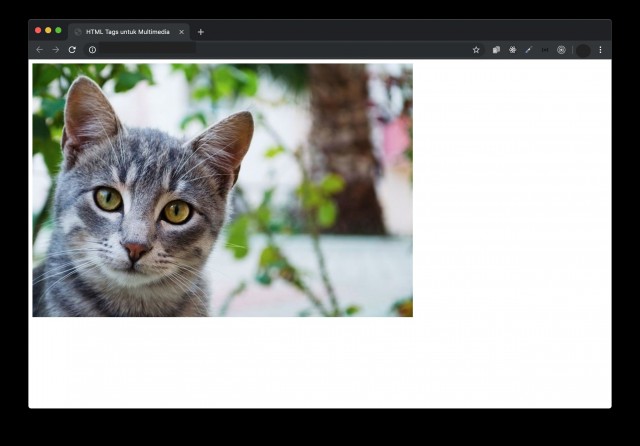
# **Gambar**

Untuk menampilkan gambar pada halaman sebuah website, maka kita membutuhkan tag <img>.

Contoh penggunaannya:

<img src="https://bit.ly/3j6eb3B" alt="Cat" />

Hasil dari kode di atas pada browser akan terlihat seperti ini:



Attribute alt memberikan informasi alternatif atas gambar jika pengguna tidak bisa melihat gambar di website kita. Gambar yang tidak muncul di browser bisa disebabkan karena koneksi internet yang buruk atau terdapat error pada attribute src. Attribute alt juga berguna bagi pengguna screen reader untuk mengetahui informasi tentang gambar tersebut.

Attribute alt harus mendeskripsikan tentang gambar yang bersangkutan jika gambar tersebut berisi informasi. Kemudian, alt juga sebaiknya menjelaskan alamat link jika gambar berada di dalam element <a>. Terakhir, gunakan alt dengan string kosong (contohnya <img alt="" />) apabila gambar hanya berfungsi sebagai dekorasi pada website.

# **Video**

### 

# **Video Element**

Sejak HTML5, sudah ada cara langsung untuk bisa menambahkan video pada halaman web kita; yaitu dengan menggunakan tag <video>. Format video yang didukung antara lain: mp4, ogg, dan webM.

Syntaxnya cukup simpel, seperti di bawah ini:

<video width="320" height="240" src="link-ke-videonya"></video>

atau dengan mengapit element <source> di antara element <video> seperti ini

<video width="320" height="240">

<source src="link-ke-video-nya" type="video/tipe-videonya"/>

</video>

Contoh penggunaannya:

<video width="320" height="240">

<source src="https://bit.ly/3j6rPni" />

</video>

Kita akan melihat hasilnya seperti ini



Lho tapi kok videonya tidak jalan? Dan tidak ada tombol play pula.

# **Autoplay**

Untuk memulai videonya secara otomatis. Kita bisa tambahkan attribute autoplay di dalam element <video>-nya seperti ini

<video width="320" height="240" autoplay>

Dengan begitu videonya akan langsung dimulai sesaat setelah halaman web kita ditampilkan di browser.

# **Controls**

Bagaimana dengan tombol play atau pause-nya?

Nah kalau kita ingin menambahkan tombol-tombol media tersebut, kita bisa tambahkan atribut controls seperti ini

<video width="320" height="240" controls>

Sekarang ada tombol play, pause, dan fullscreen kan?



# **Fallback Video Format**

Oh iya, seperti yang kita bahas di awal, kan ada beberapa format video yang didukung oleh HTML5. Masalahnya tidak semua browser mendukung format video yang sama:

* ada yang hanya mendukung mp4 seperti Internet Explorer dan Safari,
* ada juga yang mendukung semua format seperti Chrome dan Mozilla.

Bagaimana mengatasi masalah ini?

Kita bisa tambahkan beberapa element <source> di dalam <video> seperti ini

<video width="800" height="600">

<source src="link-ke-videonya" type="video/mp4" />

<source src="link-ke-videonya" type="video/ogg" />

<source src="link-ke-videonya" type="video/webm" />

</video>

Dengan begitu, andaikata browsernya tidak mendukung format mp4, dia akan beralih mencoba yang ada di bawahnya: yaitu ogg dan webm.

