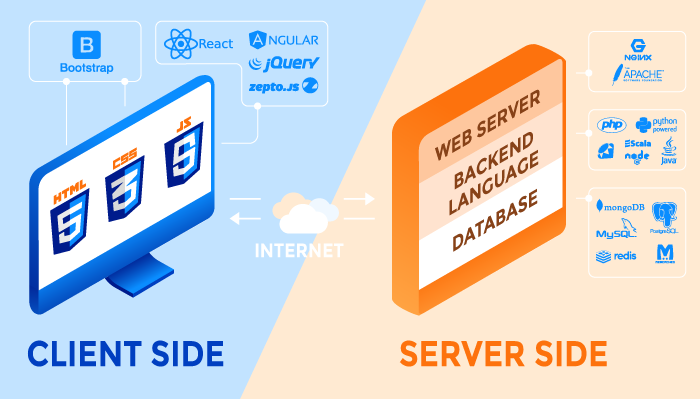
**Web Development**

*Oleh: Muhammad Hendri Januri*

*Web development* adalah suatu proses yang meliputi pembuatan website untuk jaringan internet atau jaringan privat (intranet). *Web development* mencakupi mulai dari pembuatan website statis yang sangat simpel sampai website dinamis yang sangat komplek seperti aplikasi *web*, *e-commerce*, dan sosial media. Singkatnya proses *web development* merupakan proses pembuatan aplikasi yang bisa berjalan didalam web browser sehingga dapat dipakai oleh pengguna.

Didalam proses jaringan komputer web terdapat dua sisi yang disebut *client side* dan *server side*. *Client side* berkaitan dengan bagaimana pengguna terhubung dengan aplikasi *web*, sedangkan *server side* berkaitan dengan hal-hal yang berkaitan di belakang layar seperti database dan keamanan jaringan. Kedua sisi ini saling menopang satu sama lain dan menjadi dasar dari pengembangan *web*. Masing-masing sisi memiliki tipe pengembangan yang berbeda yaitu *Front-end Development* untuk *client side* dan *Back-end Development* untuk *server side*.



[*https://rubygarage.s3.amazonaws.com/uploads/article\_image/file/709/technology-stack-diagram.jpg*](https://rubygarage.s3.amazonaws.com/uploads/article_image/file/709/technology-stack-diagram.jpg)

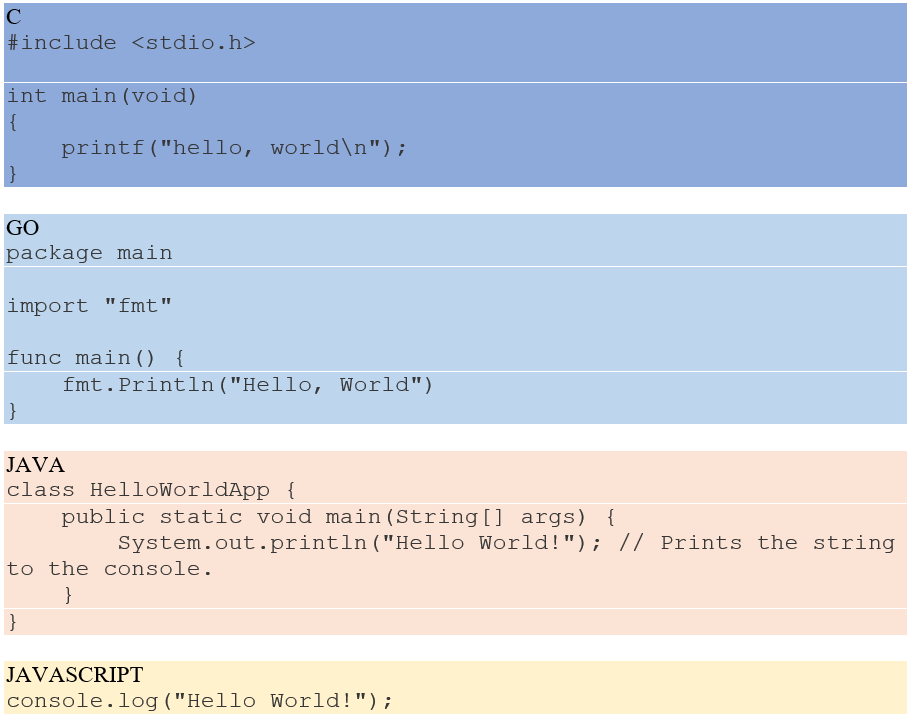
Teknologi yang dipakai dalam *front-end* meliputi HTML, CSS, dan Javascript. Teknologi yang dipakai dalam *back end* meliputi Apache, PHP, Ruby, Python, Node.js, MySQL, MongoDB, dan lain lain. Pengembangan *web* yang mencakup *Front-End* dan *Back-End* disebut sebagai *Full Stack Development.*

