1. B
2. B
3. D
4. B
5. C
6. C
7. Sim, pois o Spring Boot é amplamente utilizado em aplicações empresariais devido à sua alta robustez, modularidade e suporte para integrações com diferentes tecnologias. Além disso, por ser Open Source, possui uma comunidade muito ativa.
8. Primeiramente, no “RED”, escreva o teste unitário para um método da API antes de implementá-lo. A ideia é que este primeiro teste falhe porque a funcionalidade não foi implementada.

Segundamente, no “GREEN”, cria-se o controlador para atender o teste com um método mínimo para que o teste funcione.

Por último, no “REFACTOR”, melhore o código e rode o código novamente para garantir que ainda esteja tudo rodando perfeitamente.

@RestController → Define que a classe é um controlador REST.

@RequestMapping("/tarefas") → Define o mapeamento base da API.

@GetMapping → Mapeia requisições **GET** para um método específico.

@Service → Marca a classe como um **serviço** gerenciado pelo Spring.

@SpringBootTest → Indica que estamos testando um contexto do Spring Boot.

@AutoConfigureMockMvc → Habilita o uso do **MockMvc** para testes de API.

1. ------------------------------------- Código
2. ----------- NÃO FAZER -----------
3. C
4. A principal vantagem de usar Spring JPA é a simplificação do acesso aos dados e a implementação automática dos métodos. Além disso, ele também apresenta interfaces prontas, suporte a paginação, ordenação, integração com Hibernate e outros ORMs.
5. A)
   1. @RestController indica que a classe é um controlador Rest
   2. @ResquestMapping define o caminho básico do controlador
   3. @GetMapping mapeia o método para atender requisições HTTP Get na URL
   4. @RequestParam captura parâmetros na URL e converte em Double

B) <HTTP://localhost:8080/calculadora/somar?n1=5&n2=10>

C) Utilizaria testes unitários, testes de integração e testes de API.

1. Sim, há vantagens e desvantagens em utilizar a Injeção de Dependência via Construtor em comparação com a injeção via @Autowired no Spring Boot. As vantagens são que ele facilita teste unitários, promove imutabilidade e é mais seguro contra NullPointerException. Já das desvantagens, ele pode ser mais verboso e não funciona bem com injeção opcional.
2. ------------------------------------- Código
3. D
4. E
5. B
6. C
7. Vantagens do Spring JPA – Menos códigos, métodos de consulta automática, paginação, integração com o próprio Spring Boot.

Vantagens do JPA/Hibernate – Maior controle sobre as consultas, mais eficiência em aplicações de alto desempenho, não depende do Spring.

1. Três vantagens: DB em memória RAM mais rápido e leve, fácil configuração e integração com Spring Boot e útil para testes unitários e integrações.

Data Source é o objeto responsável por gerenciar conexões com o banco de dados. No Spring ele é configurado automaticamente com base no DB escolhido.

1. A anotação @Configuration junto com @Bean pode ser usada para configurar explicitamente a conexão com o H2 no Spring Boot.

- Informações importantes na anotação:

@Configuration: Define que esta classe contém configurações do Spring.

@Bean: Cria um bean gerenciado pelo Spring, definindo um DataSource manualmente.

driverClassName("org.h2.Driver"): Define o driver do banco.

url("jdbc:h2:mem:meubanco"): Define a URL de conexão (neste caso, um banco em memória).

username("sa") e password(""): Usuário e senha padrão do H2.

1. ------------------------------------- Código