Nama : Muh. Nur Afrizal
NIM : 222212738
Kelas / No : 2KS1 / 25

MODUL 5

PENGENALAN JAVASCRIPT

PEMROGRAMAN BERBASIS WEB

5.4.1. Deklarasi Variabel

jsDemo06A.html: https://github.com/afrzl/Pemrograman-Berbasis-Web/blob/main/P5/jsDemo06A.html

```
<script id="s1">
      // Script with id s1
    </script>
    <p>Our first JavaScript script</p>
    <script id="s2">
      // Script with id s2
      user = 'Afrizal'
      document.writeln('<p><b>Hello ' + user + '<br>\nHello
World!</b></p>')

      document.writeln('(1) The value of userAge is: ' + userAge
+ '<br>')
      document.writeln('(2) This statement is executed<br>')
    </script>
    <script id="s3">
      // Script with id s3
      var userAge = '27'
    </script>
```

7. Apakah kode JavaScript sekarang berjalan dengan tanpa error, atau ada error? Apa kesimpulan yang bisa diambil dalam kaitannya antara deklarasi variabel dengan cakupan variabel?

   Jawab: Terdapat error, karena javascript membaca code dari atas ke bawah tiap line, dan s2 diatas s3 pada posisinya. Sehingga, deklarasi variable userAge belum terbaca ketika di s2.

```
<script id="s1">
      // Script with id s1
      var userAge = '27'
    </script>
    <p>Our first JavaScript script</p>
    <script id="s2">
      // Script with id s2
      user = 'Afrizal'
```

Nama : Muh. Nur Afrizal
NIM : 222212738
Kelas / No : 2KS1 / 25

```
      document.writeln('<p><b>Hello ' + user + '<br>\nHello
World!</b></p>')

      document.writeln('(1) The value of userAge is: ' +
userAge + '<br>')
      document.writeln('(2) This statement is executed<br>')
    </script>
    <script id="s3">
      // Script with id s3
    </script>
```

8a. Apakah kode JavaScript sekarang berjalan dengan tanpa error, atau ada error? Apakah hasilnya sama dengan butir 7? Kalau tidak, mengapa?

Jawab: Tidak terdapat error, karena variabel userAge sudah dideklarasikan di s1 dimana posisi s1 diatas s2 sehingga pada saat dipanggil variabel tersebut sudah dideclare.

8b. Apa ke simpulan yang bisa diambil dalam kaitannya antara deklarasi variabel dengan cakupan variabel?

Jawab: Javascript membaca program dari atas sampai bawah, walaupun berada pada tag script yang berbeda.

5.4.2. Error Handling

jsDemo06B.html: https://github.com/afrzl/Pemrograman-Berbasis-Web/blob/main/P5/jsDemo06B.html

5.4.3. Sifat Bilangan Pecahan

jsDemo06C.html: https://github.com/afrzl/Pemrograman-Berbasis-Web/blob/main/P5/jsDemo06C.html

x = 0.09999999999999998
y = 0.1

3. Mengapa terdapat perbedaan pada hasil operasi 0.3 - 0.2 dan 0.2-0.1?

Jawab: Hal ini karena pada computer, bilangan desimal tersebut akan diconvert terlebih dahulu ke 64bit floating point. Sehingga hal tersebut memungkinkan adanya chopping di sistem komputer ketika bit terlalu besar (biasanya terjadi di bilangan irasional).

1.275 and 1.275 are equal

Nama : Muh. Nur Afrizal
NIM : 222212738
Kelas / No : 2KS1 / 25

6. Walaupun pernyataan di atas benar, namun ada yang aneh terjadi pada nilai dari variabel y. Apa yang terjadi?

Jawab: Karena sebenarnya kedua bilangan tersebut akan diconvert terlebih dahulu ke 64bit floating point terlebih dahulu. Dan jika diconvert, kedua bilangan tersebut akan sama sehingga menghasilkan equal.

5.4.4. Operasi Matematika

jsDemo07A.html: https://github.com/afrzl/Pemrograman-Berbasis-Web/blob/main/P5/jsDemo07A.html

jsDemo07A.js: https://github.com/afrzl/Pemrograman-Berbasis-Web/blob/main/P5/jsDemo07A.js

4. Ubah kode JavaScript agar menampilkan hanya tiga digit di belakang koma, seperti berikut.

Jawab:

```
for (var ii = 4, jj = 3; jj >= 0; ii++, jj--) {
  document.writeln(ii + " * " + jj + " = " + (ii *
jj).toFixed(3) + "<br>");
  document.writeln(ii + " / " + jj + " = " + (ii /
jj).toFixed(3) + "<br>");
  document.writeln("log(" + jj + ") = " +
Math.log(jj).toFixed(3) + "<br>");
  document.writeln("sqrt(" + (jj - 1) + ") = " +
Math.sqrt(jj - 1).toFixed(3) + "<br><br>");
}
```

5.4.5. Operasi Logika

jsDemo07C.html: https://github.com/afrzl/Pemrograman-Berbasis-Web/blob/main/P5/jsDemo07C.html

3. Mengapa ‘undefined’ dan ‘null’ ditampilkan menjadi string kosong Lihat https://262.ecma international.org/9.0/, Bab 22.1.3.13, untuk memahami cara kerja fungsi join.

Jawab: Karena ‘undefined’ dan ‘null’ jika diconvert ke dalam string maka akan berubah menjadi string kosong.

Nama : Muh. Nur Afrizal

NIM : 222212738

Kelas / No : 2KS1 / 25

4. Kembangkan kode JavaScript di atas dengan menambahkan kode program yang membandingkan tiap elemen dengan tiap elemen yang lain menggunakan empat operator, yaitu ==, ===, > dan <.

Jawab:

```
      for (i = 0; i < values.length; i++)
        for (j = i; j < values.length; j++) {
          document.writeln(
            '(' +
              typeof values[i] +
              ') ' +
              values[i] +
              ' == (' +
              typeof values[j] +
              ') ' +
              values[j] +
              ' : ' +
              (values[i] == values[j]) +
              '<br>',
          )
          document.writeln(
            '(' +
              typeof values[i] +
              ') ' +
              values[i] +
              ' === (' +
              typeof values[j] +
              ') ' +
              values[j] +
              ' : ' +
              (values[i] === values[j]) +
              '<br>',
          )
          document.writeln(
            '(' +
              typeof values[i] +
              ') ' +
              values[i] +
              ' < (' +
              typeof values[j] +
              ') ' +
              values[j] +
              ' : ' +
              (values[i] < values[j]) +
              '<br>',
```

Nama : Muh. Nur Afrizal
NIM : 222212738
Kelas / No : 2KS1 / 25

```
        )
        document.writeln(
          '(' +
            typeof values[i] +
            ') ' +
            values[i] +
            ' > (' +
            typeof values[j] +
            ') ' +
            values[j] +
            ' : ' +
            (values[i] > values[j]) +
            '<br>',
        )
        document.writeln('<br>')
      }
```

Output:

(number) 0 == (number) 0 : true
(number) 0 === (number) 0 : true
(number) 0 < (number) 0 : false
(number) 0 > (number) 0 : false

(number) 0 == (number) 123 : false
(number) 0 === (number) 123 : false
(number) 0 < (number) 123 : true
(number) 0 > (number) 123 : false

(number) 0 == (number) 123 : false
(number) 0 === (number) 123 : false
(number) 0 < (number) 123 : true
(number) 0 > (number) 123 : false

(number) 0 == (string) 12.3e1 : false
(number) 0 === (string) 12.3e1 : false
(number) 0 < (string) 12.3e1 : true
(number) 0 > (string) 12.3e1 : false

(number) 0 == (boolean) true : false
(number) 0 === (boolean) true : false
(number) 0 < (boolean) true : true
(number) 0 > (boolean) true : false

Nama : Muh. Nur Afrizal

NIM : 222212738

Kelas / No : 2KS1 / 25

5. Ubah kode pada latihan sebelumnya pada butir d agar hasilnya ditampilkan dalam bentuk tabel menggunakan elemen tabel HTML.

Jawab:

```
document.writeln('<table>' + '<tbody>')
for (var i = 0; i < values.length; i++) {
  for (j = i; j < values.length; j++) {
    document.writeln(
      '<tr><td>(' +
        typeof values[i] +
        ') ' +
        values[i] +
        ' == (' +
        typeof values[j] +
        ') ' +
        values[j] +
        ' : ' +
        (values[i] == values[j]) +
        '</td>',
    )
    document.writeln(
      '<td>(' +
        typeof values[i] +
        ') ' +
        values[i] +
        ' === (' +
        typeof values[j] +
        ') ' +
        values[j] +
        ' : ' +
        (values[i] === values[j]) +
        '</td>',
    )
    document.writeln(
      '<td>(' +
        typeof values[i] +
        ') ' +
        values[i] +
        ' < (' +
        typeof values[j] +
        ') ' +
        values[j] +
        ' : ' +
        (values[i] < values[j]) +
        '</td>',
```

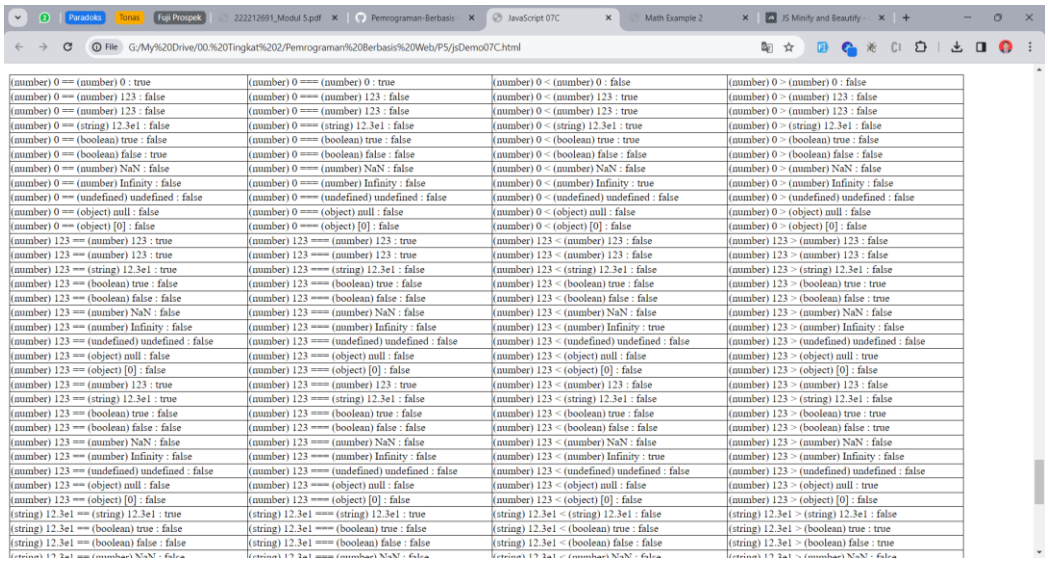
Nama : Muh. Nur Afrizal

NIM : 222212738

Kelas / No : 2KS1 / 25

```
        )
        document.writeln(
          '<td>(' +
            typeof values[i] +
            ') ' +
            values[i] +
            ' > (' +
            typeof values[j] +
            ') ' +
            values[j] +
            ' : ' +
            (values[i] > values[j]) +
            '</td></tr>',
        )
      }
    }
    document.writeln('</tbody></table>')
```

Output:



(number) 0 == (number) 0 : true	(number) 0 === (number) 0 : true	(number) 0 < (number) 0 : false	(number) 0 > (number) 0 : false
(number) 0 == (number) 123 : false	(number) 0 === (number) 123 : false	(number) 0 < (number) 123 : true	(number) 0 > (number) 123 : false
(number) 0 == (number) 123 : false	(number) 0 === (number) 123 : false	(number) 0 < (number) 123 : true	(number) 0 > (number) 123 : false
(number) 0 == (string) 12.3e1 : false	(number) 0 === (string) 12.3e1 : false	(number) 0 < (string) 12.3e1 : true	(number) 0 > (string) 12.3e1 : false
(number) 0 == (boolean) true : false	(number) 0 === (boolean) true : false	(number) 0 < (boolean) true : true	(number) 0 > (boolean) true : false
(number) 0 == (boolean) false : true	(number) 0 === (boolean) false : false	(number) 0 < (boolean) false : false	(number) 0 > (boolean) false : true
(number) 0 == (number) NaN : false	(number) 0 === (number) NaN : false	(number) 0 < (number) NaN : false	(number) 0 > (number) NaN : false
(number) 0 == (number) Infinity : false	(number) 0 === (number) Infinity : false	(number) 0 < (number) Infinity : true	(number) 0 > (number) Infinity : false
(number) 0 == (undefined) undefined : false	(number) 0 === (undefined) undefined : false	(number) 0 < (undefined) undefined : false	(number) 0 > (undefined) undefined : false
(number) 0 == (object) null : false	(number) 0 === (object) null : false	(number) 0 < (object) null : false	(number) 0 > (object) null : false
(number) 0 == (object) [0] : false	(number) 0 === (object) [0] : false	(number) 0 < (object) [0] : false	(number) 0 > (object) [0] : false
(number) 123 == (number) 123 : true	(number) 123 === (number) 123 : true	(number) 123 < (number) 123 : false	(number) 123 > (number) 123 : false
(number) 123 == (number) 123 : true	(number) 123 === (number) 123 : true	(number) 123 < (number) 123 : false	(number) 123 > (number) 123 : false
(number) 123 == (string) 12.3e1 : true	(number) 123 === (string) 12.3e1 : false	(number) 123 < (string) 12.3e1 : false	(number) 123 > (string) 12.3e1 : false
(number) 123 == (boolean) true : false	(number) 123 === (boolean) true : false	(number) 123 < (boolean) true : false	(number) 123 > (boolean) true : true
(number) 123 == (boolean) false : false	(number) 123 === (boolean) false : false	(number) 123 < (boolean) false : false	(number) 123 > (boolean) false : true
(number) 123 == (number) NaN : false	(number) 123 === (number) NaN : false	(number) 123 < (number) NaN : false	(number) 123 > (number) NaN : false
(number) 123 == (number) Infinity : false	(number) 123 === (number) Infinity : false	(number) 123 < (number) Infinity : true	(number) 123 > (number) Infinity : false
(number) 123 == (undefined) undefined : false	(number) 123 === (undefined) undefined : false	(number) 123 < (undefined) undefined : false	(number) 123 > (undefined) undefined : false
(number) 123 == (object) null : false	(number) 123 === (object) null : false	(number) 123 < (object) null : false	(number) 123 > (object) null : true
(number) 123 == (object) [0] : false	(number) 123 === (object) [0] : false	(number) 123 < (object) [0] : false	(number) 123 > (object) [0] : false
(number) 123 == (number) 123 : true	(number) 123 === (number) 123 : true	(number) 123 < (number) 123 : false	(number) 123 > (number) 123 : false
(number) 123 == (string) 12.3e1 : true	(number) 123 === (string) 12.3e1 : false	(number) 123 < (string) 12.3e1 : false	(number) 123 > (string) 12.3e1 : false
(number) 123 == (boolean) true : false	(number) 123 === (boolean) true : false	(number) 123 < (boolean) true : false	(number) 123 > (boolean) true : true
(number) 123 == (boolean) false : false	(number) 123 === (boolean) false : false	(number) 123 < (boolean) false : false	(number) 123 > (boolean) false : true
(number) 123 == (number) NaN : false	(number) 123 === (number) NaN : false	(number) 123 < (number) NaN : false	(number) 123 > (number) NaN : false
(number) 123 == (number) Infinity : false	(number) 123 === (number) Infinity : false	(number) 123 < (number) Infinity : true	(number) 123 > (number) Infinity : false
(number) 123 == (undefined) undefined : false	(number) 123 === (undefined) undefined : false	(number) 123 < (undefined) undefined : false	(number) 123 > (undefined) undefined : false
(number) 123 == (object) null : false	(number) 123 === (object) null : false	(number) 123 < (object) null : false	(number) 123 > (object) null : true
(number) 123 == (object) [0] : false	(number) 123 === (object) [0] : false	(number) 123 < (object) [0] : false	(number) 123 > (object) [0] : false
(string) 12.3e1 == (string) 12.3e1 : true	(string) 12.3e1 === (string) 12.3e1 : true	(string) 12.3e1 < (string) 12.3e1 : false	(string) 12.3e1 > (string) 12.3e1 : false
(string) 12.3e1 == (boolean) true : false	(string) 12.3e1 === (boolean) true : false	(string) 12.3e1 < (boolean) true : false	(string) 12.3e1 > (boolean) true : true
(string) 12.3e1 == (boolean) false : false	(string) 12.3e1 === (boolean) false : false	(string) 12.3e1 < (boolean) false : false	(string) 12.3e1 > (boolean) false : true