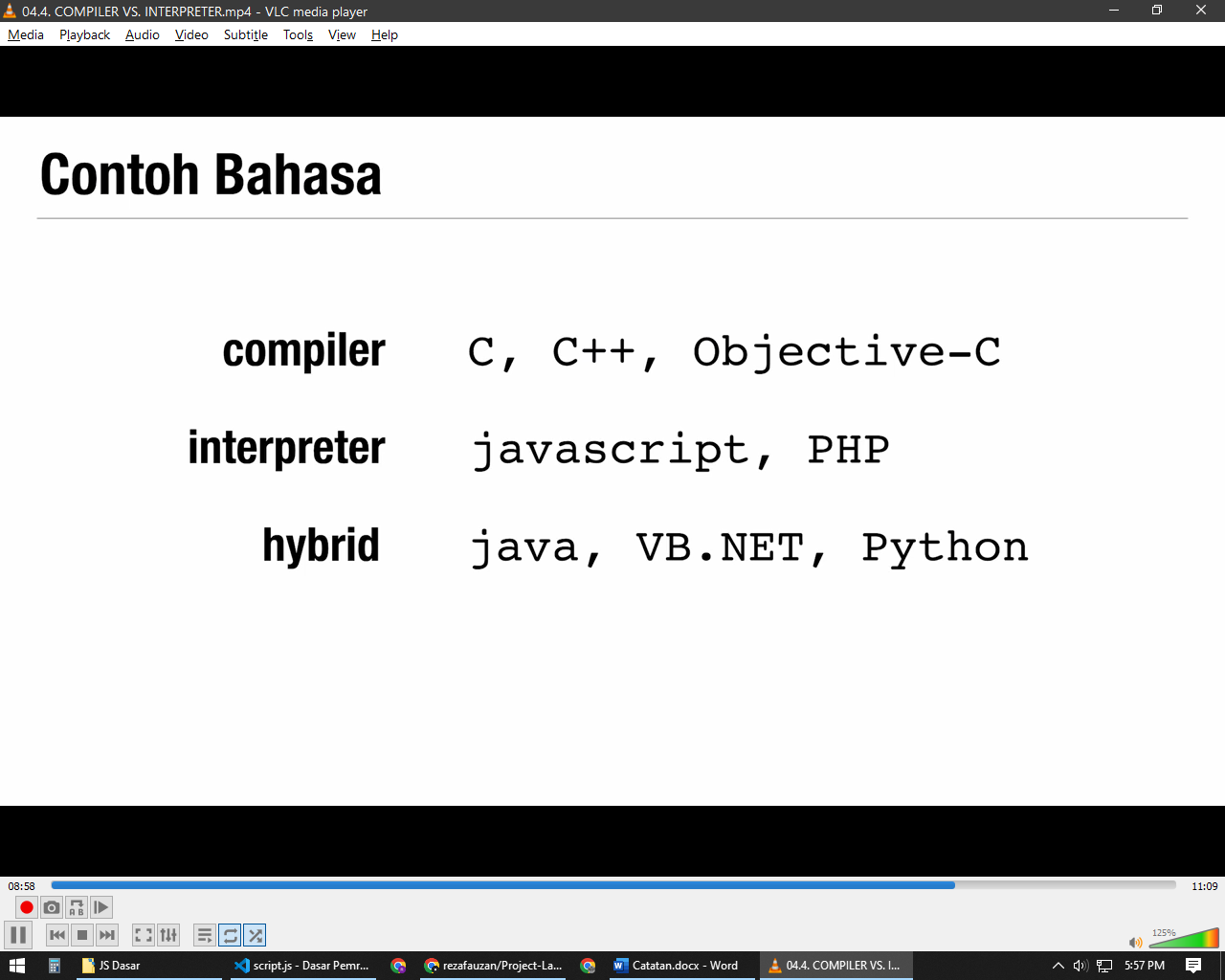
Compiler merupakan sebuah perangkat lunak untuk melakukan penerjemahan source code ke Bahasa mesin untuk kemudian menghasilkan file yang bisa dijalankan atau dieksekusi yang disebut dengan Executable biasanya menghasilkan file dengan format .exe nah proses ini disebut dengan compilasi program .exe ini bisa dijalankan dimanapun tanpa harus melalui compiler lagi

Sedangkan Interpreter source code yang ditulis tidak memerlukan konversi ke Bahasa mesin namun untuk menjalankannya memerlukan interpreter juga dan tetap berupa source code

