

Prova Desenvolvimento WEB – 25/06/2020

Professor: Ramon

Aluno: Thomas Moreira

RA: 119111467

Questão 1.

Get, Post, Put Delete

Questão 2.

O Fetch API fornece uma interface JavaScript para acessar e manipular partes do HTTP pipeline, tais como os pedidos e respostas (MDN).

O coração do Fetch são as abstrações da Interface do HTTP Request, Response, Headers, e Body payloads, juntamente com global fetch método para iniciar requisições de recursos assíncronos. Como os componentes principais do HTTP são abstraídos como objetos de JavaScript, torna-se fácil APIs fazer uso das funcionalidades.

Questão 3.

O método getElementById() retorna o elemento que possui o atributo com o valor especificado. Ele tem que ser único no documento e só é recuperado uma vez.

O método getElementsByName () retorna uma coleção de todos os elementos no documento com o nome

<p name = "nome">

O método getElementsByClassName () retorna uma coleção de todos os elementos no documento com o nome da classe especificado.

Questão 5.

- a) As rotas de uma API é uma definição de destino para uma solicitação. Ou seja, elas servem para identificar como serão os resultados que a API deverá retornar. Como exemplo quando passamos um diretório (nome) e a aplicação, nessa rota consegue saber quem chamou, o que chamou e como chamou.
- b) Javascript, PHP, C#, PYTHON.
- c) Devemos considerar os Métodos HTTP pois com ele podemos buscar, criar, alterar e deletar informações na API.
Ao enviar dados, é importante conhecer qual será a rota que ele vai percorrer para evitar enviar informações sensíveis como senhas e dados pessoais.
- d) Os principais métodos para a construção de uma API são:

GET => Este método serve para buscar dados

DELETE => Este método é utilizado para deletar dados

PUT => Estes métodos são utilizados para atualizarem informações.

POST => Este método serve para enviar dados para o sistema.
Quando for necessário enviar documentos utilizamos este método.

Questão 6.

O PHP nos oferece algumas constantes predefinidas que podem ser utilizadas em qualquer lugar da aplicação. Essas constantes são chamadas de constantes mágicas.

Elas são chamadas de constantes mágicas porque o seu valor muda dependendo de onde elas são utilizadas, isso se deve ao fato, de que, elas obtêm o valor de onde elas estão no script, por exemplo, a constante mágica `__LINE__` vai retornar a linha do código onde essa constante foi escrita.