Seperti disebutkan dalam paragraf diatas dalam proses pengembangan *front-end* memakai tiga teknologi HTML, CSS, dan Javascript. Jika dianalogikan dengan proses pembangunan sebuah rumah, HTML berfungsi sebagai struktur pondasi dan desain ruangan. Sama seperti pembangunan rumah, proses pembangunan website yang baik tidak bisa dilakukan jika tanpa struktur yang solid. CSS disini berperan sebagai pemberi corak atau warna dalam sebuah *website* agar terlihat menarik secara visual. Hal seperti warna latar belakang halaman, jenis dan ukuran font yang digunakan, ditentukan dalam CSS. Seperti rumah, maka peran CSS disini sebagai penentu warna cat tembok, desain dekorasi kamar, dan sebagainya. Sedangkan Javascript adalah bahasa pemrograman yang fungsinya untuk membuat website menjadi lebih dinamis dan interaktif.

JavaScript atau disingkat JS adalah bahasa pemrograman yang paling umum dan paling populer dalam dunia web karena hampir semua browser web menjalankan Javascript. Nama lain dari Javascript adalah ECMAScript. JS memiliki beberapa karakteristik yang membuatnya sangat populer bagi developer, seperti:

* *High Level*: menggunakan elemen menyerupai bahasa natural manusia, sehingga penggunaannya sangat mudah. Bahasa ini memiliki fitur untuk mengotomasi proses manajemen memori komputer, sehingga pengguna bisa lebih fokus untuk pengerjaan koding.
* *Dynamic Typed*: bahasa ini tidak terikat aturan untuk harus menentukan tipe data seperti apa dalam proses pendeklerasian variabel, sehingga proses pengerjaan lebih efektif dan cepat.
* *Multi Paradigm* : tidak seperti bahasa pemrograman Java yang mengharuskan penggunaan paradigma OOP (Object Oriented Programming), JS cukup fleksibel untuk menggunakan berbagai paradigma seperti Functional Programming, Prototype-Based, atau bahkan Imperative programming seperti bahasa C.
* *Interpreted*: kelebihan dari bahasa dengan basis interpreted daripada bahasa dengan basis kompilasi adalah JS tidak perlu melalui proses kompilasi sehingga dari sisi performa, bahasa tersebut lebih cepat.

Pada awal pengembangannya Javascript hanya digunakan untuk operasional pada sisi *client side*, tapi pengembangan teknologi *runtime environment* seperti Node.js memungkinkan penggunaan bahasa Javascript di sisi *server side*.



*Perbedaan antara berbagai bahasa pemrograman dalam penulisan “Hello World!”*

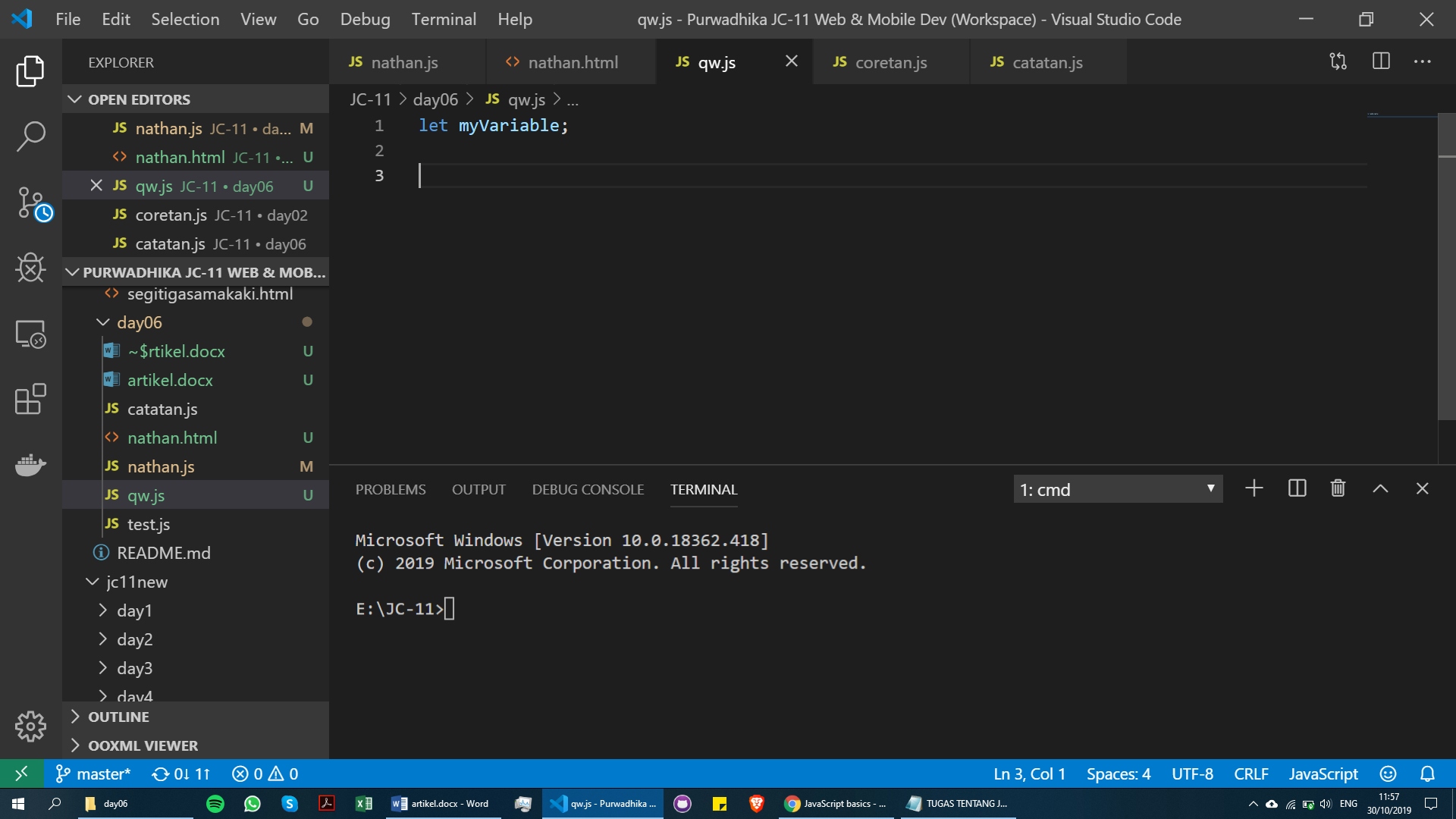
Dilihat dari grafik diatas, kita bisa ambil kesimpulan bahwa Bahasa Javascript merupakan salah satu bahasa pemrograman yang cenderung sangat mudah dipahami.

