

JAVASCRIPT

JavaScript adalah bahasa pemrograman tingkat tinggi yang digunakan terutama untuk membuat halaman web interaktif. Ini adalah salah satu teknologi penting dalam pengembangan web, selain HTML dan CSS. JavaScript memungkinkan developer menambahkan fitur seperti animasi, validasi formulir, dan interaksi dinamis lainnya ke dalam halaman web

Latihan 1, Objek Javascript Alert

```
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Javascript</title>
</head>
<body>
  <button onclick="showAlert()">Object Alert</button>
  <script type="text/javascript">
    function showAlert() {
      alert('ini alert yaa!');
    }
  </script>
</body>
</html>
```

Latihan 2 , Objek Javascript Prompt

```
<html>
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Javascript</title>
</head>
<body>
  <button onclick="showPrompt()">Object Prompt</button>
  <script type="text/javascript">
    function showPrompt() {
      var prm = prompt('Masukan Nama Anda');
      alert("Nama Anda Adalah = " + prm)
    }
  </script>
</body>
</html>
```

Latihan3, Object Javascript Confirm

```
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Javascript</title>
</head>
<body>
  <button onclick="showConfirm()">Object Confirm</button>
  <script type="text/javascript">
    function showConfirm() {
      var cnf = confirm('Apakah Anda Menyukai Kehidupan Anda Sebagai Manusia ??')
      if(cnf){
        alert(cnf);
      }else{
        alert(cnf);
      }
    }
  </script>
</body>
</html>
```

Latihan 4, Membuat Pop Up Objeck Javascript dan CSS

```
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Javascript</title>
  <style>
    #body {
      background-color: white;
    }

    #popup {
      width: 400px;
      height: 300px;
      margin: auto;
      background-color: white;
      color: black;
      display: none;
      position: relative;
    }
  </style>
</head>
<body>
  <div id="popup">
    <div id="close">X</div>
    <div id="text">
      <p>Apakah Anda Menyukai Kehidupan Anda Sebagai Manusia ??</p>
      <div id="yes">Ya</div>
      <div id="no">Tidak</div>
    </div>
  </div>
</body>
</html>
```

```

        box-shadow: 0px 5px 5px 5px #000;
    }
</style>
</head>
<body id="body">
    <div id="popup">
        <h3 align="center">Pop Up</h3>
        <hr>
        <button onclick="closePopUp()" id="btns">Close PopUp</button>
    </div>
    <button onclick="showPopUp()" id="btn">ObjectPopUp</button>
    <script type="text/javascript">
        function showPopUp() {
            document.getElementById("body").style.backgroundColor = 'dedede';
            document.getElementById("popup").style.display = 'block';
            document.getElementById("btn").style.display = 'none';
        }
        function closePopUp() {
            document.getElementById("body").style.backgroundColor = 'white';
            document.getElementById("popup").style.display = 'none';
            document.getElementById("btn").style.display = 'block';
        }
    </script>
</body>
</html>

```

Latihan 5 , Menampilkan Grafik dengan Vanilla js dan CSS

```

<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Grafik Batang dengan CSS dan Vanilla JavaScript</title>
    <style>
        .chart-container {
            display: flex;
            align-items: flex-end;
            width: 100%;
            height: 300px;
            border-bottom: 2px solid #333;
            padding-bottom: 30px;

```

```

}

.bar {
  background-color: steelblue;
  color: white;
  margin: 0 20px;
  width: 55px;
  display: flex;
  flex-direction: column;
  justify-content: flex-end;
  align-items: center;
  position: relative;
}

.bar::before {
  content: attr(data-label);
  position: absolute;
  bottom: -20px;
  font-size: 0.8em;
  color: #555;
}

.bar::after {
  content: '';
  position: absolute;
  top: 0;
  left: 0;
  width: 100%;
  background-color: #000;
}
</style>
</head>
<body>
<div class="chart-container">
  </div>

<script>
  const chartContainer = document.querySelector('.chart-container');
  const data = [75, 90, 50, 80, 65, 90, 75, 55];
  const labels = ['Jan', 'Feb', 'Mar', 'Apr', 'Mei', 'Jun', 'Jul', 'Aug'];
  chartContainer.innerHTML = '';

```

```

    data.forEach((value, index) => {
        const bar = document.createElement('div');
        bar.classList.add('bar');
        bar.style.height = `${value}%`;
        bar.dataset.label = labels[index] + ',' + value + '%';
        chartContainer.appendChild(bar);
    });
</script>
</body>
</html>

```

Latihan 6, membuat jam digital dengan objek javascript

```

<!DOCTYPE html>
<html>
<head>
    <title>Jam Sederhana</title>
</head>
<body>
    <h1 id="jam"></h1>

    <script>
        function tampilkanWaktu() {
            const waktuSekarang = new Date();
            const jam = waktuSekarang.getHours().toString().padStart(2, '0');
            const menit = waktuSekarang.getMinutes().toString().padStart(2, '0');
            const detik = waktuSekarang.getSeconds().toString().padStart(2, '0');
            const waktuString = `${jam}:${menit}:${detik}`;
            document.getElementById('jam').textContent = waktuString;
        }
        setInterval(tampilkanWaktu, 1000);
    </script>
</body>
</html>

