**Documentação Técnica do Projeto de Banco de Dados**

**Introdução ao Projeto**

**Descrição do Projeto**

Este projeto foi desenvolvido para gerenciar um sistema acadêmico, permitindo o controle eficiente de alunos, professores, turmas, disciplinas, frequências e avaliações. O sistema possibilita o registro de presença, notas e matrículas, além de manter os dados de usuários de forma organizada. (REFORMULAR P/ NOVO MODELO DO PROJETO(ATUAL))

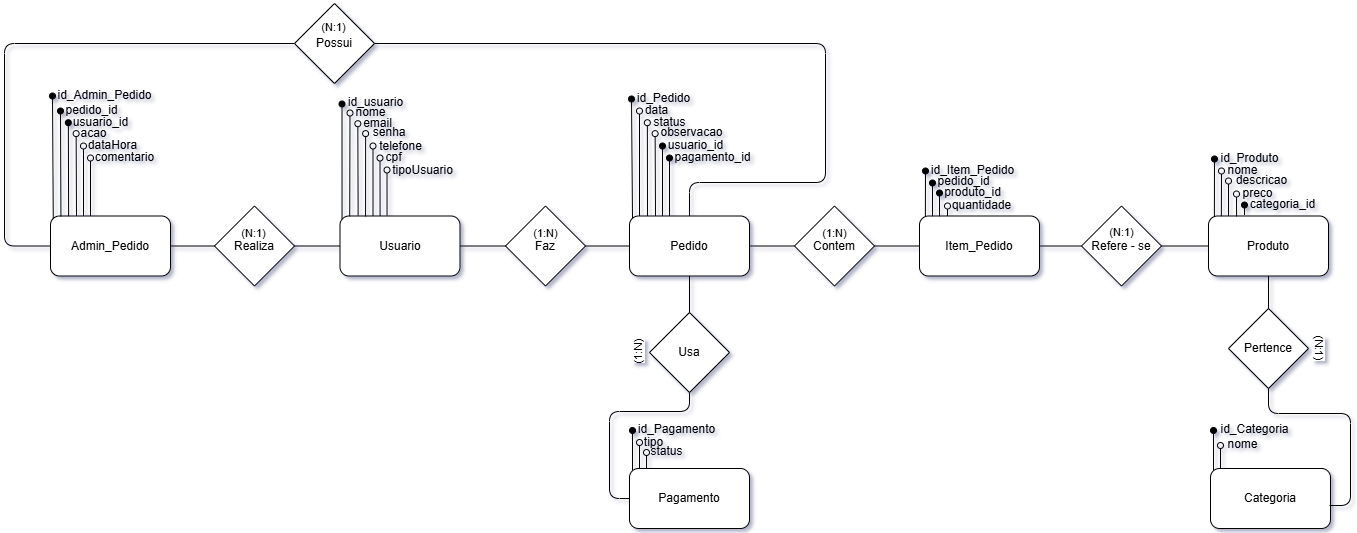
**Objetivo do Banco de Dados**

O objetivo do banco de dados é organizar e gerenciar as informações acadêmicas da instituição, como dados de usuários, alunos, professores, turmas, disciplinas, frequência e avaliações, garantindo integridade, consistência e eficiência no acesso e no processamento das informações educacionais. (REFORMULAR P/ NOVO MODELO DO PROJETO(ATUAL))

**Modelagem de Dados**

**Diagrama Entidade-Relacionamento (DER)**

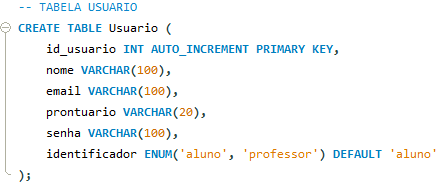
O Diagrama Entidade-Relacionamento (DER) ilustra as entidades e relacionamentos do sistema, como mostrado abaixo:

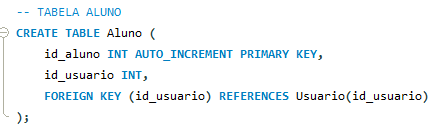


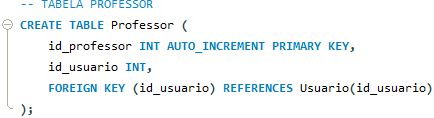
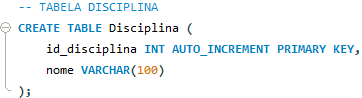
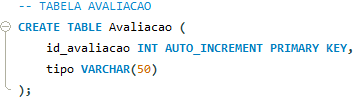
**Estrutura do Banco de Dados**

