



Questão 01

O que é o JSON?

- ☒ a) Um formato de dados em texto legível e manipulável por humanos e computadores.
- b) Um framework para desenvolvimento web.
- c) Um formato de dados disponível apenas no javascript.
- d) Uma biblioteca do angular.
- e) Uma linguagem de programação.

Questão 02

Qual o objetivo da interface Pessoa no typescript?

- a) Implementar métodos em services.
- b) Definir o contrato usado no componente.
- c) Intercambiar implementações de services.
- d) Implementar métodos em Pessoa.
- ☒ e) Modelar o tipo de dados retornado e recebido pelas funções de forma a ter uma checagem pelo compilador do uso no código.

Questão 03

O que o generic do método http do angular faz para nós? Por exemplo: `http.get<Pessoa>()`.

- a) Serializa o dado para interface Pessoa.
- b) Retorna um cabeçalho do tipo Pessoa.
- ☒ c) Diz o tipo de dados que será retornado no corpo da mensagem quando transformado de json em objeto.
- d) Diz o tipo de dados que é passado como parâmetro ao fazer um post.
- e) Diz o tipo de dado que é aceito nos cabeçalhos da requisição.

Questão 04

Onde é declarado os serviços no Angular?

- a) No parâmetro imports do módulo.
- ☒ b) No parâmetro providers do módulo.
- c) No parâmetro exports do módulo.
- d) No parâmetro declarations do módulo.

- e) Nas rotas do modulo AppRoutingModuleModule.

Questão 05

No exemplo da API REST os métodos post, put e delete do HttpClient poderiam ter retornado <string> ao invés de uma resposta HttpResponseMessage. Qual das afirmativas a seguir é falsa sobre as vantagens de obter o HttpResponseMessage?

- a) Obter a resposta Http completa pode tornar a aplicação mais lenta e pesada.
- b) Obter a resposta Http completa pode facilitar o tratamento de erros e exceções.
- c) A resposta Http completa pode conter informações adicionais úteis para a aplicação.
- d) Apenas o corpo da resposta pode não fornecer informações suficientes para o tratamento de erros e exceções.
- e) Ter acesso ao status da resposta pode ser usado para pegar o status retornado pelo backend para tratar erros de formulário ou erros de sistema.

Questão 06

Leia o texto a seguir:

“A utilização de requisições HTTP é uma prática comum em aplicações web, principalmente na comunicação entre o cliente e o servidor. O framework Angular possui uma biblioteca própria para realização de requisições HTTP.”

Agora, considere que uma aplicação Angular está consumindo uma API RESTful para obter informações de uma base de dados. Para realizar uma requisição GET para a URL "https://api.example.com/users/1", é possível utilizar o serviço HttpClient da seguinte forma:

```
import { HttpClient } from '@angular/common/http';
import { Injectable } from '@angular/core';
import { Observable } from 'rxjs';

@Injectable()
export class UserService {
  private apiUrl = 'https://api.example.com';

  constructor(private http: HttpClient) {}

  getUserById(id: number): Observable<any> {
    const url = `${this.apiUrl}/users/${id}`;
    return this.http.get<any>(url);
  }
}
```

Com base nesse código, qual é a função da linha "return this.http.get<any>(url);"?

- a) Definir a URL da API para a consulta dos dados do usuário com base no id informado.
- b) Enviar a requisição HTTP do tipo GET para a URL informada e retornar o resultado em formato JSON em um Observable.
- c) Enviar a requisição HTTP do tipo POST para a URL informada e retornar o resultado em formato JSON.
- d) Verificar se o usuário com o id informado está disponível na base de dados.
- e) Converter o resultado da requisição HTTP para um objeto do tipo Observable.