# **Suara**

Mirip dengan video, HTML5 juga mendukung adanya element yang berisi suara/audio. Untuk menampilkan suara pada halaman sebuah website bisa menggunakan tag <audio>. Format yang didukung pun meliputi mp3, wav, dan ogg.

Syntax penulisannya juga ada dua seperti dengan tag <video>:

<audio controls src="link-ke-file-audio"></audio>

Atau menggunakan tag <source> yang diapit tag <audio>

<audio controls>

<source src="link-ke-file-audio" type="audio/tipe-audionya" />

</audio>

Contoh penggunaannya:

<audio controls>

<source src="https://bit.ly/2EbrKA4" type="audio/mp3" />

</audio>

# **Fallback Audio Format**

Contoh penggunaannya bisa seperti ini:

<audio controls>

<source src="cat\_meow.ogg" type="audio/ogg" />

<source src="cat\_meow.mp3" type="audio/mp3" />

<source src="cat\_meow.wav" type="audio/wav" />

Your browser does not support the audio element.

</audio>

Lho itu kenapa ada banyak element <source>-nya?

Dengan adanya beberapa <source> dengan type yang berbeda-beda, apabila sebuah browser tidak mendukung file audio yang kita berikan, dia akan secara otomatis membaca file yang berikutnya. Jadi jika kita ingin semua fitur website kita bisa diakses semua orang, sebaiknya kita memberi beberapa jenis file audio; jaga-jaga apabila file audionya tidak didukung.

# **Controls**

Kalau kalian coba kode yang tadi, kalian bisa lihat ada tombol medianya seperti play, pause, dan timeline audionya. Itu karena kita memberikan attribute controls ke dalam element <audio>-nya.

# **Loop**

Jika kita memberikan attribute loop di dalam element <audio> kita, suara yang dimainkan akan terus mengulang tiap kali selesai. Mungkin ini cocok untuk background music yang memang terus dimainkan tanpa henti selama pengguna terus berada di halaman web tersebut.

# **Autoplay**

Dengan attribute autoplay, suara akan langsung dimainkan sesaat setelah halaman web selesai di-load. Seperti yang barusan kita bahas, mungkin attribute ini cocok untuk background music pada sebuah website yang

# **IFrame**

Kita juga bisa menampilkan halaman website lainnya di halaman website yang kita buat dengan menggunakan tag <iframe>.

Contoh penggunaannya adalah seperti ini:

<iframe></iframe>

# **Width dan Height**

Jadi halaman html yang lain itu akan diperlakukan sebagai sebuah element dalam struktur file HTML kita: bisa diatur tinggi-lebarnya juga dengan menambahkan attribute width dan height seperti ini:

<iframe width="lebar iframe" height="tinggi iframe" ></iframe>

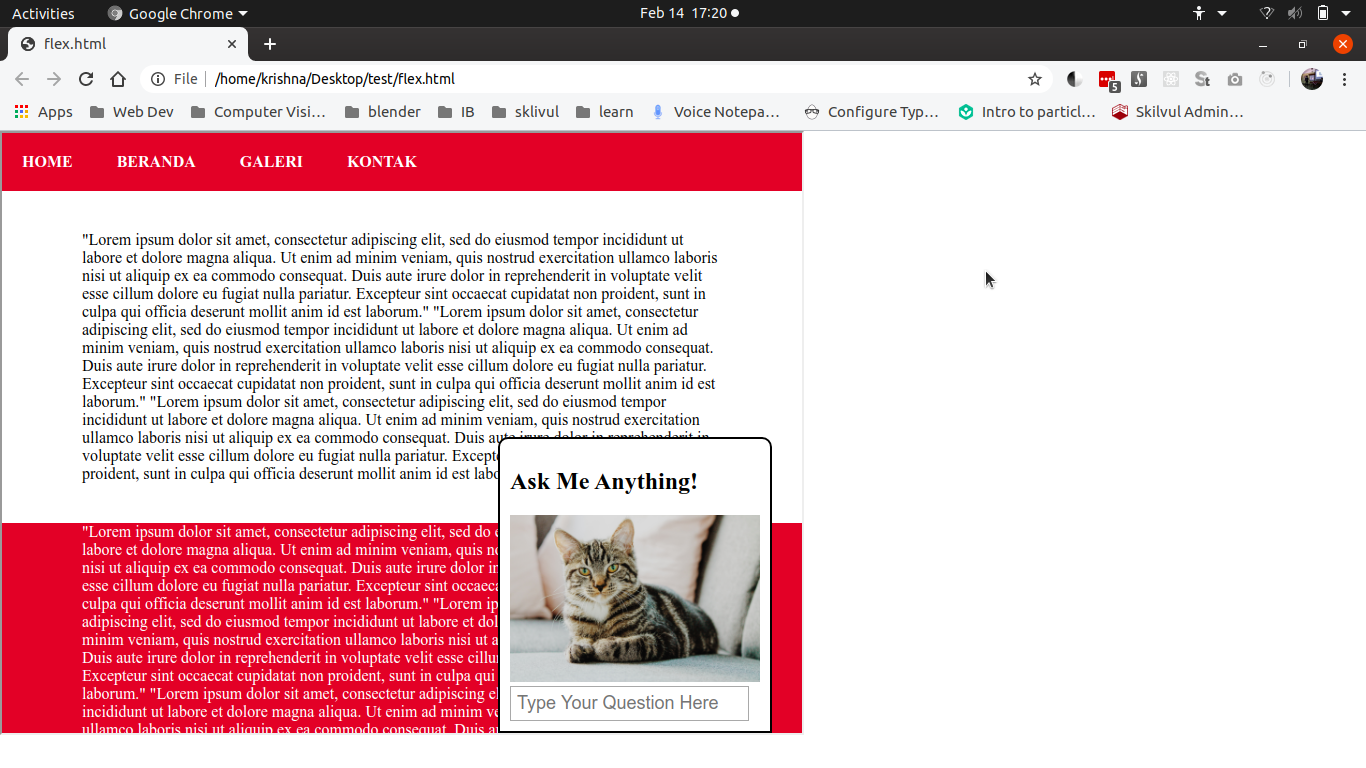
# **Sumber Konten HTML**

<iframe> itu kan isinya konten HTML, kita bisa menambahkan attribute src untuk memberitahu konten apa yang harus ditampilkan.

Cara menambahkan attribute src ke dalam <iframe> adalah seperti berikut:

<iframe src="link-ke-konten-html"></iframe>

Sekarang halaman web kita akan menampilkan halaman web lain di dalamnya.



# **Scrolling**

Andaikata konten HTML yang kita masukkan ke dalam <iframe> kita itu lebih besar dari ukuran <iframe>-nya sendiri, secara default akan ada scrollbar untuk scroll halamannya.

Nah kita memiliki opsi untuk menghilangkan scrollbar tersebut dengan menambahkan attribute scrolling="no" di dalam <iframe> seperti ini:

<iframe scrolling="no"></iframe>

# **Frameborder**

Kalau kalian lihat, di bagian pinggir <iframe> akan ada pinggiran berwarna abu-abu bukan? Kalau kalian merasa itu jelek atau tidak rapi, kalian bisa menambahkan attribute frameborder="0" di dalam <iframe>-nya seperti ini

<iframe frameborder="0"></iframe>

Untuk merangkum cara penggunaan semua attribute di <iframe>, kita bisa melihat contoh berikut:

<iframe src="https://bit.ly/32jR7ro" width="200" height="300" scrolling="no" frameborder="0"></iframe>

Kode di atas akan menghasilkan <iframe> seperti berikut:

* sumber konten yang mengarah ke halaman https://bit.ly/32jR7ro
* lebar sebesar 200 dan tinggi sebesar 300
* tidak mempunyai scrollbar apabila konten HTML di dalamnya lebih besar dari ukuran <iframe>
* tidak mempunyai border yang membungkus element <iframe>