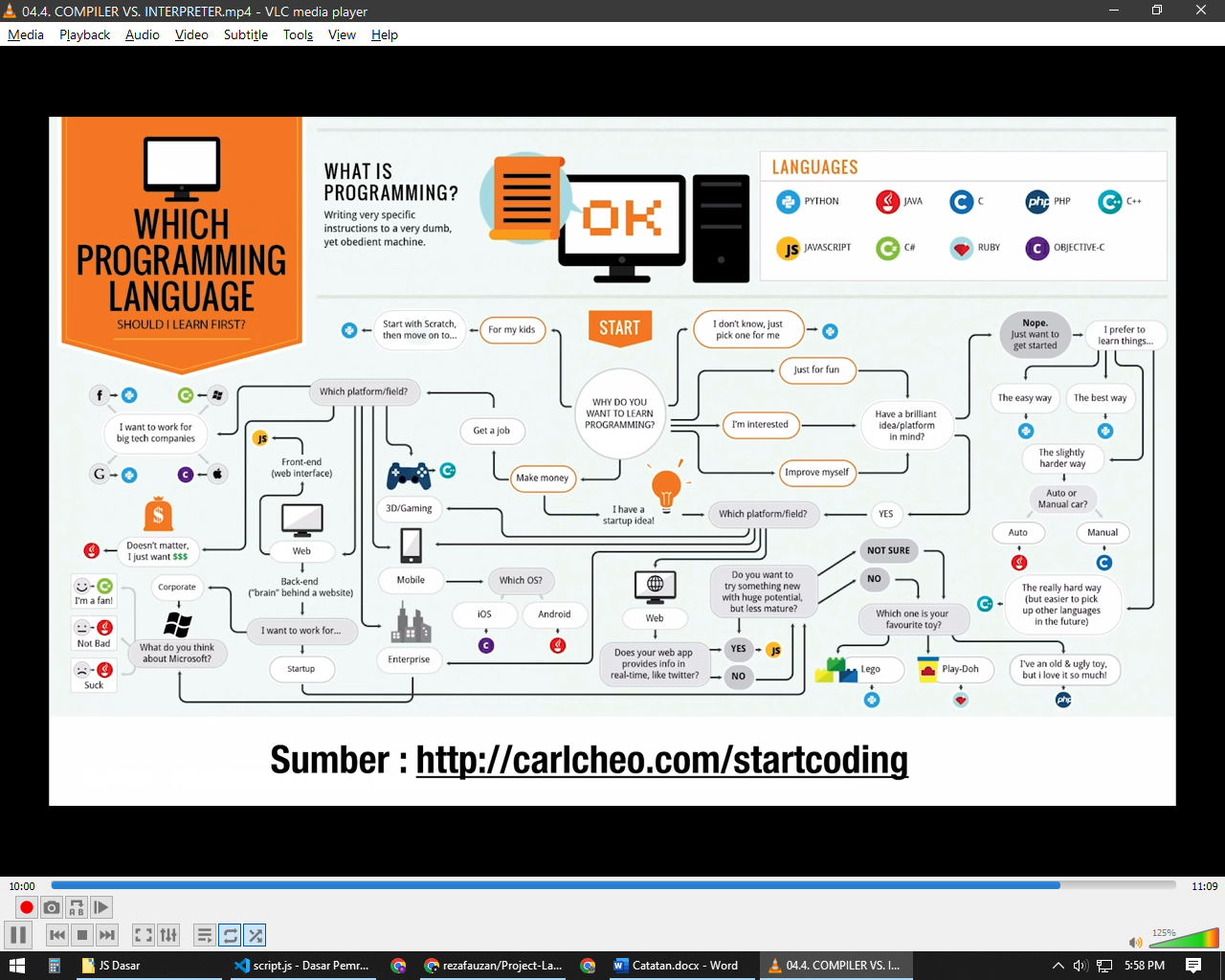


Namun ada juga Bahasa pemrograman yang menggunakan proses compilasi dan interpretasi atau disebut dengan **Hybrid** cara kerjanya yaitu source code yang ditulis dicompilasi namun bukan menjadi executable melainkan menjadi intermediate language atau biasa disebut byte code

Byte code ini sebelum dijalankan akan dilakukan yang Namanya Just In Time Compilation (JIT Compilation) yaitu melakukan compilasi sambal menjalankan programnya



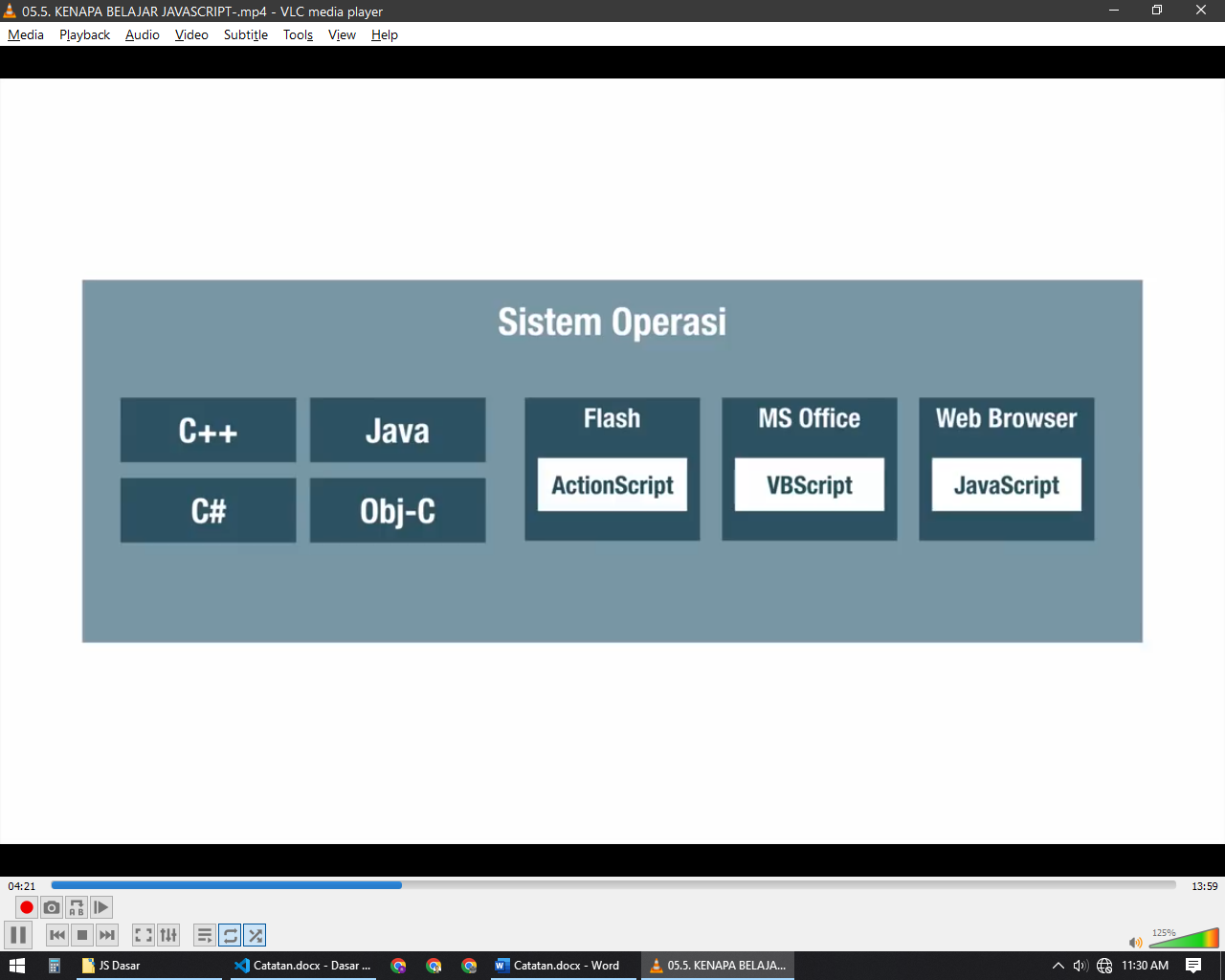




## Mengapa Javascript?

Javascript merupakan sebuah Bahasa pemrograman yang **High Level ,Scripting, Untyped, dan Interpreted.** Karena Javascript merupakan Bahasa yang high level sehingga akan lebih mudah dipelajari dan dipahami

Javascript disebut dengan Bahasa **Scripting** yang artinya javascript berjalan diatas program lain yang lebih besar



