



DESENVOLVIMENTO DE SOFTWARE PARA PERSISTÊNCIA

**Trabalho Prático 2 - Mapeamento Objeto-Relacional com SQLAlchemy e
SQLModel**

Discentes:

Docente:

Francisco Victor da Silva Pinheiro

QUIXADÁ - CE

JANEIRO DE 2025

Repositório dos códigos: <https://github.com/gabriel-bri/InnovateAcademyTech/>

1. Definir uma entidade e criar uma classe python

Entidade **Curso** que contém 6 atributos listados abaixo:

- id_curso: Optional[int] (Primary Key)
- nome_curso: str
- descricao: str
- categoria_id: Optional[int] (Foreign Key para Categoria)
- horas_totais: str
- modulos: List["Modulo"] (Relacionamento com Modulo)
- instrutor_id: int (Foreign Key para Instrutor)

Entidade **Aluno** que contém 5 atributos listados abaixo:

- id_aluno: Optional[int] (Primary Key)
- nome_completo: str
- descricao: Optional[str]
- contato_email: str
- saldo: float (Valor inicial: 0.0)

Entidade **Aula** que contém 5 atributos listados abaixo:

- id_aula: Optional[int] (Primary Key)
- titulo: str
- descricao: Optional[str]
- duracao: time
- material: Optional[str]

Entidade **Avaliacao** que contém 6 atributos listados abaixo:

- id_avaliacao: Optional[int] (Primary Key)
- curso_id: Optional[int] (Foreign Key para Curso)
- aluno_id: Optional[int] (Foreign Key para Aluno)
- nota: float
- comentario: Optional[str]
- data_avaliacao: datetime (Data de criação)

Entidade **Categoria** que contém 4 atributos listados abaixo:

- id_categoria: Optional[int] (Primary Key)
- nome_categoria: str
- descricao: Optional[str]
- data_criacao: datetime (Data de criação)

Entidade **Certificado** que contém 5 atributos listados abaixo:

- id_certificado: Optional[int] (Primary Key)
- aluno_id: Optional[int] (Foreign Key para Aluno)

- curso_id: Optional[int] (Foreign Key para Curso)
- data_emissao: datetime (Data de criação)
- codigo_verificacao: str

Entidade **Inscricao** que contém 6 atributos listados abaixo:

- id_inscricao: Optional[int] (Primary Key)
- aluno_id: Optional[int] (Foreign Key para Aluno)
- curso_id: Optional[int] (Foreign Key para Curso)
- data_inscricao: datetime (Data de inscrição)
- status: str (Exemplo: "Ativo", "Concluído", "Cancelado")
- progresso: float (Percentual de conclusão)

Entidade **Instrutor** que contém 5 atributos listados abaixo:

- id_instrutor: Optional[int] (Primary Key)
- nome_completo: str
- descricao: Optional[str]
- especialidade: Optional[str]
- contato_email: str

Entidade **Modulo** que contém 4 atributos listados abaixo:

- id_modulo: Optional[int] (Primary Key)
- nome_modulo: str
- descricao: Optional[str]
- curso_id: Optional[int] (Foreign Key para Curso)

Entidade **Suporte** que contém 5 atributos listados abaixo:

- id_suporte: Optional[int] (Primary Key)
- aluno_id: Optional[int] (Foreign Key para Aluno)
- curso_id: Optional[int] (Foreign Key para Curso)
- data_abertura: datetime (Data de criação)
- descricao_problema: str

2. Introdução ao ORM e Configuração do Banco de Dados

SGBD utilizado: MySQL

3. Criar uma API REST com FastAPI

Usando FastAPI, vocês vão criar endpoints para implementar cada funcionalidade solicitada. Cada funcionalidade será implementada em um endpoint específico. Abaixo estão os detalhes de cada funcionalidade que a API deverá oferecer.

3. Funcionalidades da API

Funcionalidade	Responsável	Status
----------------	-------------	--------

F1: Inserir uma entidade no banco de dados	 	Concluído
F2: Listar todas as entidades do banco		Concluído
F3: CRUD completo da entidade		Concluído
F4: Mostrar a quantidade de entidades		ncluído
F5: Implementar paginação e limitação de resultados		Concluído
F6: Filtrar entidades por atributos específicos		Concluído
F7: Criar migrações com Alembic		Concluído
F8: Configurar logs para monitoramento		Concluído