

Introdução Formulários

Desenvolvimento Web

→ Objetivo da Aula

- Revisão de conteúdo CSS
- Conhecer protocolo http
- Compreender o que é API.
- Compreender Formulários
- Exercício em Aula

O que é HTTP?

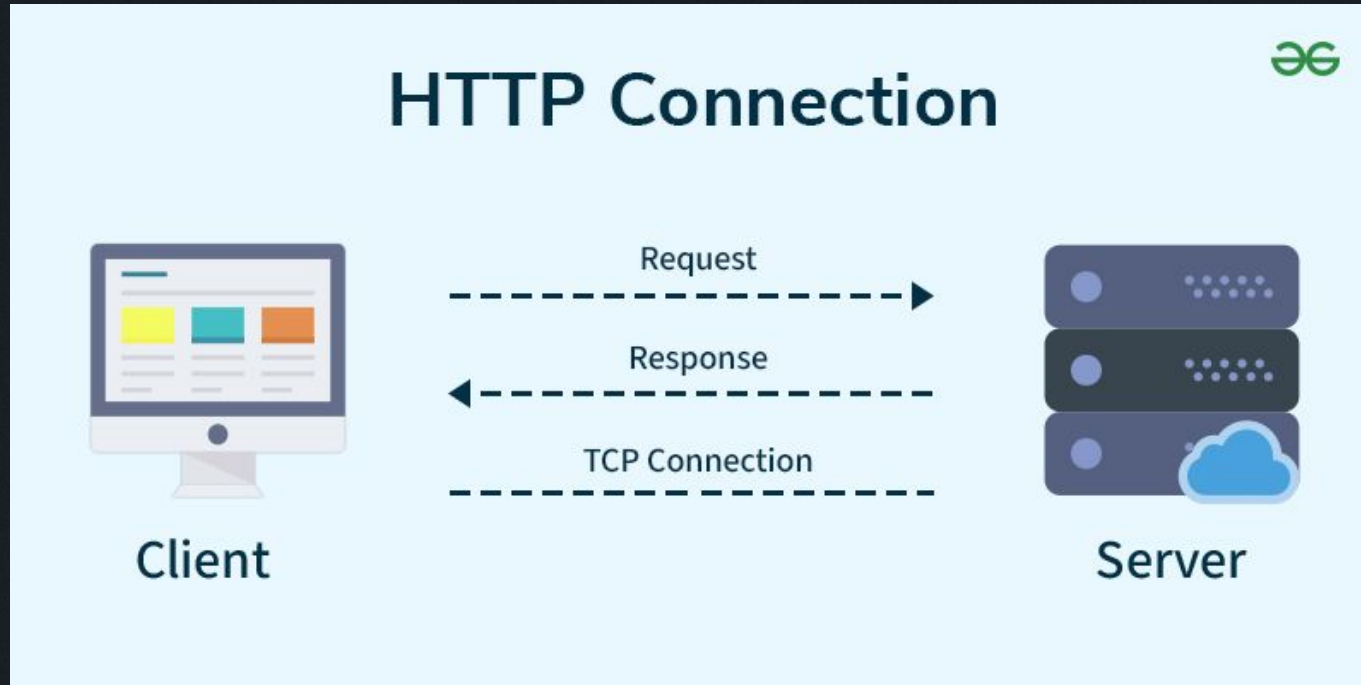
- **H**ypertext **T**ransfer **P**rotocol (Protocolo de Transferência de Hipertexto)
- Utilizada para comunicação entre cliente e servidor
- **U**RL usada para localizar o servidor
- **M**ethod, definir qual o tipo de requisição

Verbo HTTP	Operação	Idempotente
GET	Leitura de recursos	Sim
POST	Criação de recursos	Não
PUT	Atualização de recursos	Sim
DELETE	Remoção de recursos	Sim

Métodos HTTP

/books		
GET	/books	Lists all the books in the database
DELETE	/books/{bookId}	Deletes a book based on their id
POST	/books	Creates a Book
PUT	/books/{bookId}	Method to update a book
GET	/books/{bookId}	Retrieves a book based on their id

Funcionamento



O que é uma API externa?

- Uma API externa, também conhecida como **Web Service** é “um sistema de software projetado para suportar comunicação interoperável, máquina-a-máquina através de uma rede.”

API vs Web API vs Web Service

- **API:** É um acrônimo para Application Programming Interface que significa Interface de Programação de Aplicação.
- **Web API:** Nesse contexto, é uma API de navegador pode estender a funcionalidade de um navegador da web(browser).
- **Web Service / Web API (Server):** É uma API de servidor que pode disponibilizar as funcionalidades de um back-end.

Como as APIs externas funcionam?

A arquitetura de API geralmente é explicada em termos de **cliente** e **servidor**. A aplicação que envia a requisição é chamada de cliente e a aplicação que envia a resposta é chamada de servidor.

LocalStorage armazenamento

```
// Armazenando um valor simples
localStorage.setItem('username', 'JohnDoe');

// Armazenando um objeto como JSON
const user = {
  name: 'John Doe',
  email: 'john.doe@example.com'
};
localStorage.setItem('user', JSON.stringify(user));
```

LocalStorage captura de dados

```
// Obtendo um valor simples
const username = localStorage.getItem('username');
console.log(username); // "JohnDoe"

// Obtendo um objeto armazenado como JSON
const storedUser = localStorage.getItem('user');
if (storedUser) {
  const user = JSON.parse(storedUser);
  console.log(user.name); // "John Doe"
  console.log(user.email); // "john.doe@example.com"
}
```

LocalStorage exclusão de dados

```
// Removendo um item específico
localStorage.removeItem('username');

// Limpando todos os dados do localStorage
localStorage.clear();
```


Consultando APIs externas

Introdução à API's externas

Cliente



Servidor

Tag Form HTML

- Form é utilizado para capturar os dados do usuário e realizar o envio para o servidor.

Estrutura de um Form

```
<form action="/submit" method="POST">  
  <label for="name">Nome:</label>  
  <input type="text" id="name" name="name">  
  
  <label for="email">Email:</label>  
  <input type="email" id="email" name="email">  
  
  <button type="submit">Enviar</button>  
</form>
```

Estrutura de um Form

```
<form action="/submit" method="POST">  
  <label for="name">Nome:</label>  
  <input type="text" id="name" name="name">  
  
  <label for="email">Email:</label>  
  <input type="email" id="email" name="email">  
  
  <button type="submit">Enviar</button>  
</form>
```


Introdução linguagem HTML

Introdução HTML



Exercício

- Escrever uma **página HTML**. A página deve conter:
 - Um Formulário para obter: Nome, telefone, CPF, uma lista com time favorito.
 - Gravar esses dados no navegador. Local storage
 - Listar dados Salvos