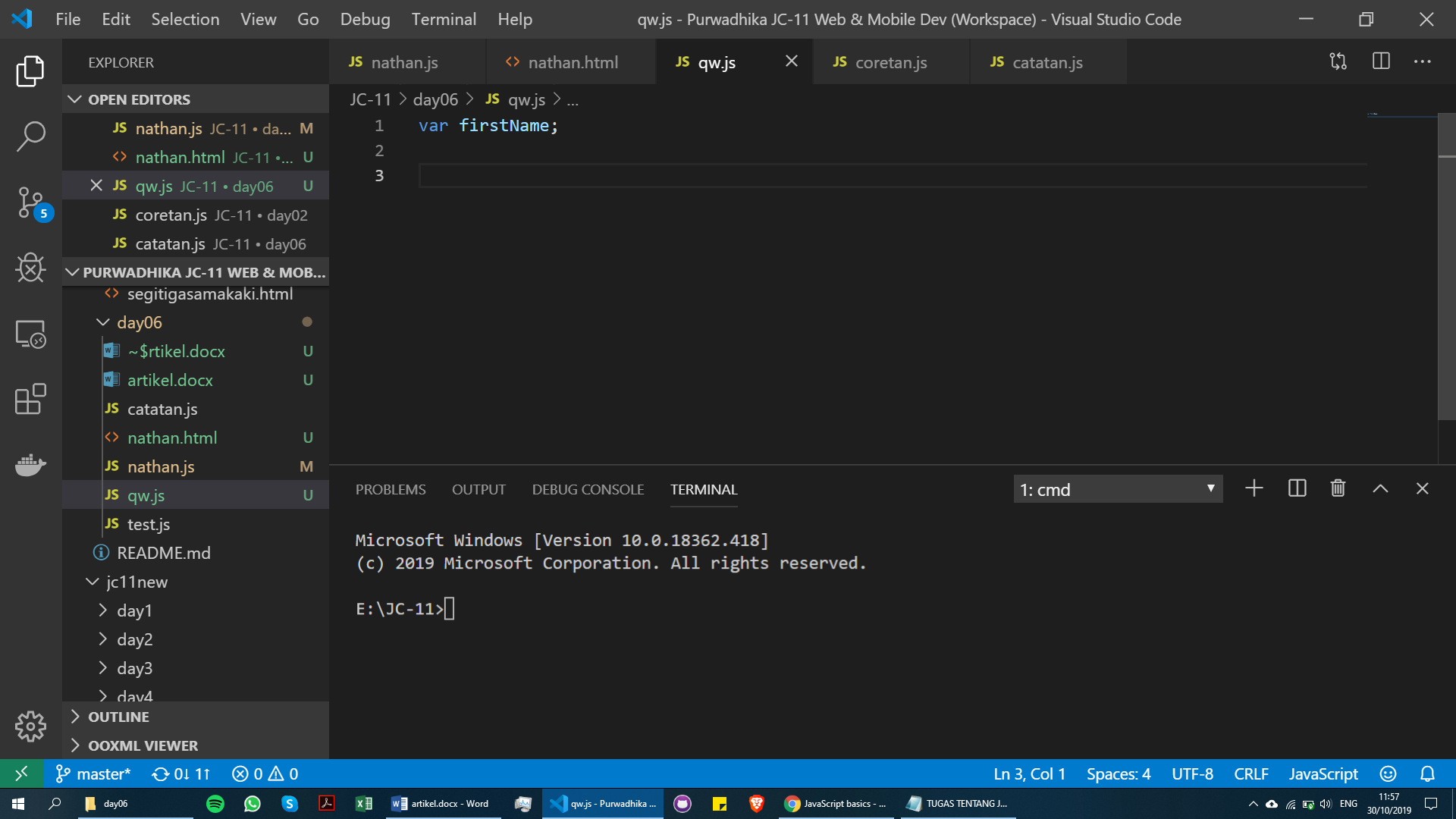
Dibawah ini adalah beberapa fitur Javascript yang umumnya juga terdapat didalam bahasa pemrograman lainnya, sehingga dengan pengetahuan *programming fundamental* yang baik, kita bisa dengan cepat untuk menguasai bahasa ini.

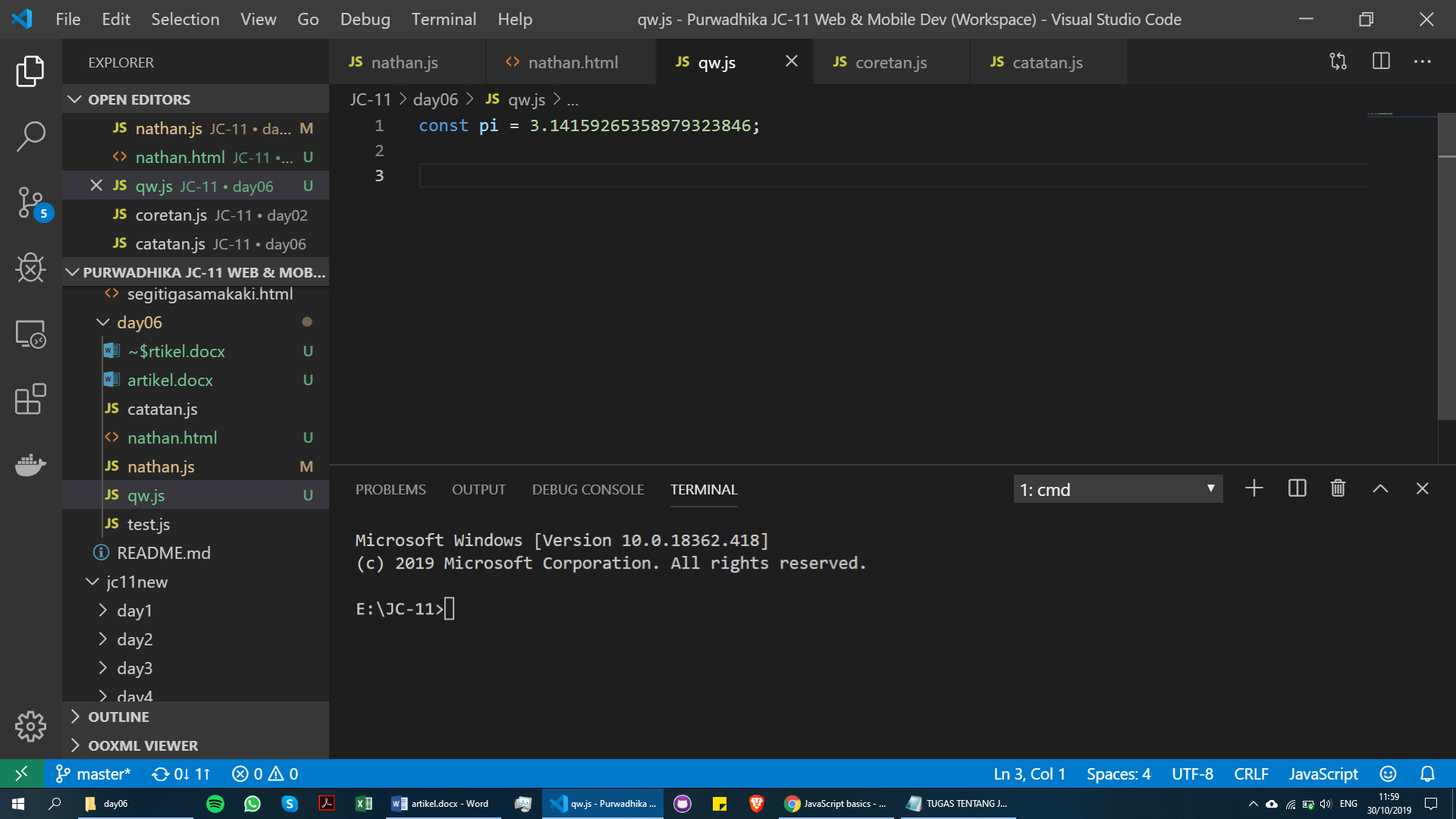
* Variabel

Variabel berfungsi sebagai tempat untuk menampung suatu *value* (nilai). Didalam ES6 atau Javascript 2015, deklarasi variabel dapat menggunakan var, let dan const.

Contoh pengunaannya:







var digunakan untuk menentukan variabel tanpa batasan *scope* atau cakupan. let digunakan pada variabel yang hanya digunakan pada *scope* tertentu seperti didalam block {} dan const digunakan untuk deklarasi variabel dengan nilai yang jarang diubah atau sangat sulit berubah di waktu yang akan datang seperti nilai konstanta.

Karena Javascript bersifat *Dynamic Typed*, Deklarasi variabel dalam JS bisa memiliki nilai dengan tipe data yang bermacam-macam seperti *String*, *Number*, *Boolean*, *Array*, dan *Object*.

* Operator

Operator dalam JS sama dengan bahasa pemrograman pada umumnya yaitu:

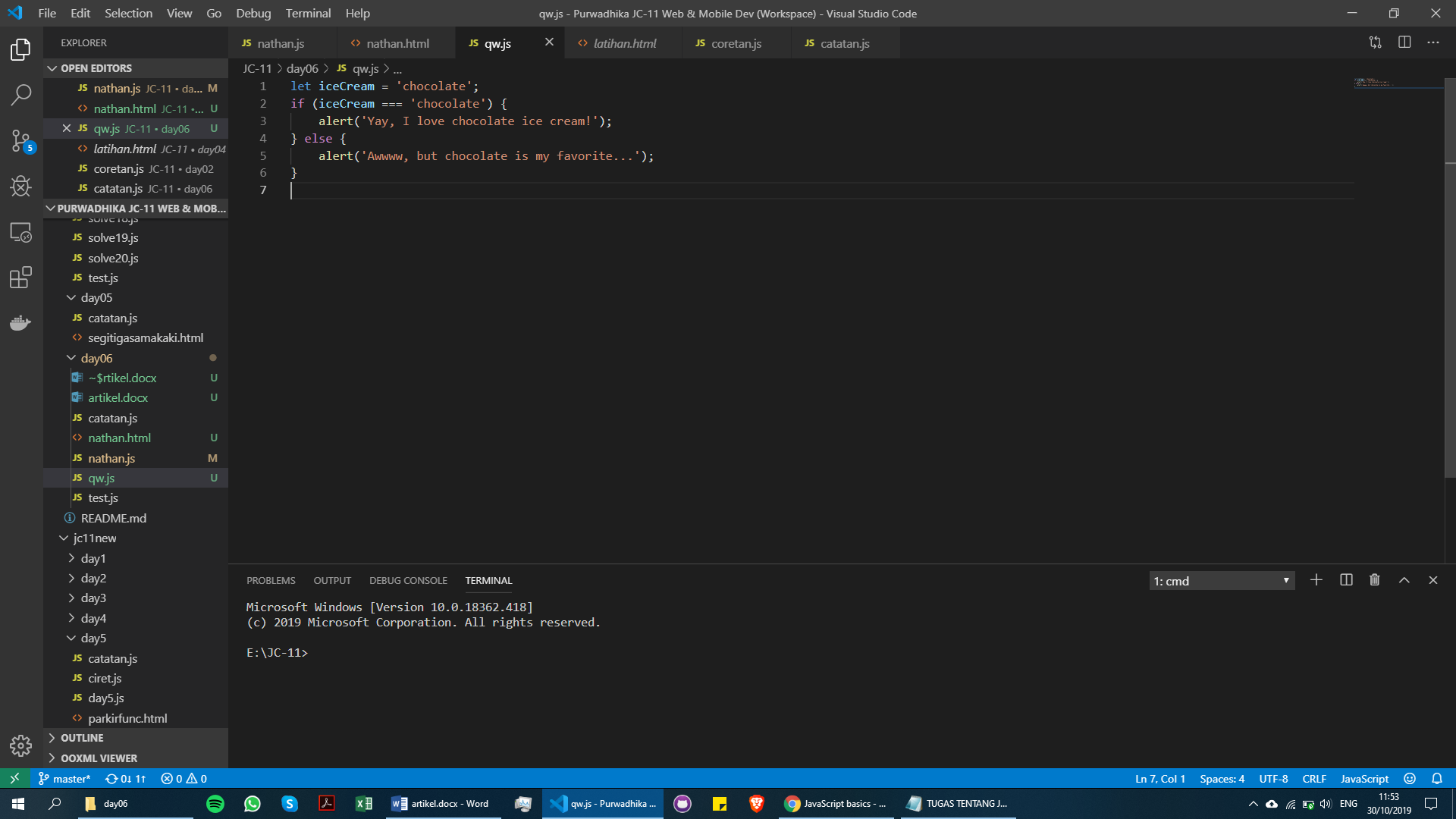
1. “ + ” untuk melakukan operasi penambahan dan juga berfungsi untuk menambahkan variabel satu dan lainnya.
2. “ − ” untuk melakukan operasi pengurangan
3. “ \* ” untuk operasi perkalian.
4. “ / ” untuk operasi pembagian.
5. “ = ” symbol *assignment*

