

Discente: Samuel Oliveira Silva Bianch

Implementação: <https://github.com/samuelbianch/LP2/tree/main/AlunoVO/crud-aluno-vo/src/main/java/ifmt/cba>

Respostas Questionário

1. A grande vantagem da enumeração é padronizar o que será informado pelo usuário, visto que em uma grande diversidade de usos, cada um pode digitar uma mesma informação de formas diferentes.
2. As classes VO's tem por papel nesta implementação o encapsulamento dos dados, sendo então responsável pelos métodos de acesso, tanto para consulta, quanto para alteração do objeto a ser manipulado, ou seja, ela é um comunicador entre a camada de visualização e controle.
3. Haveria uma comunicação direta entre as classes DAO e a camada *View*.
4. A classe EnderecoVO é agregada em AlunoVO, então na classe AlunoVO guarda-se um objeto do tipo EnderecoVO.
5. O endereço pode ser interpretado como um tipo composto na base de dados (sendo então criada outra entidade), entretanto nesta implementação é apenas uma opção para que suas associações se mantenham separadas, como uma regra de negócio desta aplicação.
6. Quando no campo “numero” enviamos ao banco um valor do tipo String do java, recebe-se uma *PersistenciaException*.
7. A grande vantagem é a facilidade na manutenção do código da conexão com o banco de dados, visto que diante de qualquer alteração no acesso ao banco, altera-se apenas a classe responsável por este acesso.
8. Diferente do VO, a classe DAO é responsável pelo encapsulamento do objeto acessado no banco de dados, ela possui os métodos do *CRUD* da aplicação e persiste no banco.
9. A classe AlunoNegocio tem o papel da validação de dados do objeto de AlunoVO, então ela garante que nenhum dado será persistido de forma equivocada no banco de dados.
10. Talvez para trabalhar no modelo MVC, falta aplicar o conceito do Model, modelando então a aplicação e até mesmo conversando com o *schema* sobre adições/exclusões de tabelas e colunas na base.