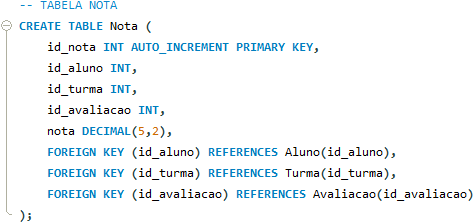
**Scripts de Criação do Banco de Dados**

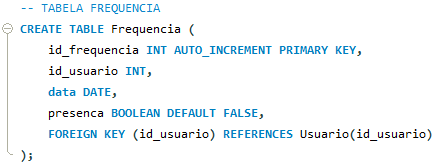
Os scripts SQL para criação das tabelas e seus relacionamentos são apresentados abaixo:







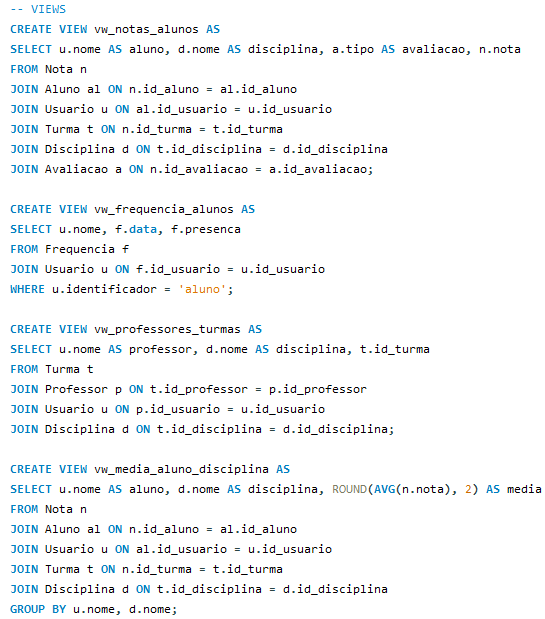


**Descrição das Tabelas**

* **Tabela Usuario**: Armazena informações dos usuários cadastrados no caderno.
  + id\_usuario: Identificador único do usuario.
  + nome: Nome completo do usuario.
  + email: Endereço de e-mail do usuario.
  + prontuario: Prontuário do usuário no sistema.
  + senha: Senha de cadastro no sistema.
  + identificador: Define se o usuário é aluno ou professor.
* **Tabela Aluno**: Relaciona - se com Usuario, específico para alunos.
  + id\_aluno: Primary Key (PK).
  + id\_usuario: Foreign Key (FK).
* **Tabela Professor**: Relaciona - se com Usuario, específico para professores.
  + id\_professor: Primary Key (PK).
  + id\_usuario: Foreign Key (FK).
* **Tabela Disciplina**: Contém as disciplinas oferecidas.
  + id\_disciplina: Primary Key (PK).
  + nome: Nome da disciplina.
* **Tabela Turma**: Vincula professores às disciplinas.
  + id\_turma: Primary Key (PK).
  + id\_disciplina: Foreign Key (FK).
  + id\_professor: Foreign Key (FK).
* **Tabela Avaliacao**: Define o tipo de avaliação.
  + id\_avaliacao: Primary Key (PK).
  + tipo: Tipo de avaliacao.
* **Tabela Nota**: Registro das notas dos alunos por turma e avaliação.
  + id\_nota: Primary Key (PK).
  + id\_aluno: Foreign Key (FK).
  + id\_turma: Foreign Key (FK).
  + id\_avaliacao: Foreign Key (FK).
  + nota: Valor da nota.
* **Tabela Frequencia**: Controla presença/falta dos alunos.
  + id\_frequencia: Primary Key (PK).
  + id\_usuario: Foreign Key (FK).
  + data: Define a data específica da frequência.
  + presenca: Define se estava presente ou não.
* **Relacionamentos**:
  + **Usuario** se relaciona **1:N** com **Aluno** e **Professor**;
  + **Professor** leciona várias **Turmas**;
  + **Turma** tem várias **Notas** e se relaciona com **Disciplina**;
  + **Nota** conecta **Aluno, Turma** e **Avaliacao**;
  + **Frequencia** registra presença dos **Usuarios**.

