

Exercício 6 da Lista JUNIT

Nesse exercícios, foram criados 8 testes para verificar as saídas do método e as condicionais. Foram aplicadas técnicas de JUnit 5, utilizando assertivas como `assertThrows` para verificar exceções e `assertEquals` para comparar os resultados com os valores esperados. Os testes cobriram cenários como lista vazia, nula e escores fora do intervalo (negativo ou maior que 100), além de verificar a atribuição correta dos conceitos conforme o escore.

Exercício 5 da Lista de Mocking

Este conjunto de testes utiliza o Mockito para criar mocks de dependências externas da classe `TurmaController`, como `TurmaDAO` (responsável pela persistência de dados) e `VerificadorDeCodigos` (responsável pela validação de códigos de disciplina e turma). A classe de teste é organizada com o uso da anotação `@BeforeEach` para inicializar os objetos necessários antes de cada execução de teste, garantindo um ambiente limpo para cada caso.

Os testes são realizados em diferentes cenários, com a utilização de métodos como `when(...).thenReturn(...)` para configurar o comportamento dos mocks e simular as respostas esperadas. A lógica de testes envolve verificar a validade do código da disciplina e da turma, além de garantir que o comportamento do sistema seja adequado, seja em caso de erro (como código inválido ou turma já existente), seja em sucesso (quando a turma é salva corretamente ou falha ao tentar salvar no banco de dados).

Além disso, a ferramenta JUnit 5 é usada para estruturar os testes, com assertivas como `assertEquals` para comparar o retorno do método `cadaststrarTurma` com os valores esperados.

Exercício 7 cx-branca-GFC

Os testes na classe `EmissarioTest` verificam o comportamento do envio de mensagens para destinatários. Primeiramente, são testados cenários com listas de destinatários vazias ou nulas, retornando a mensagem "nomes nao informados". Em seguida, são validados casos em que não há usuários ou quando os registros são vazios, retornando "nao ha usuarios". Os testes também verificam o sucesso no envio de mensagens, com criptografia e interação com o servidor de e-mail, retornando "mensagens enviadas", e falhas no servidor, que resultam em "servidor de email offline". Por fim, é validado o caso onde usuários da lista não são encontrados, retornando "usuarios nao encontrados". Todos os testes são realizados com mocks para isolar a classe `Emissario`.