

AVALIAÇÃO DESENVOLVEDOR

NOME:

CONTATO (E-mail e Telefone):

GITHUB (se tiver): github.com/_____

1. Desenhe um schema de banco de dados com tabelas que atendam aos requisitos descritos abaixo; poderá ser utilizado diagramas, texto ou qualquer outra forma de representação de schema de banco de dados, contanto que sejam apontados todos os relacionamentos e os tipos de dados:

Deseja-se guardar os seguintes dados:

- alunos (id, matricula, nome, sexo, idade e endereco)
- professores (id, nome, cpf)
- disciplinas (id, nome)
- turma (id, nome, turno).

Modele de uma forma que obedeça as seguintes regras:

1. Um aluno poderá se matricular em N turmas.
2. Um professor poderá dar aula em N turmas e N disciplinas.
3. Uma turma só poderá ter 1 professor por disciplina (Não é permitido ter 2 professores dando aula na mesma turma com a mesma disciplina)
4. Uma turma poderá ter aula de várias disciplinas. (Ex: turma 1º A poderá ter aulas de Português, Matemática, etc.)

2. Com base no schema criado na questão anterior, crie as seguintes consultas em linguagem SQL:

- a. Listar quantidade de alunos com mesmo nome.
- b. Listar as turmas cujo aluno “José” está matriculado com o professor “Armando”.
- c. Listar alunos(as) do sexo feminino que não estão em turmas da disciplina "Artes".
- d. Listar os alunos da disciplina de “Matemática” com nome em ordem alfabética.

3. Selecione as respostas corretas:

a. Qual o resultado quando o seguinte fragmento de código for executado?

```
int a = 8;
int b = 3;
float f = a++/b--;
System.out.println( f );
```

- A) 3.0
- B) 4.5
- C) 2.0
- D) 2.6

b.O que acontece ao tentar compilar e executar o seguinte código?

```
public class TestEqual {
    public static void main(String args[]) {
        Integer i = new Integer(7);
        Long l = new Long(7);
```

```

        System.out.println(l.equals(i));
    }
}

```

- A. imprimirá - true
- B. imprimirá - false
- C. erro de compilação

c. Dado o trecho de código abaixo, escreva um código em Java que imprima apenas os nomes que começa com "ana" (case insensitive, ou seja, maiúsculas e minúsculas não devem fazer diferença). (User característica Java 8 ou superior será considerado bônus)

```

List<String> nomes = Arrays.asList("Ana Carla", "João Carlos", "Mário Augusto",
"Reberto Ana Nascimento", "ana Maria", "Sophia ana");

```

d. Dado o trecho de código abaixo, utilize Lambda Expressions introduzido no Java 8 para reduzir a quantidade de código digitado. (Utilizar outras características a mais do Java 8 ou superior será considerado com bônus)

```

new Thread(new Runnable() {
    @Override
    public void run() {
        System.out.println("Hello world!");
    }
}).start();

```

4. Cite pelo menos 2 padrões de projeto (Design Patterns) que você já utilizou e discorra brevemente como ele resolveu o problema em questão. Caso não tenha utilizado, pode discorrer sobre os que você já ouviu falar.

5. Comente qualquer ideia a respeito de metodologias ágeis.

6. Das bibliotecas/frameworks/padrões abaixo, assinale as que você conhece, já ouviu falar ou utilizou, e descreva para que serve com suas palavras:

[] Lombok: _____

[] Spring Security: _____

[] Mockito: _____

[] Angular/React: _____

[] JPA: _____

[] PrimeFaces: _____

[] Injeção de Dependências: _____

A resolução das próximas questões deverão ser implementadas no eclipse. Utilizar o projeto pré-configurado.

7. Crie uma classe chamada Main com um método que receba 2 listas de Strings e retorne uma só lista contendo a junção das duas listas recebidas mas sem conter elementos repetidos, ou seja, se a mesma string estiver nas duas listas, a lista retornada não pode repetir esse elemento. A lista retornada deverá estar com os elementos ordenados em ordem alfabética.

8. Implemente o que se pede abaixo:

a) Criar a classe com mapeamento JPA para a entidade Aluno.

b) Criar classe que vai acessar a base e executar as operações com a entidade Aluno, pode ser utilizado Spring Data Repositories, EntityManagers, JDBC, etc.

c) Implementar um serviço REST utilizando SpringBoot para a entidade Aluno (matricula, nome, sexo, idade e cidadeNatal), o serviço deverá conter as operações CRUD (salvar, atualizar, deletar, listar todos e buscar aluno por matrícula). O Serviço deverá utilizar a classe criada no item “b” para fazer as operações na base.

Extra: Será considerado um plus qualquer recurso extra que implementar, como por exemplo, implementar testes ou outros endpoints no serviço de alunos. Mostre suas habilidades!