**Views**

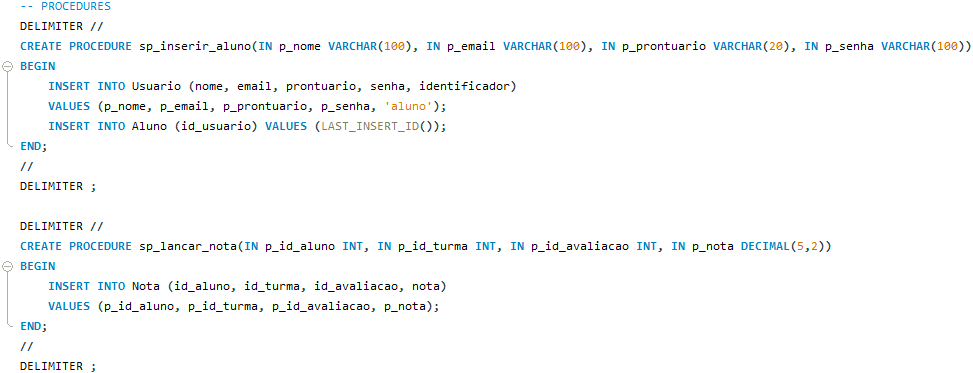
**Descrição das Views**



* **Views Criadas**:

Foram criadas 4 views para facilitar a consulta de dados importantes.

* + Facilitam a leitura e reutilização de consultas complexas, melhorando a organização e desempenho das consultas no sistema.
* **Views Criadas**:
  + **vw\_notas\_alunos:** Lista as notas de cada aluno por disicplina;
  + **vw\_frequencia\_alunos:** Mostra presença/falta por aluno;
  + **vw\_professores\_turmas:** Professores com suas turmas e disciplinas;
  + **vw\_media\_aluno\_disciplina:** Média geral de notas dos alunos.



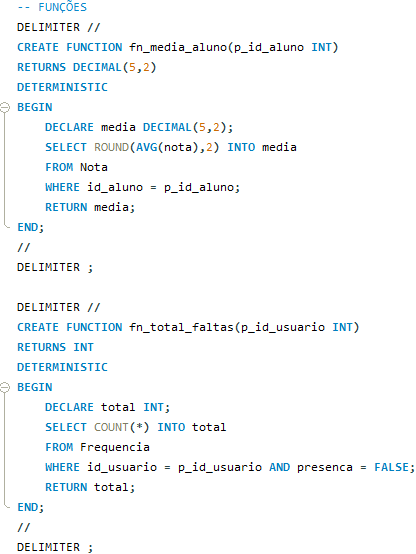
* **Procedures** :

Implementadas para automatizar tarefas comuns, como:

* + **Cadastrar aluno;**
  + **Lançar nota.**
* **Por que usar procedures?**

Elas encapsulam a lógica do sistema, promovendo reutilização, padronização e segurança dos dados.

* **Procedures Criadas**:
  + **sp\_inserir\_aluno:** Cadastra novo aluno;
  + **sp\_lancar\_nota:** Lança uma nova nota.



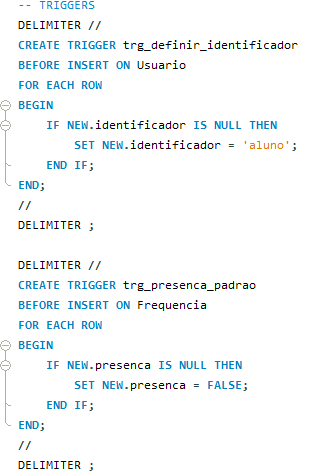
* **Functions** :

Implementadas para automatizar tarefas comuns, como:

* + **Calcular média;**
  + **Contar faltas.**
* **Por que usar funções?**

Elas encapsulam a lógica do sistema, promovendo reutilização, padronização e segurança dos dados.

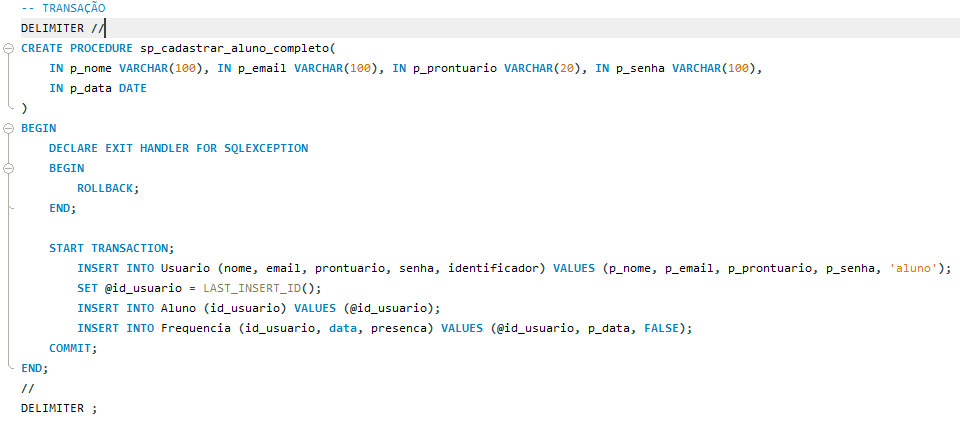
* **Functions Criadas**:
  + **fn\_media\_aluno:** Calcula média de um aluno;
  + **fn\_total\_faltas:** Conta faltas totais de um aluno.