```

Latihan 7, Membuat Kalender Sederhana Dengan Objek Javascript

```
<!DOCTYPE html>
<html lang="id">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Kalender Sederhana</title>
  <style>
    body {
      font-family: sans-serif;
      display: flex;
      justify-content: center;
      align-items: center;
      min-height: 100vh;
      background-color: #f4f4f4;
    }

    .calendar {
      background-color: #fff;
      border-radius: 8px;
      box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
      padding: 20px;
      text-align: center;
    }

    .calendar-header {
      display: flex;
      justify-content: space-between;
      align-items: center;
      margin-bottom: 15px;
    }

    .calendar-header button {
      background: none;
      border: none;
      cursor: pointer;
      font-size: 1.2em;
      padding: 5px;
    }
```

```
.calendar-title {
  font-size: 1.5em;
  font-weight: bold;
}

.calendar-grid {
  display: grid;
  grid-template-columns: repeat(7, 1fr);
  gap: 5px;
}

.calendar-grid div {
  padding: 10px;
  text-align: center;
  border-radius: 4px;
}

.calendar-grid .weekday {
  font-weight: bold;
  color: #555;
}

.calendar-grid .day {
  cursor: pointer;
}

.calendar-grid .today {
  background-color: #007bff;
  color: white;
}

.calendar-grid .inactive {
  color: #ccc;
}

</style>
</head>

<body>
  <div class="calendar">
    <div class="calendar-header">
      <button id="prevMonth">&lt;</button>
      <h2 id="currentMonth"></h2>
    </div>
  </div>
</body>
```

```

        <button id="nextMonth">&gt;</button>
    </div>
    <div class="calendar-grid" id="calendarDays">
    </div>
</div>

<script>
    const currentMonthElement = document.getElementById('currentMonth');
    const calendarDaysElement = document.getElementById('calendarDays');
    const prevMonthButton = document.getElementById('prevMonth');
    const nextMonthButton = document.getElementById('nextMonth');

    let currentDate = new Date();

    function renderCalendar() {
        const year = currentDate.getFullYear();
        const month = currentDate.getMonth();

        const firstDayOfMonth = new Date(year, month, 1);
        const lastDayOfMonth = new Date(year, month + 1, 0);
        const daysInMonth = lastDayOfMonth.getDate();
        const firstDayOfWeek = firstDayOfMonth.getDay(); // 0 (Minggu) - 6 (Sabtu)

        const monthNames = [
            'Januari', 'Februari', 'Maret', 'April', 'Mei', 'Juni',
            'Juli', 'Agustus', 'September', 'Oktober', 'November', 'Desember'
        ];

        currentMonthElement.textContent = `${monthNames[month]} ${year}`;
        calendarDaysElement.innerHTML = '';

        // Hari-hari dalam seminggu
        const weekdays = ['Min', 'Sen', 'Sel', 'Rab', 'Kam', 'Jum', 'Sab'];
        weekdays.forEach(day => {
            const weekdayElement = document.createElement('div');
            weekdayElement.classList.add('weekday');
            weekdayElement.textContent = day;
            calendarDaysElement.appendChild(weekdayElement);
        });

        for (let i = 0; i < firstDayOfWeek; i++) {
            const emptyDay = document.createElement('div');

```



```

        calendarDaysElement.appendChild(emptyDay);
    }

    for (let day = 1; day <= daysInMonth; day++) {
        const dayElement = document.createElement('div');
        dayElement.classList.add('day');
        dayElement.textContent = day;

        if (year === new Date().getFullYear() && month === new
Date().getMonth() && day === new Date().getDate()) {
            dayElement.classList.add('today');
        }

        calendarDaysElement.appendChild(dayElement);
    }
}

prevMonthButton.addEventListener('click', () => {
    currentDate.setMonth(currentDate.getMonth() - 1);
    renderCalendar();
});

nextMonthButton.addEventListener('click', () => {
    currentDate.setMonth(currentDate.getMonth() + 1);
    renderCalendar();
});

renderCalendar();
</script>
</body>

</html>

```

Latihan 8, War Game Javascript

```

<!DOCTYPE html>
<html>
<head>
<style>
body{ margin:10px; background:#CCC; }
#my_canvas{ background:#FFF; border:#000 1px solid; }
</style>

```

```

<script>
function initCanvas(){
    var ctx = document.getElementById('my_canvas').getContext('2d');
    var cW = ctx.canvas.width, cH = ctx.canvas.height;
    var enemies = [ {"id":"enemy1","x":100,"y":-20,"w":40,"h":20},
                     {"id":"enemy2","x":225,"y":-20,"w":40,"h":20},
                     {"id":"enemy3","x":350,"y":-20,"w":40,"h":20},
                     {"id":"enemy4","x":100,"y":-70,"w":40,"h":20},
                     {"id":"enemy5","x":225,"y":-70,"w":40,"h":20},
                     {"id":"enemy6","x":350,"y":-70,"w":40,"h":20}
    ];
    function renderEnemies(){
        for(var i = 0; i < enemies.length; i++){
            ctx.fillStyle = "blue";
            ctx.fillRect(enemies[i].x, enemies[i].y+=.5, enemies[i].w, enemies[i].h);
        }
    }
    function Launcher(){
        this.y = 280, this.x = cW*.5-25, this.w = 50, this.h = 50, this.dir,
this.bg="orange", this.missiles = [];
        this.render = function(){
            if(this.dir == 'left'){
                this.x-=5;
            } else if(this.dir == 'right'){
                this.x+=5;
            }
            ctx.fillStyle = this.bg;
            ctx.fillRect(this.x, this.y, this.w, this.h);
            for(var i=0; i < this.missiles.length; i++){
                var m = this.missiles[i];
                ctx.fillStyle = m.bg;
                ctx.fillRect(m.x, m.y-=5, m.w, m.h);
                this.hitDetect(this.missiles[i],i);
                if(m.y <= 0){ // If a missile goes past the canvas boundaries, remove
it
                    this.missiles.splice(i,1); // Splice that missile out of the
missiles array
                }
            }
            if(enemies.length == 0){
                clearInterval(animateInterval); // Stop the game animation loop
                ctx.fillStyle = '#FC0';
            }
        }
    }
}

```

```

        ctx.font = 'italic bold 36px Arial, sans-serif';
        ctx.fillText('Level Complete', cW*.5-130, 50, 300);
    }
}

this.hitDetect = function(m,mi){
    for(var i = 0; i < enemies.length; i++){
        var e = enemies[i];
        if(m.x+m.w >= e.x && m.x <= e.x+e.w && m.y >= e.y && m.y <= e.y+e.h){
            this.missiles.splice(this.missiles[mi],1); // Remove the missile
            enemies.splice(i,1); // Remove the enemy that the missile hit
            document.getElementById('status').innerHTML = "You destroyed "+
e.id;
        }
    }
}

}

}

var launcher = new Launcher();
function animate(){
    //ctx.save();
    ctx.clearRect(0, 0, cW, cH);
    launcher.render();
    renderEnemies();
    //ctx.restore();
}
var animateInterval = setInterval(animate, 30);
var left_btn = document.getElementById('left_btn');
var right_btn = document.getElementById('right_btn');
var fire_btn = document.getElementById('fire_btn');
left_btn.addEventListener('mousedown', function(event) {
    launcher.dir = 'left';
});
left_btn.addEventListener('mouseup', function(event) {
    launcher.dir = '';
});
right_btn.addEventListener('mousedown', function(event) {
    launcher.dir = 'right';
});
right_btn.addEventListener('mouseup', function(event) {
    launcher.dir = '';
});
fire_btn.addEventListener('mousedown', function(event) {

```

```

launcher.missiles.push({"x":launcher.x+launcher.w*.5,"y":launcher.y,"w":3,"h":10,"bg":
"red"});
    });
}
window.addEventListener('load', function(event) {
    initCanvas();
});
</script>
</head>
<body>
<canvas id="my_canvas" width="500" height="350"></canvas>
<div>
    <button id="left_btn">Move Left</button>
    <button id="right_btn">Move Right</button>
    <button id="fire_btn">Fire Missile</button>
</div>
<h3 id="status"></h3>
</body>
</html>

```

Latihan 9, Fetch Javascript

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Fetch Javascript</title>
</head>
<body>
    <script type="text/javascript">
        function ambildata() {
            fetch('https://jsonplaceholder.typicode.com/todos/1')
                .then(response => response.json())
                .then(json => console.log(json))
        }
        window.addEventListener("load", async () => {
            await ambildata();
        })
    </script>

```

```
</body>
</html>
```

10. API (Application Programming Interface)

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Fetch Javascript</title>
</head>

<body>
  <h1>API DATA</h1>
  <p id="textarea"></p>
  <script type="text/javascript">
    function ambildata() {
      const data = [
        {
          "id": "1",
          "username": "patra1",
          "email": "patra1@localhost",
          "alamat": "pasar mede fatmawati"
        },
        {
          "id": "2",
          "username": "patra2",
          "email": "patra2@localhost",
          "alamat": "pasar mede fatmawati"
        },
        {
          "id": "3",
          "username": "patra3",
          "email": "patra3@localhost",
          "alamat": "pasar mede fatmawati"
        },
        {
          "id": "4",
          "username": "patra4",
```

```
        "email": "patra4@localhost",
        "alamat": "pasar mede fatmawati"
    }
]

    document.getElementById("textarea").innerHTML='<textarea
class="form-control" cols="100%" rows="20%">' + JSON.stringify(data, undefined, 4) +
'</textarea>';
    }

    window.addEventListener("load", async () => {
        await ambildata();
    })

</script>
</body>

</html>
```