Selain itu juga terdapat symbol “ ! ” yang artinya *Not* atau bentuk negasi dari statement tersebut.

* *Conditional*

*Conditional* merupakan struktur code yang digunankan untuk mengetes nilai *Boolean* dari suatu statement. Bentuk *conditional* yang paling sering digunakan adalah *if*, *else*, *else* *if.*

Contohnya:



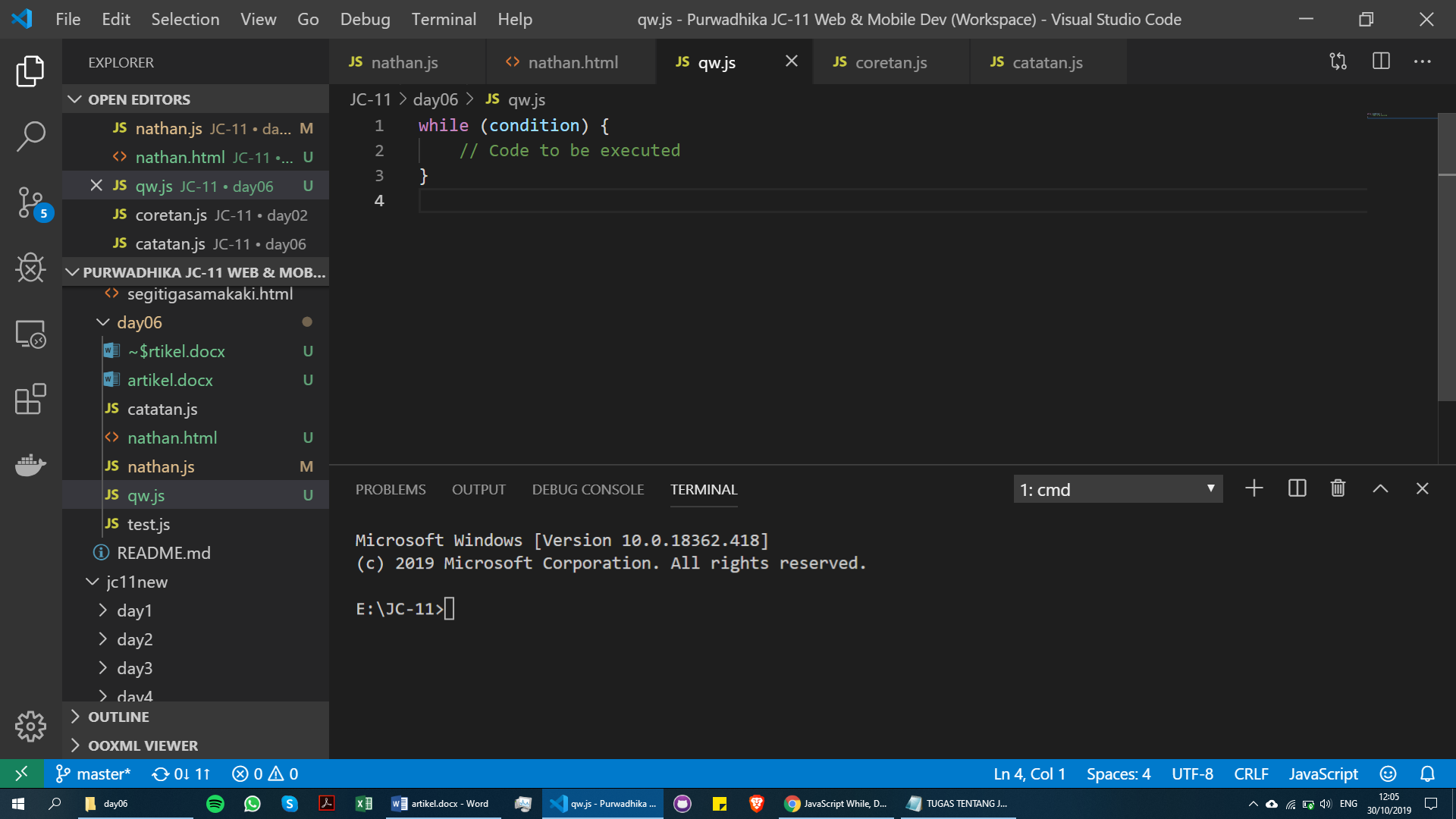
Code diatas adalah salah satu contoh penggunaan *if else* yang benar.

* *Loop*

*Loop* atau *Looping* adalah salah satu fitur yang digunakan untuk eksekusi program yang sifatnya berulang. Beberapa looping yang sering digunakan didalam Javascript adalah adalah:

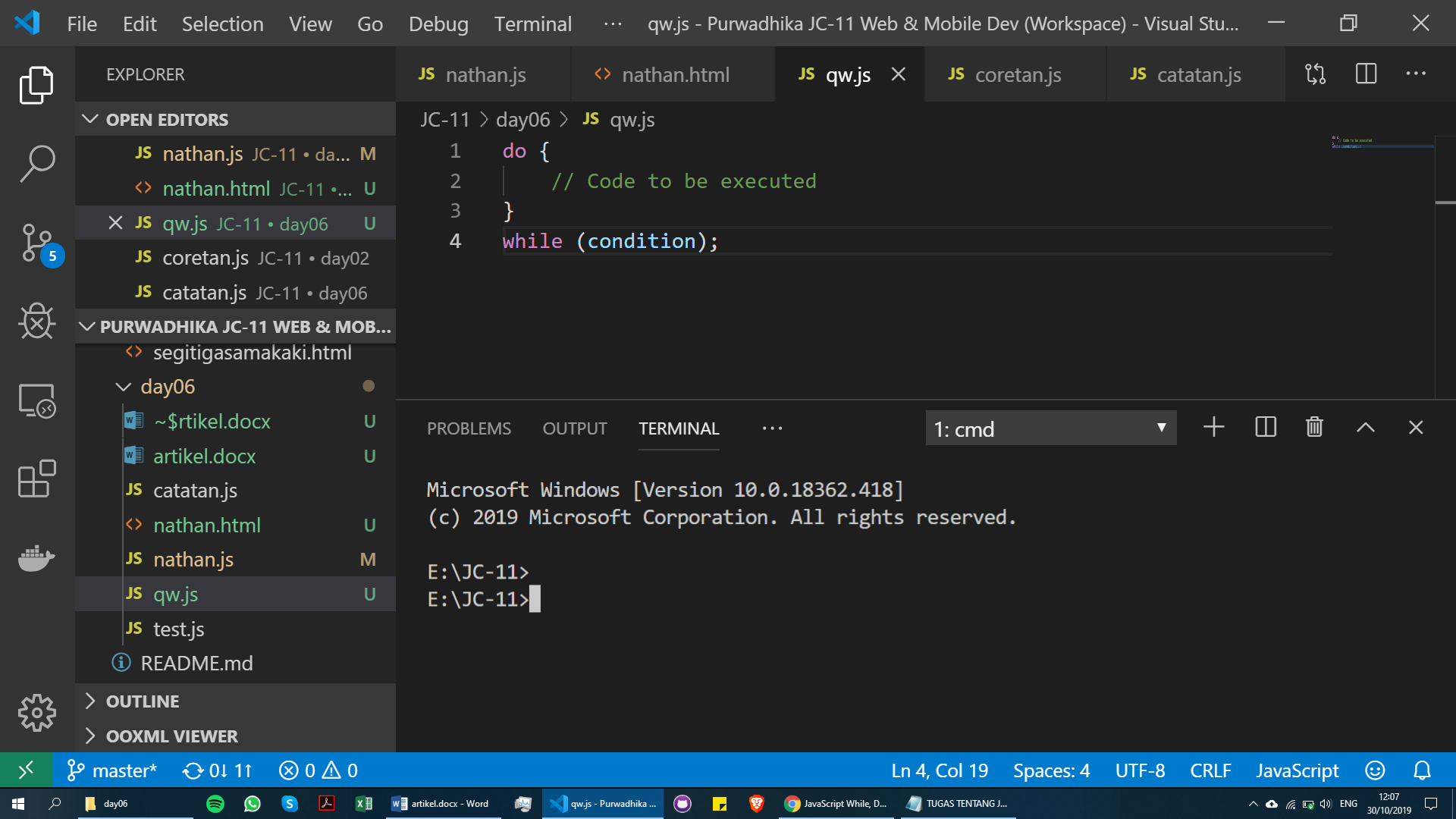
1. *While* *loop*

*While loop* merupakan bentuk paling simpel dari sebuah fungsi loop.While loop akan melakukan pengulangan eksekusi selama nilai *Boolean* dari *condition* tetap *True*.



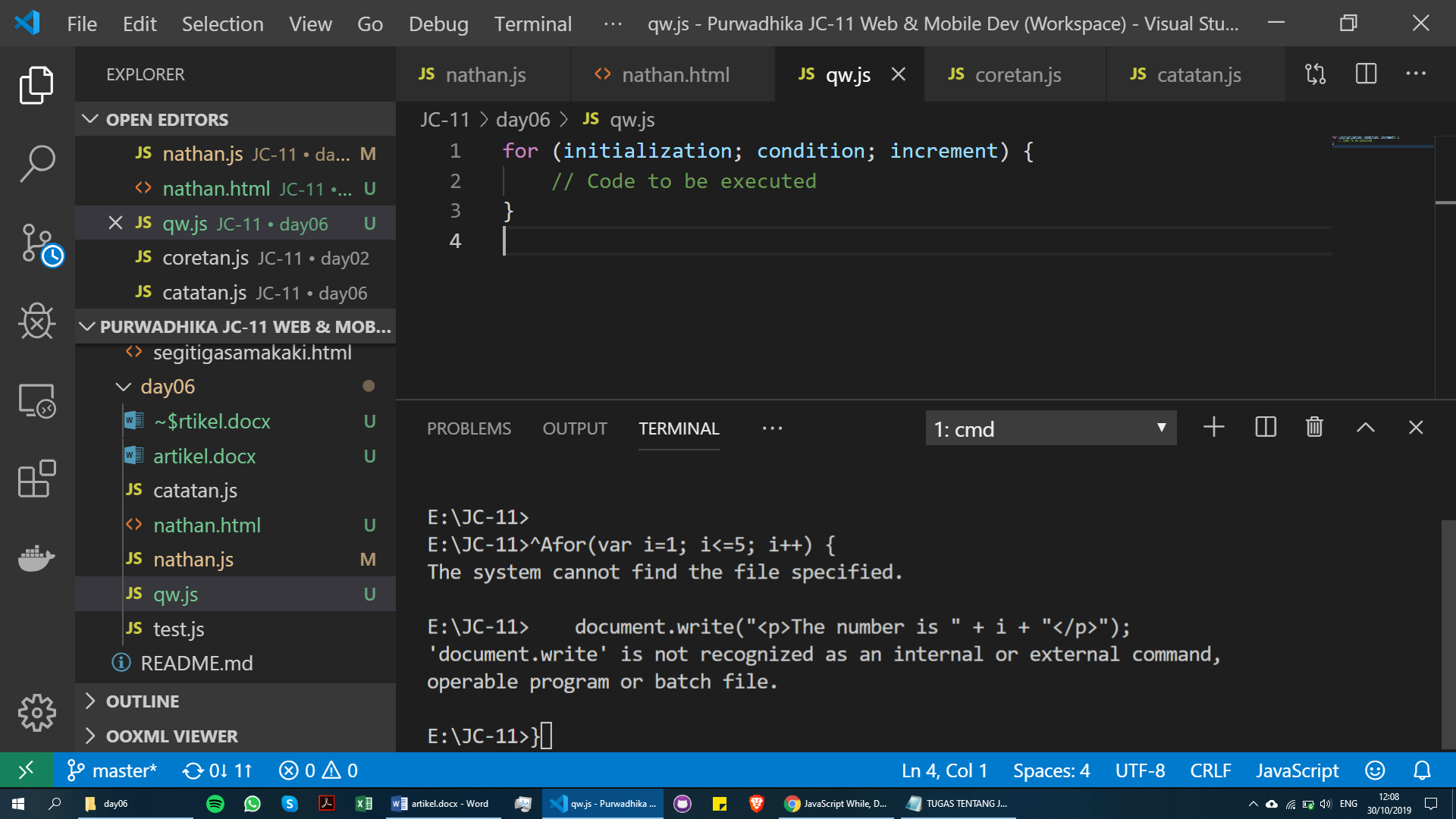
1. Do while loop

Berbeda dari *while* *loop*, evaluasi *Condition* dilakukan pada tahap akhir proses, sehingga memungkinkan setidaknya satu kali eksekusi walaupun *Condition* bernilai *Boolean* *false*



1. For loop

Loop ini digunakan untuk mengeksekusi code dengan beberapa kali pengulangan.



* *Function*

Function berisi beberapa set code yang bisa di panggil kapan saja untuk melakukan eksekusi code spesifik. Tujuan utama *function* sebagai jalan pintas. Alih alih mengetik ulang set code yang sudah pernah digunakan, cukup dengan satu set code, function yang baik bisa digunakan secara berulang.