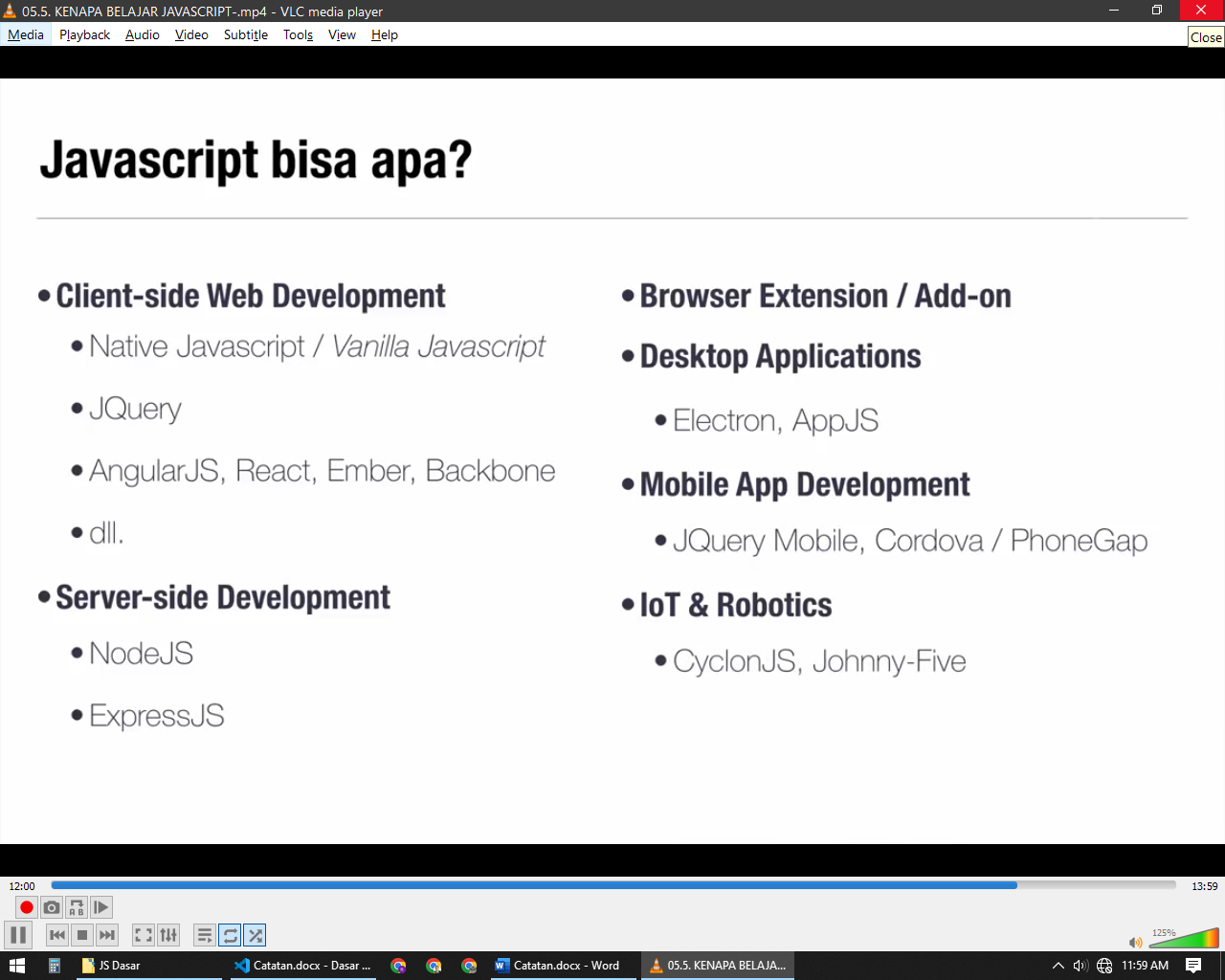
Javascript juga merupakan Bahasa yang Untyped atau juga disebut Dynamically Typed yang artinya javascript tidak perlu mendeklarasikan type data nya , ketika membuat variable, variablenya bisa diisi dengan type data apapun ini berbeda dengan Bahasa pemrograman yang Statically Typed atau Typed yang dimana sebelum membuat variable diharuskan untuk mendeklarasikan type datanya dan hanya bisa diisi satu type data saja

Javascript bisa digunakan untuk **Client Side Web Development(Front End)** atau membuat website dari sisi client dengan native JS atau disebut Vanila JS maupun menggunakan Framework Seperti Jquery React dll

Dan juga sekarang Javascript bisa digunakan untuk **Server Side Development(Back End)** menggunakan Node JS Atau Express JS

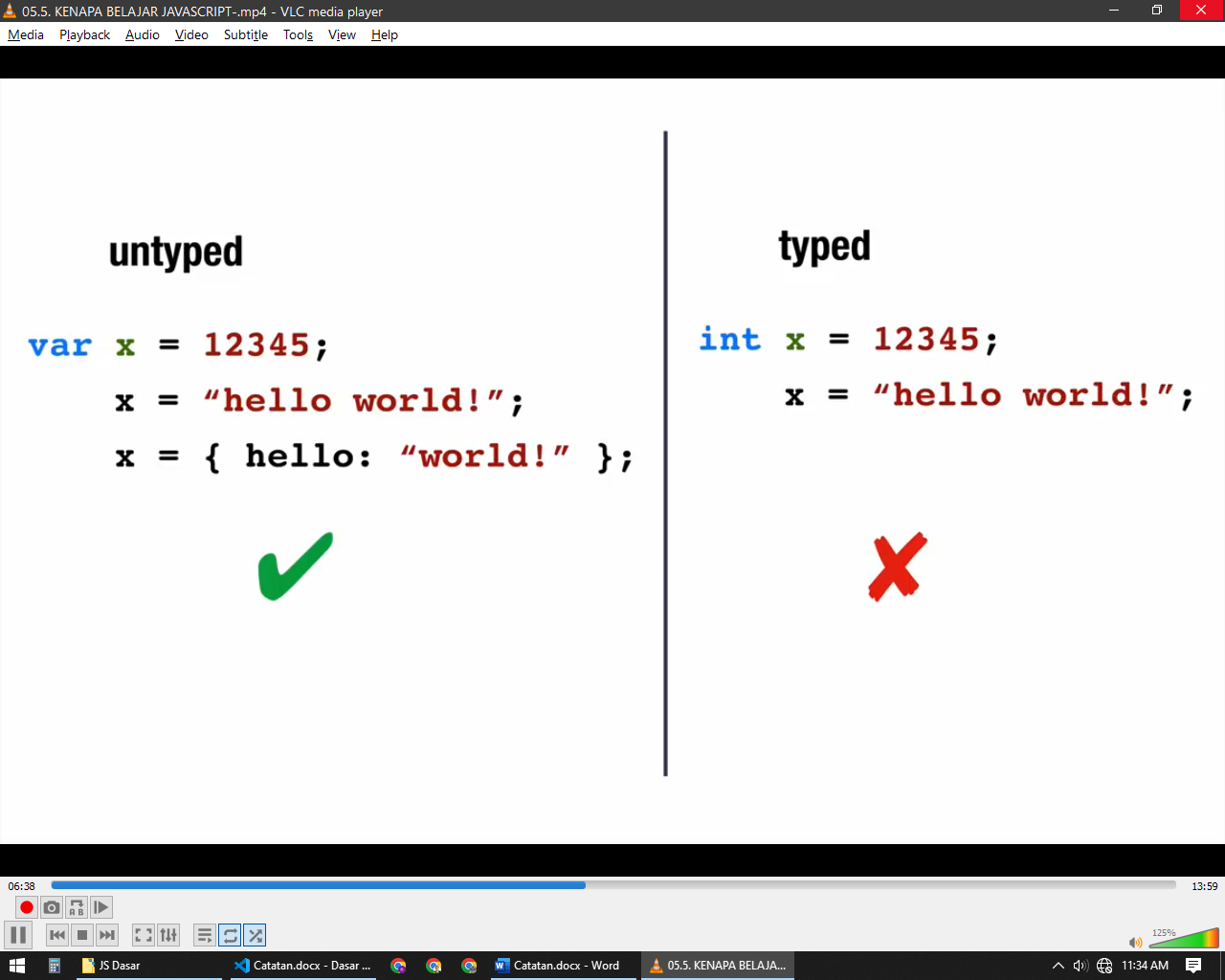
Javascript juga biasanya digunakan dalam pembuatan browser Extension/Add On,Aplikasi Dekstop menggunakan Electron,App JS dan Mobile menggunakan JQuery Mobile,Cordova

Javascript Juga bisa digunakan untuk membuat IOT & Robotics menggunakan CyclonJS dan Jhonny Five



Dan Javascript merupakan Bahasa yang interpreted dimana untuk menjalankannya kita memerlukan interpreter dan khususnya javascript memerlukan paling tidak aplikasi browser untuk menjalankannya

Javascript gratis dan tidak memerlukan instalasi program khusus cukup menggunakan browser yang biasanya sudah terdapat Javascript interpreter atau disebut Javascript Engine

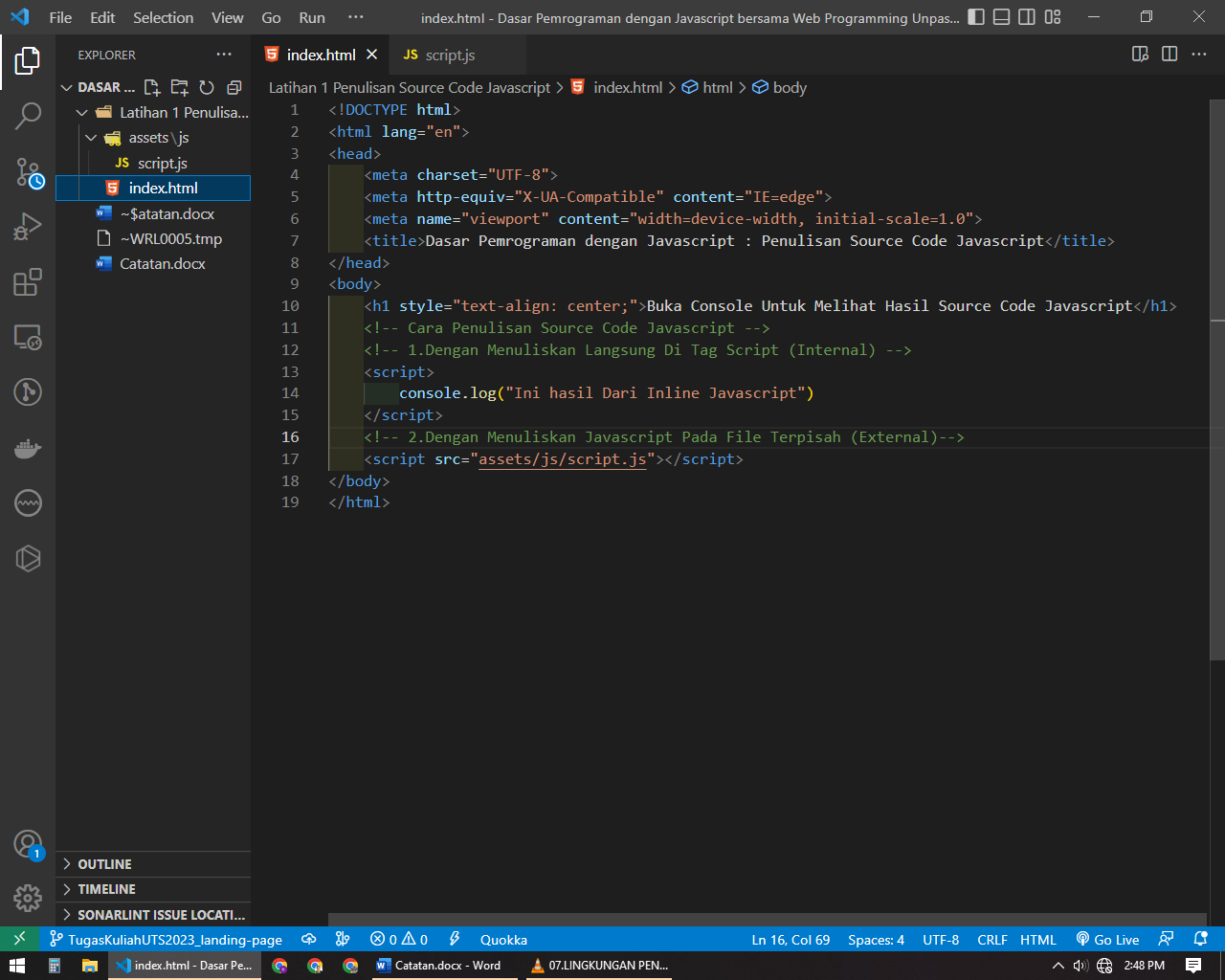


## Lingkungan Pengembangan Javascript

Javascript merupakan Bahasa scripting yang bisa dijalankan dengan javascript engine atau interpreter yang terdapat didalam browser jadi untuk pengembangan javascript hanya membutuhkan sebuah browser dan code editor

Dan untuk memulai menulis source code javascript cukup dengan menambahkan script kedalam html atau menuliskan dan menjalankan langsung melalui console yang ada pada browser yang bisa di akses melalui hotkey F12 atau klik kanan inspect element masuk ke tab console

Penulisan source code javascript di dalam HTML bisa melalui langsung dituliskan didalam html didalam tag script Teknik ini dinamakan inline JS dan ada juga dengan menuliskan source code pada file terpisah dinamakan external js

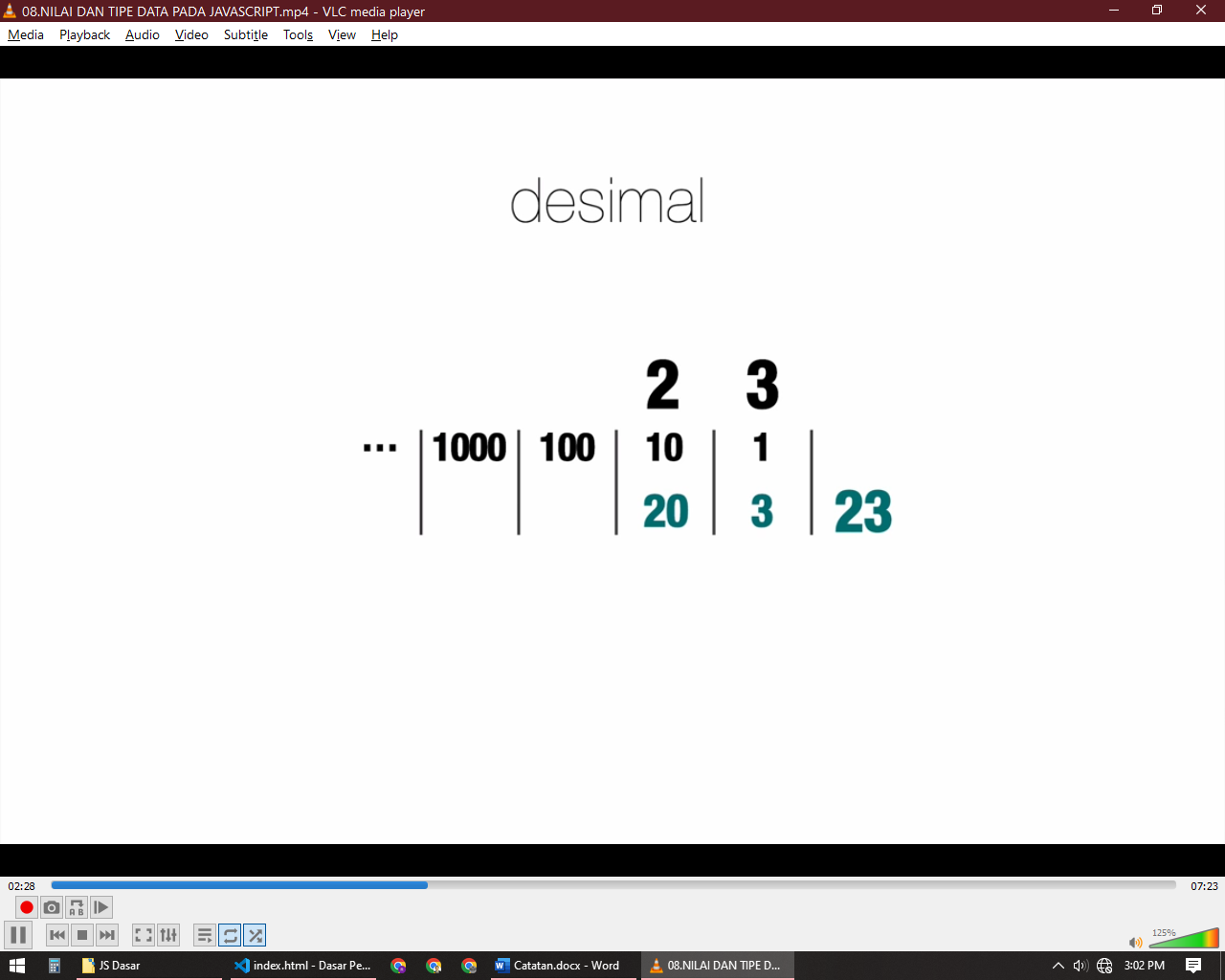


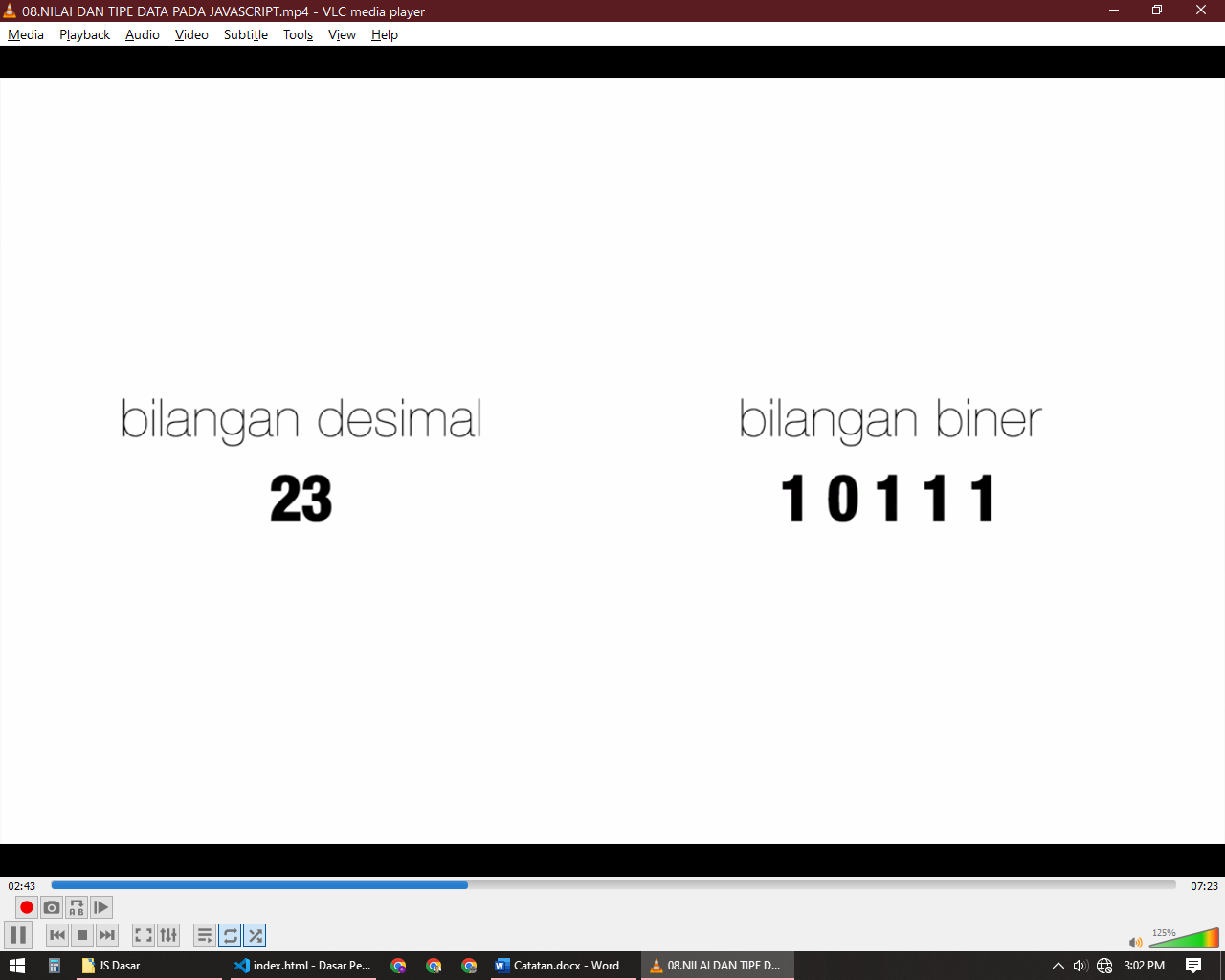
## Nilai dan Tipe Data Pada Javascript

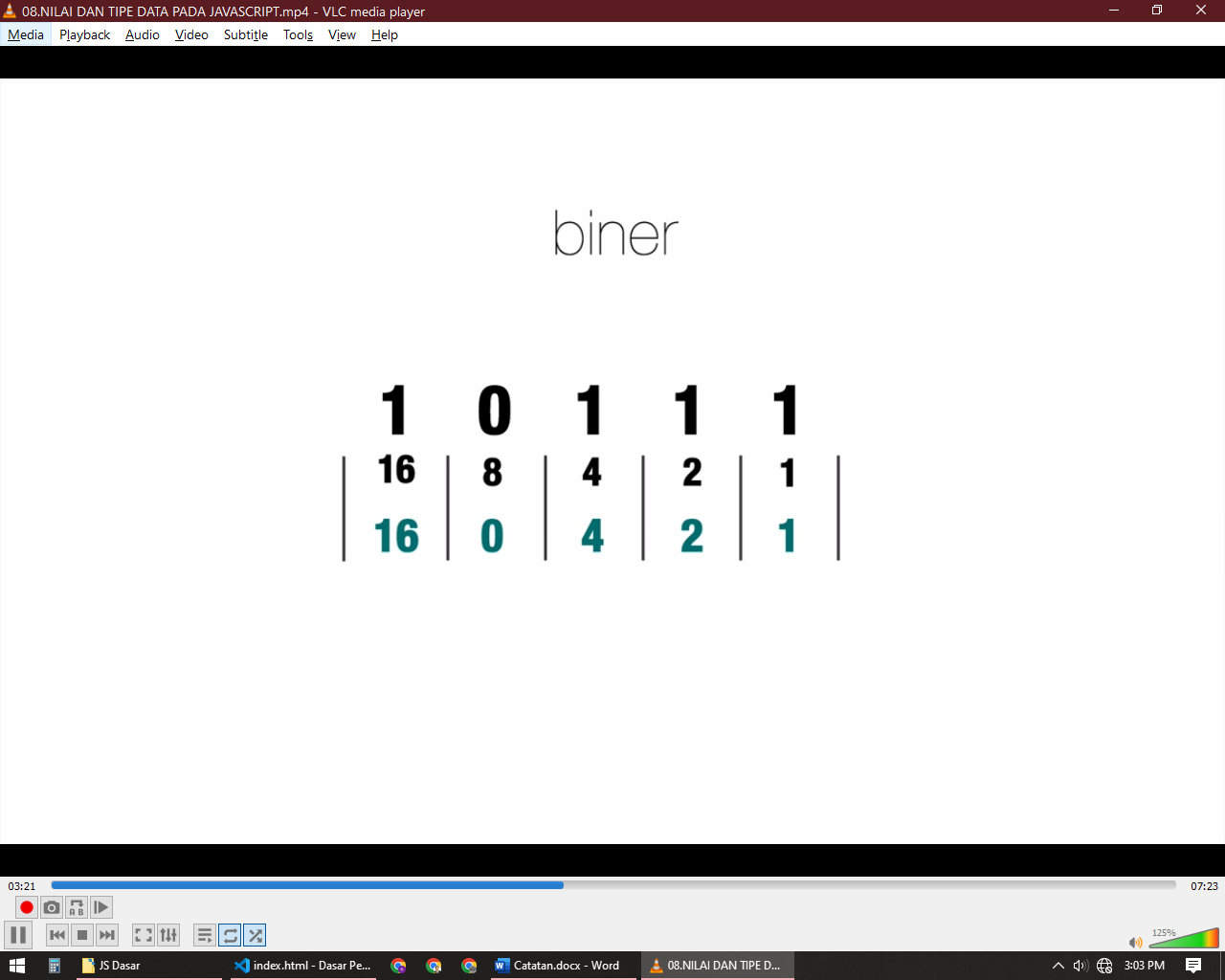
Sejatinya Komputer bergerak dan bisa melakukan sesuatu karena bagian terkecil dalam komputer yang disebut dengan data dan data ini tersimpan kedalam serangkaian bit/binary digit yaitu satuan terkecil dari data yang ada di dalam komputer

Bit ini terdiri dari dua nilai yaitu 0 dan 1 karena secara elektronika digit 1 terjadi ketika salah satu komponen didalam komputer yang Bernama transistor dialiri oleh listrik dan 0 ketika transistor tidak dialiri listrik

Sebagai contoh komputer menampilkan bilangan 23 bagaimana sebuah komputer merepresentasikan angka 23 dengan binary ini







Dan didalam komputer terdapat banyak sekali bit dan untuk bekerja dengan bit bit tersebut tanpa bingung sebuah komputer membagi bit bit kedalam potongan yang disebut dengan nilai/value

Dan meskipun nilai terdiri dari sekumpulan bit bisa saja nilai tersebut memiliki peran yang berbeda yang disebut dengan Tipe Data

Di dalam javascript terdapat 6 tipe data yaitu :

* Number

Di dalam javascript tidak ada yang Namanya bilangan bulat atau integer melainkan hanya ada pecahan atau **floating point**  dan ukuran tampungannya sebesar 64-bit( ) atau lebih tepatnya sebesar 18.446.744.073.709.551.616

Akan tetapi angka tersebut tidak dapat digunakan semuanya dan ada beberapa jenis angka yang bisa dijadikan nilai pada tipe data number ini yaitu

1. Angka tanpa decimal ex: 10,100,200 (namun angka ini akan direpresentasikan sebagai pecahan dengan akurasi 15 digit)
2. Angka dengan decimal ex: 0.4,0.5,0.6,0.9 maksimal 17 digit dibelakang koma
3. Eksponen ex: 123e5 //12300000, 123e-5 // 0.00123
4. Bilangan Negatif ex: -888

Ps:jangan mengawali angka di Javascript dengan angka 0 karena akan dianggap sebagai bilangan oktal(basis 8 ex: 022 //18) sama juga jika mengawali dengan 0x akan dianggap sebagai nilai hexadecimal(basis 16 ex: 0xFF //255)

1. Angka Special
   * Infinity ex: 2/0 //infinity
   * -infinity ex: -2/0 //-infinity
   * NaN(Not a Number) ex : 0/0 //NaN

NaN ini juga muncul ketika angka dibagi dengan bukan angka ex: 100/”Apel” = NaN

Namun apabila stringnya berupa angka javascript akan menganggapnya sebagai angka sehingga berlaku operasi matematika ex : 100 / “10” = 10



* String
* Boolean
* Object
* Function
* Undefined

Dan karena javascript merupakan untyped language untuk membuat sebuah variable dengan tipe data number bisa langsung menuliskan nilainya dengan angka dan untuk string tinggal menuliskannya didalam kutip begitu juga dengan tipe data lainnya contoh :

* Number

let x = 1945

* String

let nama = “Reza Fauzan Adhima”

* Boolean

let statusBekerja = true

* Object

let makananKesukaan = {

“Nasi Goreng”,

“Kwetiau Rebus”,

“Daging Panggang”,

“Keju”

}

* Function

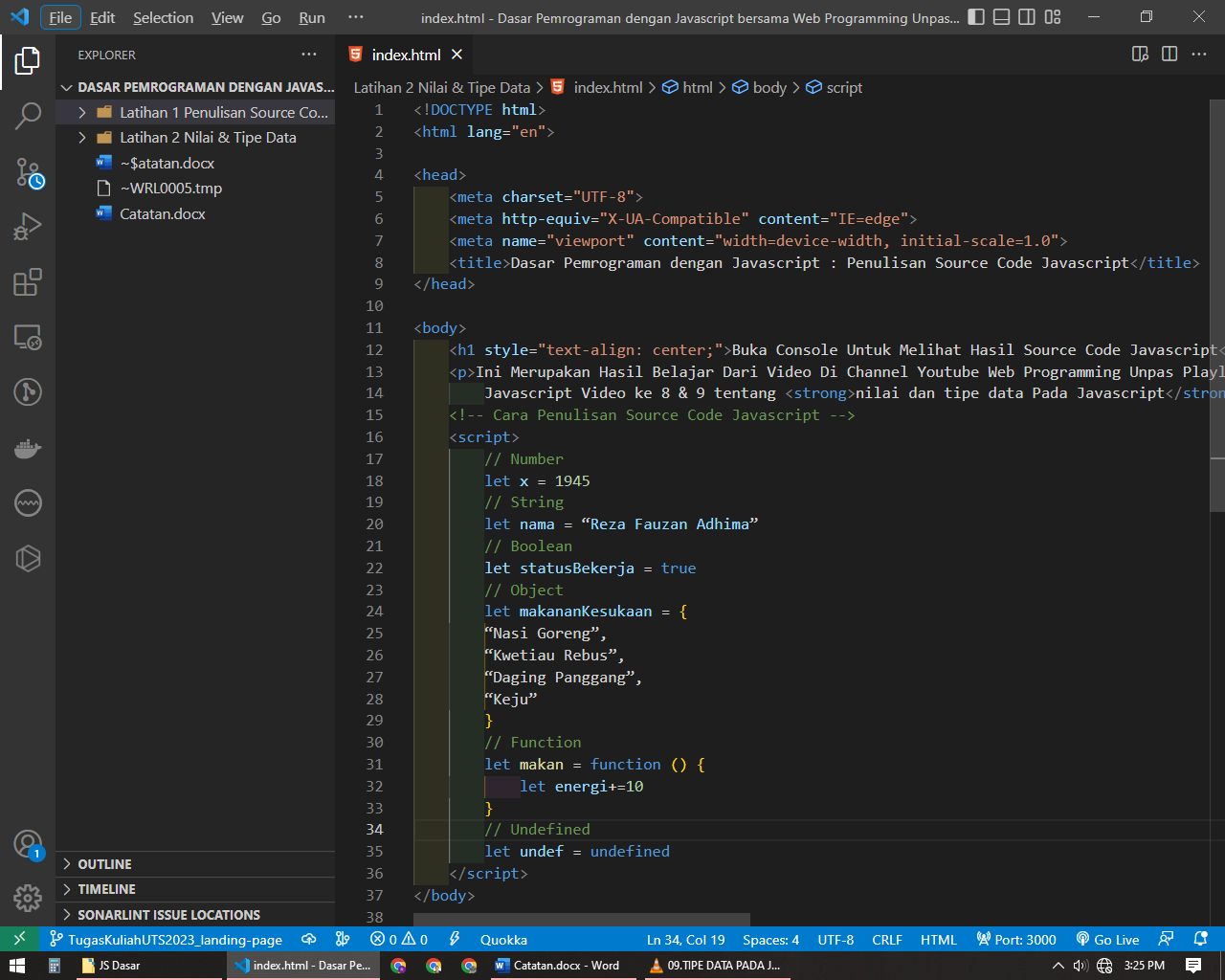
let makan= function (){

let energi+=10

}

* Undefined

let x = undefined



Nilai-nilai tersebut akan secara otomatis dirubah kedalam bit secara otomatis oleh javascriptnya dan setiap nilai yang dibuat akan disimpan disuatu tempat di komputer pada umumnya terletak di dalam memory

Selama program berjalan nilai nilai tersebut akan disimpan dan bisa memenuhi kapasitas pada ram akan tetapi ini sangat jarang terjadi karena biasanya program program sederhana tidak memakan begitu banyak RAM

Ketika program dihentikan nilai yang tersimpan didalam RAM secara otomatis dihilangkan dan bit pada RAM Menjadi kosong untuk digunakan lagi

## Operator

Operator adalah sebuah symbol yang digunakan untuk melakukan operasi terdapat beberapa operator yang ada pada javascript diantaranya adalah Operator Binary, Ternary dan Unary :

1. **Operator Binary**

Operator Binary membutuhkan 2 operand (***operand1 operator operand2***) contoh 4 + 5

(4) sebagai operand1 (+) sebagai operator dan (5) sebagai operand2

* + Operator Aritmatika( **+ , - , / , \* , %** )

Operator ini digunakan untuk melakukan operasi matematika dan memiliki ***Operator Presedense*** yaitu urutan operasi matematik (Kur() Ka\* Ba/ Ta+ Ku-)

* + Operator Penugasan / Assignment ( **= , += , -= , \*= , /=, %=** )

Contohnya adalah var x = 10

Ini artinya 10 akan di tugaskan untuk berada di dalam variable x

Meskipun tandanya adalah = ini tidak bermakna perbandingan tapi penugasan

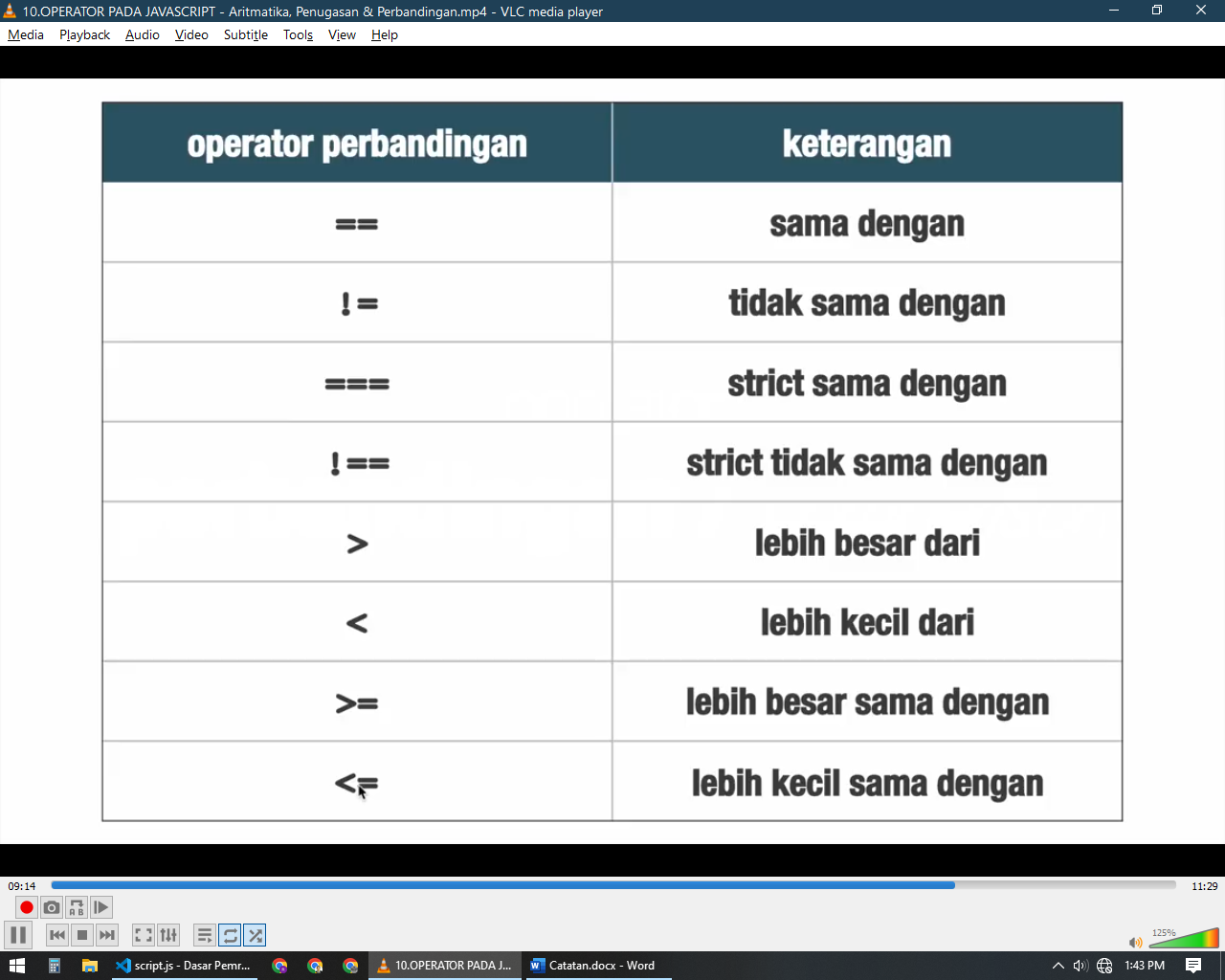
Dan nilai variable ini disimpan di dalam memory

Operator = juga biasa digunakan untuk menimpa nilai

Akan tetapi operator selain = tidak menghilangkan nilai sebelumnya melainkan menjumlah atau mengurang sesuai operatornya

Misal x += 2 maka hasilnya akan 12

* + Operator Perbandingan ( **== , != , === , !== , > , < , >= , <=** )



Dan operator ini menghasilkan nilai Boolean (True / False) contohnya 10>5 maka akan menghasilkan true

* + Operator Logika ( **&&(AND) , ||(OR) , !(NOT)**)

Operator ini digunakan untuk menentukan logika dari beberapa ekspresi yang digabungkan

Contoh:

(x > y) && (x%2)

Operator AND ini akan mengecek dua kondisi dan harus bernilai true keduanya

(x > y) || (x%2)

Operator OR ini akan mengecek dua kondisi dan hanya salah satu yang harus bernilai true

!(x > y)

Sedangkan Operator NOT akan membalikan hasil yang harusnya true jadi false atau sebaliknya

* + Operator String ( **+ (Concatenate)** )

Di dalam javascript operator (+) ini memiliki dua fungsi jika operandnya adalah angka maka operatornya menjadi aritmatika sedangkan jika operandnya adalah string maka akan menjadi operator penggabung string (Concatenate) dan apabila salah satu operandnya adalah string maka akan dianggap operator string

1. **Operator Ternary ( (Kondisi) ? Statement Jika kondisi True : Statement Jika Kondisi False )**

Operator Ternary membutuhkan 3 operand

* + Operator Conditional

Operator ini digunakan untuk melakukan pengecekan pada sebuah kondisi dan menentukan nilai yang dihasilkan ketika kondisinya bernilai true atau false

Contoh : ( x%2 == 0 ) ? “Genap” : “Ganjil”

1. **Operator Unary**

Operator Unary hanya membutuhkan 1 operand (***operator operand atau operand operator***)

* + Operator TypeOf(operand)

Operator ini digunakan untuk mengembalikan atau mencari type data pada suatu nilai

Contoh : typeof(true)

## Type Data String

## 