* **Triggers**

Duas triggers foram criadas para automatizar comportamentos:

* + Preencher automaticamente o tipo de usuário como "aluno";
  + Definir a presença como "faltou" por padrão.
* **Por que usar triggers?**
  + Elas automatizam comportamentos esperados;
  + Reduz erros manuais e garante consistência de dados.
* **Triggers Criadas**:
  + **trg\_definir\_identificador:** Define o tipo de usuário como “aluno” por padrão;
  + **trg\_presenca\_padrao:** Define frequência como faltou (FALSE) por padrão.



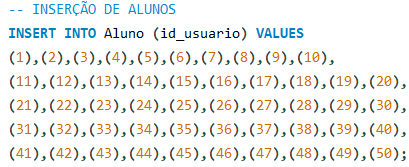
* **Transaction:**
  + Foi criada uma **transação** para garantir que o cadastro completo de um aluno (incluindo nota e frequência) ocorra de forma atômica: **ou tudo é inserido corretamente ou nada é inserido**.
* **Por que usar transações?**
  + Garante consistência total, especialmente em operações críticas com múltiplas tabelas;
* **Transaction Criada**:
  + **sp\_cadastrar\_aluno\_completo:** Registra aluno, nota e frequência em uma operação única e segura (transacional).

**Estrutura do Banco de Dados - (Dados Fictícios Inseridos)**

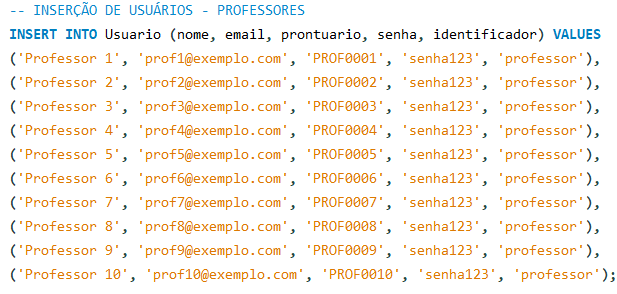
**Scripts de Criação do Banco de Dados e dados fictícios adicionados ao sistema.**

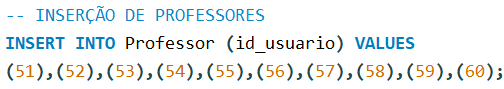
****

****

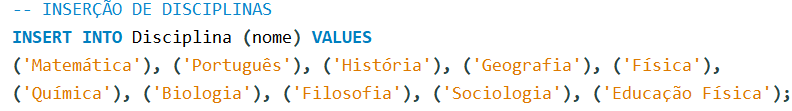
****

* **Inserção de dados - Aluno - na tabela Usuario**:
  + **idUsuario, nome, login, senha e perfil.**
* **Finalidade**:
  + Cadastrar usuários no sistema, para login e gerenciamento de permissões.

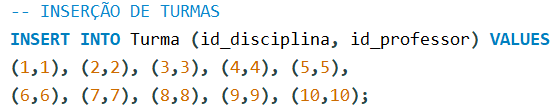
****

****

* **Inserção de dados - Professores - na tabela Usuario**:
  + **idUsuario;**
  + **nome;**
  + **login;**
  + **senha;**
  + **perfil.**
* **Finalidade**:
  + Cadastrar usuários no sistema, para login e gerenciamento de permissões.

****

* **Inserção de dados - Professores - na tabela Usuario**:
  + **Disciplina.**
* **Finalidade**:
  + Definir as disciplinas ministradas no sistema.

****

* **Inserção de dados na tabela Turma**:
  + **idUsuario;**
  + **descricao.**
* **Finalidade**:
  + Registrar as turmas disponíveis para disciplinas.

**Estrutura do Banco de Dados - (Visualização de dados)**

**Scripts de Criação para visualização do Banco de Dados e dos dados fictícios adicionados ao sistema.**

Os scripts SQL para criação das tabelas e seus relacionamentos são apresentados abaixo:



