

1. (1,0 ponto) Explique o conflito existente entre os conceitos do modelo de banco de dados relacional e do modelo de programação orientada a objetos. Dê alguns exemplos destes conflitos.

R: O conflito existe por haver uma diferença entre a representação dos dados no modelo relacional em comparação ao modelo orientado a objetos. O modelo relacional, como sabemos, se baseia na ideia do armazenamento de dados em linhas, colunas, que possuem relacionamentos com outras tabelas.

Por outro lado, o modelo orientado a objetos possui, além dos atributos, métodos para manipular informações, tornando-o mais complexo o processo de persistência de dados.

Conflitos :

Representação de dados: O modelo de banco de dados relacional requer que os dados sejam armazenados em tabelas com um esquema fixo. Na programação orientada a objetos, os dados são representados em objetos que podem ser modificados dinamicamente. Essa diferença pode levar a problemas quando se tenta mapear uma tabela de banco de dados para um objeto em um programa orientado a objetos.

Relacionamentos: No modelo relacional, utilizamos chaves estrangeiras para relacionar tabelas. No modelo orientado a objetos, utilizamos a referência dos objetos como forma de relacionamento. Ou seja, isso pode ocasionar algum tipo de problema na construção de relacionamentos mais complexos.

2. (1,0 ponto) Cite duas funcionalidades de uma aplicação em que seria possível usar a persistência poliglota. Por exemplo: para a funcionalidade A seria mais interessante usar um banco de dados relacional, já para a funcionalidade B o ideal seria usar um banco de dados orientado a documentos. Justifique por que as funcionalidades mencionadas devem ser associadas aos modelos de banco de dados definidos.

R: Persistência Poliglota : é uma técnica que permite a utilização de múltiplos modelos de banco de dados em uma mesma aplicação. Isso é interessante porque aproveita as vantagens de cada modelo para cada funcionalidade específica.

Exemplo: Em um sistema de gerenciamento de produtos, pode ser interessante o uso de um banco de dados relacional para armazenar uma série de informações mais estruturadas e bem definidas, como nome do produto, tipo, valor e etc.

Em uma segunda funcionalidade, podemos utilizar um MongoDB, que é do tipo documento, para armazenar um grande volume de dados referente a vendas.

3. (1,0 ponto) Explique com suas palavras o que é mapeamento objeto relacional e por que ele é necessário em aplicações orientadas a objetos baseadas em bancos de dados relacionais. Existe algo similar para bancos de dados orientados a documentos?

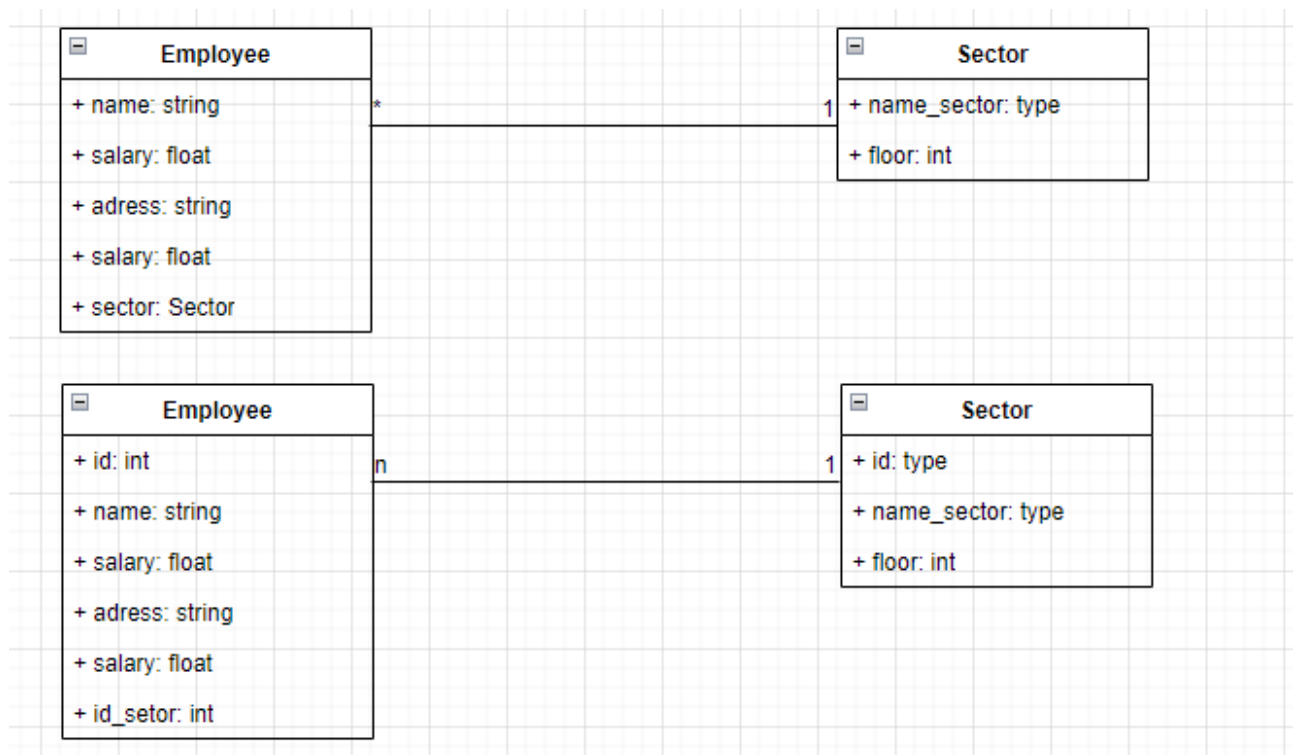
R: Mapeamento Objeto Relacional, ou ORM, é uma técnica utilizada em aplicações POO visando mapear objetos e, posteriormente, convertê-los para uma tabela definida em um banco de dados relacional. Em bancos de dados orientados a documentos, o equivalente seria o ODM (Mapeamento de Documentos Orientado a Objetos). Afinal, ele trabalha com representações do objeto no formato JSON.

4. (1,0 ponto) Considerando o conceito de unicidade dos objetos, por que é importante não usar atributos ligados ao domínio de negócio nas chaves primárias das tabelas? Por que o uso de chaves compostas em tabelas também não é recomendado?

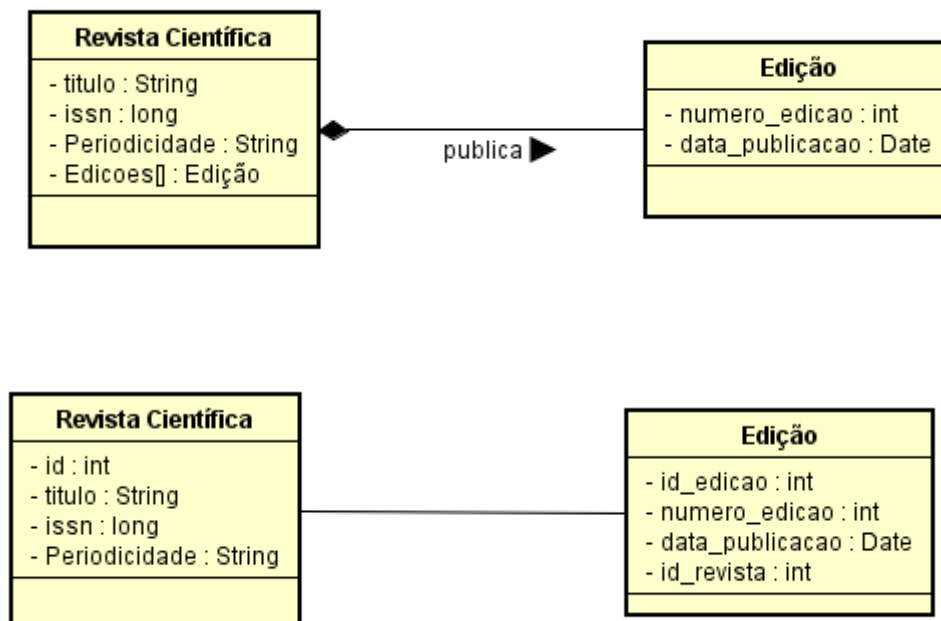
R: Usar atributos do negócio como chaves primárias pode dar ruim. Pode gerar problemas com a integridade dos dados e referências em outras tabelas. Além disso, alguns atributos podem sofrer alterações ao longo do tempo, a exemplo do CPF, RG e etc. Chaves compostas também não são a melhor opção, porque complicam o gerenciamento do banco de dados. A melhor opção é usar uma chave simples e única que não tenha nada a ver com o negócio e que não seja composta por mais de um atributo. Assim, a gente garante que os objetos no banco de dados são únicos.

5. (5,0 pontos) Desenhe um diagrama de classes e um diagrama entidade-relacionamento para representar cada técnica listada a seguir, as quais são usadas para mapear relacionamentos de modelos de objetos para bancos de dados. Obs.: Use classes diferentes das apresentadas nos exemplos da aula. O diagrama de classes não precisa conter métodos, apenas atributos.

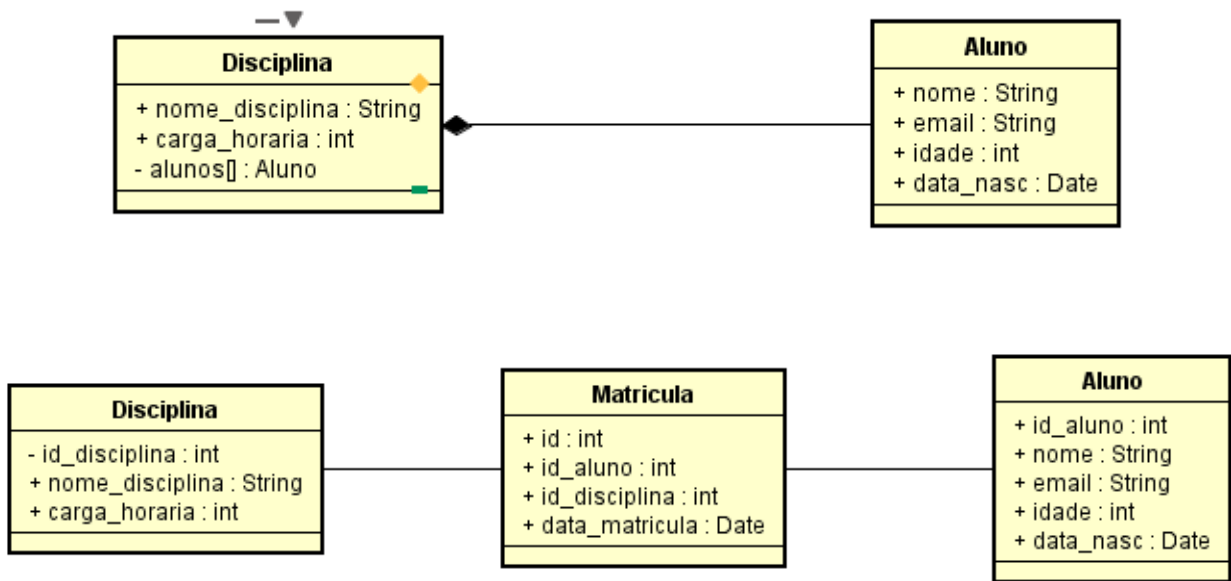
a) Foreign Key Mapping – Associação (duas classes são mapeadas para duas tabelas)



b) Foreign Key Mapping – Composição (duas classes são mapeadas para duas tabelas)



c) Association Table Mapping – Agregação (duas classes são mapeadas para três tabelas)



d) Association Table Mapping – Agregação (três classes são mapeadas para três tabelas)

e) Single Table Inheritance (três classes são mapeadas para uma tabela)

f) Concrete Table Inheritance (três classes são mapeadas para duas tabelas)

g) Class Table Inheritance (três classes são mapeadas para três tabelas)

6. (1,0 ponto) Explique com suas palavras como funciona a técnica Lazy Initialization, ressaltando qual é a vantagem de usá-la.

R: Técnica Lazy Initialization é uma estratégia útil para adiar a inicialização de objetos até o momento em que são realmente necessários, economizando recursos computacionais e melhorando o desempenho da aplicação. Essa técnica utiliza métodos interceptadores, que executa automaticamente uma função quando uma propriedade não existente no objeto é requisitada.