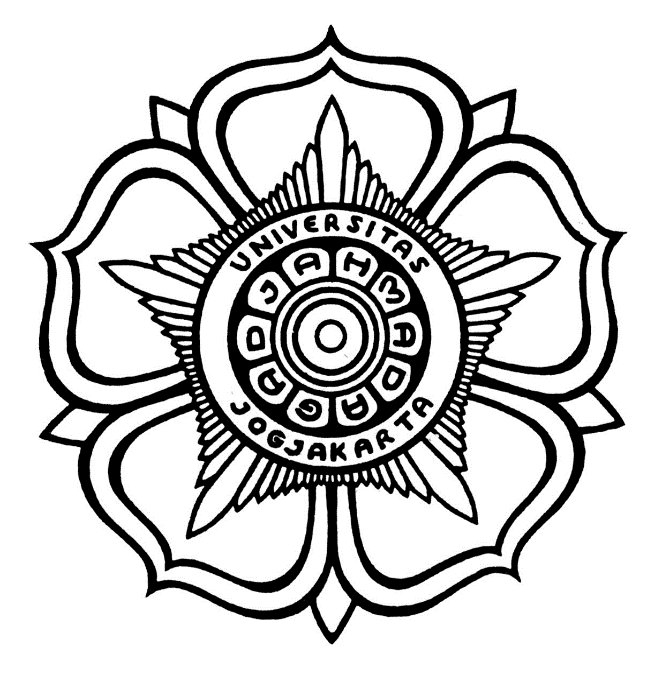
**LAPORAN PRAKTIKUM  
PEMROGRAMAN WEB III  
PERTEMUAN KE-4**



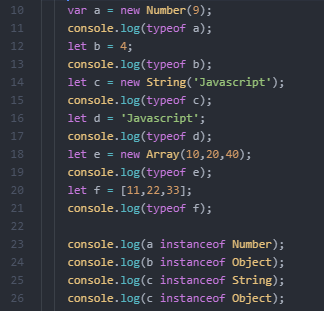
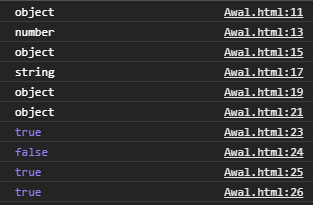
Oleh:  
Maulana Adam Sahid (18/431735/SV/15706)

**D3-KOMPUTER DAN SISTEM INFORMASI  
DEPARTEMEN TEKNIK ELEKTRO DAN INFORMATIKA  
SEKOLAH VOKASI – UNIVERSITAS GADJAH MADA  
2021**

**Laporan Praktikum PWEB3A 2021**

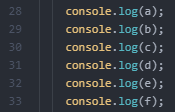
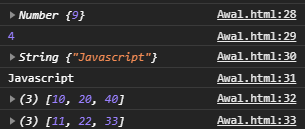
1. **Mengenali Tipe Objek Variabel**

Dalam Javascript terdapat kelas-kelas objek yang sudah disediakan secara default yaitu Number, String, dan Array. Dengan menggunakan kelas-kelas tersebut, maka suatu variabel tsb. akan dianggap sebagai variabel bertipe data object. Karena pada dasarnya kelas-kelas tersebut dalam Javascript adalah sebuah objek. Objek sendiri adalah suatu tipe data abstrak yang dibuat sendiri oleh developer yang didalamnya terdapat satu atau lebih properties maupun methods.

Dari contoh kode di atas dapat diketahui bahwa ada perbedaan variabel “a” dengan variabel “b” dan variabel “c” dengan “d”. Variabel “b” ketika dicheck tipenya ternyata ia adalah number sedangkan “a” ketika dicheck bertipe object padahal keduanya sama-sama diinputkan sebuah angka/number. Hal ini karena pada variabel “a” yang dinputkan adalah sebuah objek yang mengambil template dari kelas bernama “Number” terdahulu baru kemudian di dalam objek tersebut didefinisikan sebuah properti berupa angka. Sehingga variabel “a” memiliki tipe data sebagai objek yang didalamnya diisikan sebuah properti angka primitif, yaitu 9. Sedangkan pada variabel “b” angka/number 4 diinputkan secara langsung pada “b” sehingga “b” bukanlah bertipe objek melainkan number secara langsung. Karena ia diberikan value berupa angka primitif secara langsung.

Begitupula yang terjadi pada “c” dan “d” yang satu diperlakukan sebagai objek sedangkan yang lain lagi diperlakukan sebagai value primitif. Pada “c” kata “Javascript” ketika masuk ke dalam konstruktor dari kelas “String” dipecah tiap-tiap charnya kemudian disebar sebagai value ke tiap-tiap properties yang berurutan sesuai dengan indeks masing-masing char. Sedangkan pada variabel “d” kata “Javascript” diperlakukan sebagai value string biasa yang terdiri atas array of chars yang diurutkan oleh indeks secara langsung.

Sedangkan untuk kasus “e” dengan “f” ternyata tidaklah berbeda. Karena pada dasarnya ketika kita mendeklarasikan array seperti cara “f” itu sejatinya jugalah sedang membuat sebuah objek berkelas Array seperti pada cara pendeklarasian variabel “e”.

1. **Membuat Tipe Objek**

Dalam Javascript kita dapat membuat objek dengan dua cara yaitu dengan membuat fungsi pendefinisi objek sebagai template/blueprint atau dengan langsung mendeklarasikan objek di dalam suatu variabel.





*B.1 Contoh membuat dan menggunakan objek dengan mendefinisikan blueprintnya terlebih dahulu*





*B.2 Contoh membuat objek dengan langsung memasukkan dalam suatu variabel dan penggunaannya*

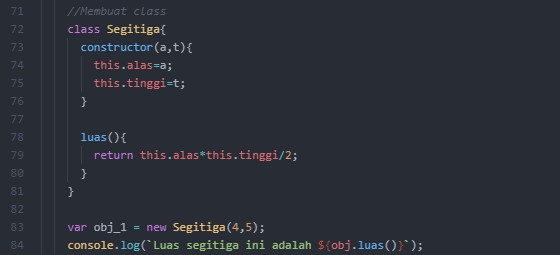
1. **Kelas**

Dalam ES6 kini Javascript sudah mendukung adanya pembuatan suatu kelas sebagai fasilitas untuk mendukung OOP. Sehingga kita dapat mengimplementasikan OOP pada Javascript sebagaimana OOP dalam bahasa pemrograman lainnya.

Pada Javascript ES6 kita dapat membuat kelas dengan menuliskan tipe fungsinya terlebih dahulu yaitu “class” kemudian diikuti dengan nama kelas yang ingin kita buat, misalnya “Segitiga”. Kemudian dilanjutkan dengan dua kurung kurawal pembuka dan penutupnya.

Untuk mendefinisikan konstruktornya, kita dapat mendefiniskan dengan mendefinisikan method “contructor” dikuti dengan kurung buka dan tutup yang di dalamnya berisi selarik parameter-parameter konstruktor kelas kemudian dikuti dengan kurung kurawal yang di dalamnya kita bisa definisikan atribut-atribut kelas tsb.

Sedangkan dalam membuat suatu method dalam kelas di ES6, kita dapat membuatnya cukup dengan mendefinisikan fungsi method tsb. dalam scope kelas yang kita buat tersebut setingkat/selevel dengan “constructor” pada kelas tersebut. Untuk lebih jelasnya dpat melihat pada tangkapan layar berikut.





Untuk membuat objek baru dengan kelas tersebut dan menggunakannya kita dapat melakukan seperti ketika kita membuat dan memanggil sebuah objek dengan cara pada screenshot B.1 sebelumnya.

1. **Pewarisan**

Pewarisan/inheritance adalah salah satu konsep dalam OOP yang mengizinkan kita untuk membuat suatu kelas dengan menwarisi kelas atribut dan method dari kelas lain. Kelas lain yang diwariskan atribut dan method disebut dengan super class (kelas induk). Sedangkan kelas baru yang mewarisi atribut dan method dari superclass itu disebut dengan subclass (kelas turunan).

Untuk menerapkan pewarisan dalam Javascript kita dapat dengan membuat kelas seperti biasa. Kemudian setelah menuliskan nama subclass mesti dilanjutkan dengan menggunakan kata kunci extends kemudian diikuti oleh nama superclass.





1. **Tugas Prakikum**
2. Membuat Kelas BangunDatar yang Mengakomodir Persegi Panjang dan Lingkaran

Dalam membuat kelas BangunDatar kita buat seperti kelas biasa pada umumnya dengan parameter konstruktornya adalah “tipe”, “a”, dan “b”. Kemudian di dalam konstruktornya diberikan terlebih dahulu percabangan untuk mengkategorikan argumen apakah yang dimasukkan dalam parameter “tipe” barulah kemudian di dalam kondisi-kondisi percabangan tersebutlah didefinisikan atribut-atribut suatu objek yang dibuat dengan kelas BangunDatar tersebut. Kemudian setelah menentukan pendefinisian atribut-atribut kelas dalam konstruktor, selanjutnya adalah mendefinisikan method getLuas() seperti contoh berikut.



1. Membuat Kelas Turunan dari Kelas BangunDatar dengan Nama BangunRuang Berisikan Volume Balok dan Tabung

Untuk membuat kelas BangunRuang yang mewarisi kelas BangunDatar kita dapat menuliskan kelas sebagaimana membuat kelas pada umumnya. Tetapi setelah pendefinisian nama kelas tersebut diikuti dengan keyword “extends” dan nama superclassnya, yaitu BangunDatar. Kemudian pada konstruktornya kita berikan tiga parameter seperti pada BangunDatar kemudian ditambah lagi satu parameter, yaitu “c”, untuk mendefinisikan atribut “tinggi” bangun kelas BangunRuang. Setelah itu di dalam operasi konstruktornya mesti dilakukan “super” terhadap variabel-variabel parameter yang ada dalam kelas BangunDatar dan juga pada kelas BangunRuang. Fungsi “super” adalah untuk melakukan overriding pada variabel-variabel parameter tersebut.



Sebagai testing kita bisa melakukan koding seperti berikut:



