

Jobsheet (JS) Pemrograman Web

Memahami Pemrograman Berorientasi Obyek

1. Tujuan

Mengetahui pemrograman berorientasi obyek pada web klien

Memahami penerapan pemrograman berorientasi obyek pada web klien

2. Tes Pengetahuan Awal

1. Soal 1

.....

.....

.....

.....

2. Soal 2

.....

.....

.....

.....

3. Soal 3

.....

.....

.....

.....

4. Soal 4

.....

.....

.....

.....

5. Soal 5

.....

.....

.....

.....

3. Alat dan Bahan

- a. PC/Laptop
- b. Aplikasi text editor (notepad++, sublime text, VS Code dll)
- c. Browser

4. Landasan Teori

Pemrograman Berorientasi Obyek (PBO)

Pemrograman berorientasi obyek adalah sebuah paradigma dalam pemrograman yang mengelompokkan fungsi, prosedur, variabel ataupun konstanta terkait dalam satu kesatuan (kelas) seperti obyek dalam dunia nyata. Obyek sendiri merupakan wujud dari kelas, secara umum dikenal dengan nama instan. Teknik-teknik dalam pemrograman ini merupakan pengembangan dari teknik sebelumnya seperti modularitas, polimorfisme, dan enkapsulasi. Saat ini ada banyak bahasa pemrograman yang telah mendukung pemrograman berorientasi obyek ini, seperti C++, C#, Java, Javascript, PHP, Python, Ruby, dan Objective C.

Dalam PBO setiap program akan menyatakan kumpulan dari obyek dan bukan lagi kumpulan fungsi ataupun kumpulan baris kode. Pada PBO ini setiap obyek memiliki kemampuan untuk menerima pesan, memproses data dan mengirim data ke obyek lainnya. Setiap obyek dapat dilihat sebagai program sendiri yang memiliki peran dan tanggungjawab masing-masing. Pengembangan program dengan menggunakan teknik PBO ini populer dalam program berskala besar karena fleksibel dan memudahkan dalam perbaikan.

PBO: Pemrograman Berbasis Prototipe

Javascript menggunakan PBO dengan model pemrograman berbasis prototipe. Model ini tidak mengenal kelas, yang ada adalah obyek. Obyek dalam model ini dapat dimodifikasi secara runtime seperti penambahan variabel (properti) ataupun metode (fungsi/prosedur). Model ini juga dikenal dengan istilah pemrograman PBO tanpa kelas, berorientasi prototipe atau berbasis instan.

Javascript Object Oriented Programming

Obyek Utama (Bawaan)

Javascript memiliki banyak obyek bawaan seperti Math, Array, String dan Object. Setiap obyek dalam Javascript adalah instan dari obyek Object, sehingga memiliki semua properti dan metode yang ada.

Obyek Buatan

Javascript merupakan bahasa PBO yang menggunakan kata kunci function untuk mendefinisikan suatu kelas. Oleh karena itu, pembuatan kelas sama dengan pembuatan fungsi dalam Javascript. Berikut ini merupakan format yang dapat digunakan untuk mendefinisikan kelas pada javascript.

```
function NamaKelas () {  
  }  
}
```

Atau dalam bentuk yang lebih lengkap

```
function NamaKelas (argumen1, argumen2, ..., argumenN)
{
    this.namavar1 = nilaiInisial;
    this.namavar2 = nilaiInisial;
    ...

    this.namaFungsil = function(parameter1, parameter2, ...,
parameterN){
        ...
    }
    ...
}
```

Hal yang dapat diperhatikan pada format diatas adalah `namavar1` dan `namavar2` merupakan properti dari kelas tersebut. Pembuatan properti dari suatu obyek dilakukan dengan menambahkan kata kunci `this` di awal dari nama properti tersebut, seperti diperlihatkan dalam format diatas. Sedangkan pembuatan metode atau fungsi dari obyek dilakukan dengan cara yang sama hanya dengan properti dengan menambahkan definisi fungsinya dengan menggunakan kata kunci `function`.

Pembuatan obyek dari kelas yang telah didefinisikan diatas dapat dilakukan dengan menggunakan format berikut.

```
var namaObyek = new NamaKelas ( ) ;
```

Kemudian untuk mengakses properti ataupun metode yang dimiliki oleh suatu obyek dapat dengan menggunakan operator titik (.) setelah nama obyek diikuti nama properti atau metodenya. Mengenai aturan penamaan kelas dapat mengikuti aturan penamaan fungsi pada materi sebelumnya

5. Langkah Kerja

- Buat folder di drive D:, dengan nama folder nama anda. Misal: `prakweb_smt2_nama`
- Buat subfolder `praktikum12` di dalam folder yang sudah dibuat sebelumnya.
- Buka aplikasi `notepad++` / `sublime text` sebagai web editor.
- Simpan elemen-elemen struktur dokumen HTML tersebut dalam format `.html` dan file javascript eksternal dengan ekstensi `.js`
- Jalankan masing-masing contoh elemen struktur HTML tersebut dengan menggunakan web browser.

Percobaan

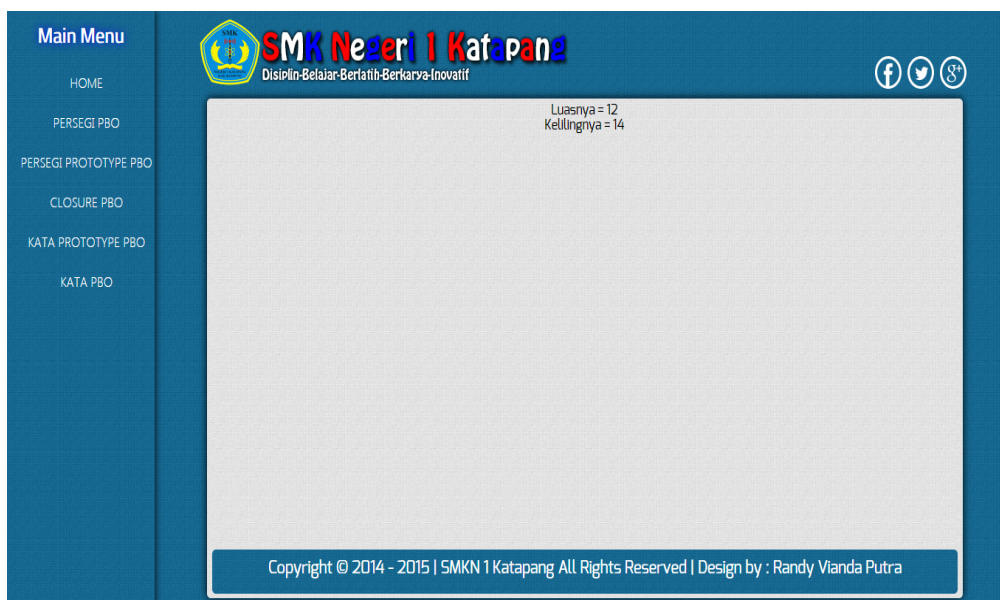
Contoh penerapan PBO.

Buatlah halaman web sebagai berikut:

```
<script>
function SegiEmpat(){
    this.panjang = 0;
    this.lebar = 0;
    this.hitungLuas = function(){
        var luas = this.panjang * this.lebar;
        return luas;
    }
    this.hitungKeliling = function(){
        var kel = 2 * (this.panjang + this.lebar);
        return kel;
    }
}
// contoh penggunaannya
var s = new SegiEmpat();
s.panjang = 4;
s.lebar = 3;
document.write("Luasnya = "+s.hitungLuas()+"<br>");
document.write("Kelilingnya = " +s.hitungKeliling());
</script>
```

Simpan file tersebut dengan nama persegi_pbo.html

Hasilnya :



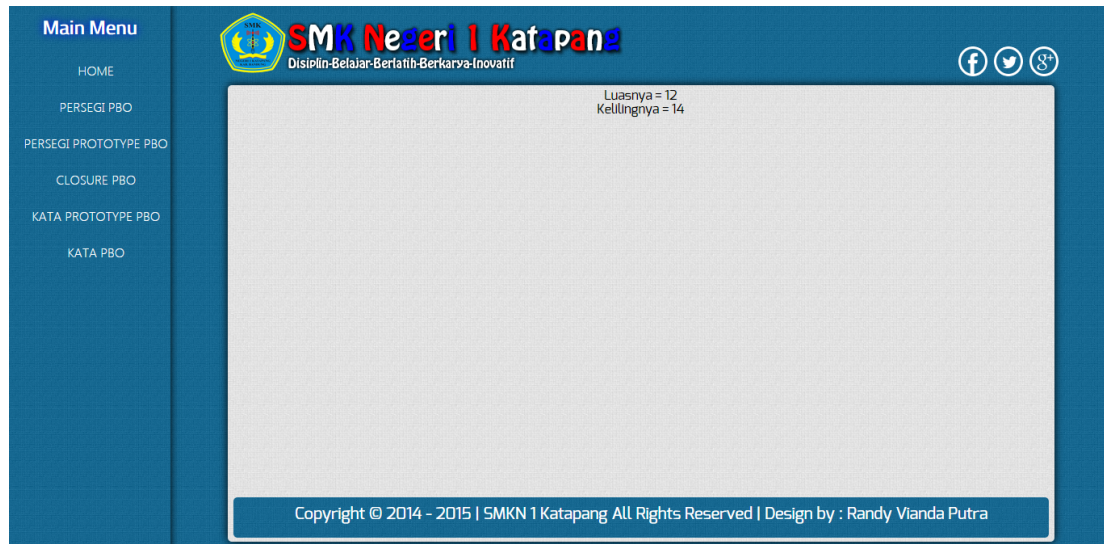
Contoh menghitung luas dan keliling persegi menggunakan prototype :

Buatlah halaman web sebagai berikut:

```
<script>
function SegiEmpat(){}
SegiEmpat.prototype.panjang = 0;
SegiEmpat.prototype.lebar = 0;
SegiEmpat.prototype.hitungLuas = function(){
    var luas = this.panjang * this.lebar;
    return luas;
}
SegiEmpat.prototype.hitungKeliling = function(){
    var kel = 2 * (this.panjang + this.lebar);
    return kel;
}
// contoh penggunaannya
var s = new SegiEmpat();
s.panjang = 4;
s.lebar = 3;
document.write("Luasnya = "+s.hitungLuas()+"<br>");
document.write("Kelilingnya = " +s.hitungKeliling());
</script>
```

Simpan file tersebut dengan nama persegi_proto_pbo.html

Hasilnya:

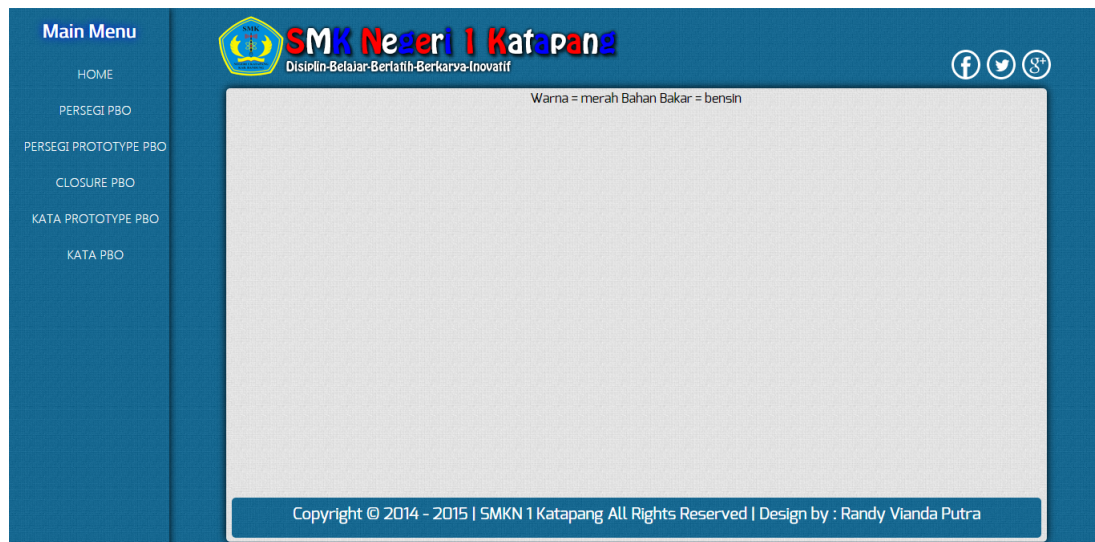


Contoh lainnya :

```
<script language="javascript" type="text/javascript">
function mobil(wrn, bb){
    this.warna = wrn;
    this.bahanBakar = bb;
}
var h = new mobil("merah", "bensin");
document.write("Warna = "+h.warna+" Bahan Bakar = "+h.bahanBakar);
</script>
```

Simpan file tersebut dengan nama closure_pbo.html

Hasilnya:



Contoh lainnya membuat kata dengan prototype :

```
<script language="javascript" type="text/javascript">
function mobil(wrn, bb){
    this.warna = wrn;
    this.bahanBakar = bb;
}
mobil.prototype.merk = "Lamborghini";
var h = new mobil("merah", "bensin");
document.write("Mobil dengan merk "+h.merk+" Memiliki Warna "+h.warna+" dan menggunakan bahan bakar"+h.
    bahanBakar);
</script>
```


Simpan file tersebut dengan nama kata_proto_pbo.html

Hasilnya:



Contoh lainnya membuat kata tanpa prototype :

```
<script language="javascript" type="text/javascript">
function mobil(wrn, bb){
    this.warna = wrn;
    this.bahanBakar = bb;
    this.tampil = function(){
        return "Warna mobil ini = " + this.warna + " dan bahan bakarnya = " + this.bahanBakar;
    }
}
var h = new mobil("merah", "bensin");
document.write(h.tampil());
</script>
```

Simpan file tersebut dengan nama kata_pbo.html

Hasilnya:



6. Tugas

- 1.) Buatlah program cara menghitung volume limas segi empat menggunakan OOP.
- 2.) Buatlah program seperti yang telah disebutkan diatas dengan tampilan semenarik mungkin.

Referensi

Buku Teks Pemrograman Web

www.w3schools.com.