



BACHARELADO EM CIÊNCIA E TECNOLOGIA

BANCO DE DADOS

Sistema de Gestão de Jornada Acadêmica

Eduardo Paulo Goes, RA: 156465, Integral
Rafael Pires Moreira Silva, RA: 163978, Noturno
José Vanderlei da Silva Júnior, RA: 163851, Noturno

Prof. Daniela Musa

São José dos Campos - SP

Dezembro - 2023

Introdução:

Este projeto foi idealizado para atender à necessidade de organizar e gerenciar a jornada acadêmica de estudantes de Ciência da Computação (ou de outros cursos) de forma integrada e eficiente. A proposta é desenvolver uma solução que centralize as principais tarefas de planejamento e acompanhamento acadêmico, substituindo o uso de planilhas e proporcionando uma visão completa do percurso universitário.

Entre os interesses dos estudantes estão:

- Planejar a grade horária para os próximos semestres;
- Monitorar o progresso acadêmico em comparação com a trajetória ideal recomendada pela instituição;
- Acompanhar o cronograma de cada disciplina, garantindo que datas de entrega de trabalhos e provas não sejam perdidas;
- Registrar, diariamente, os assuntos discutidos em cada aula, servindo de referência para a organização dos estudos.

Geralmente, nas universidades, cada curso possui uma trajetória ideal que define o conjunto de disciplinas obrigatórias necessárias para a conclusão do curso. Essa trajetória inclui informações sobre o semestre e o ano recomendados (a partir do ingresso do estudante) e o número de créditos de cada disciplina. Além das disciplinas obrigatórias, os cursos também exigem um número adicional de créditos obtidos por meio de disciplinas extras, que são aquelas que não integram a trajetória ideal.

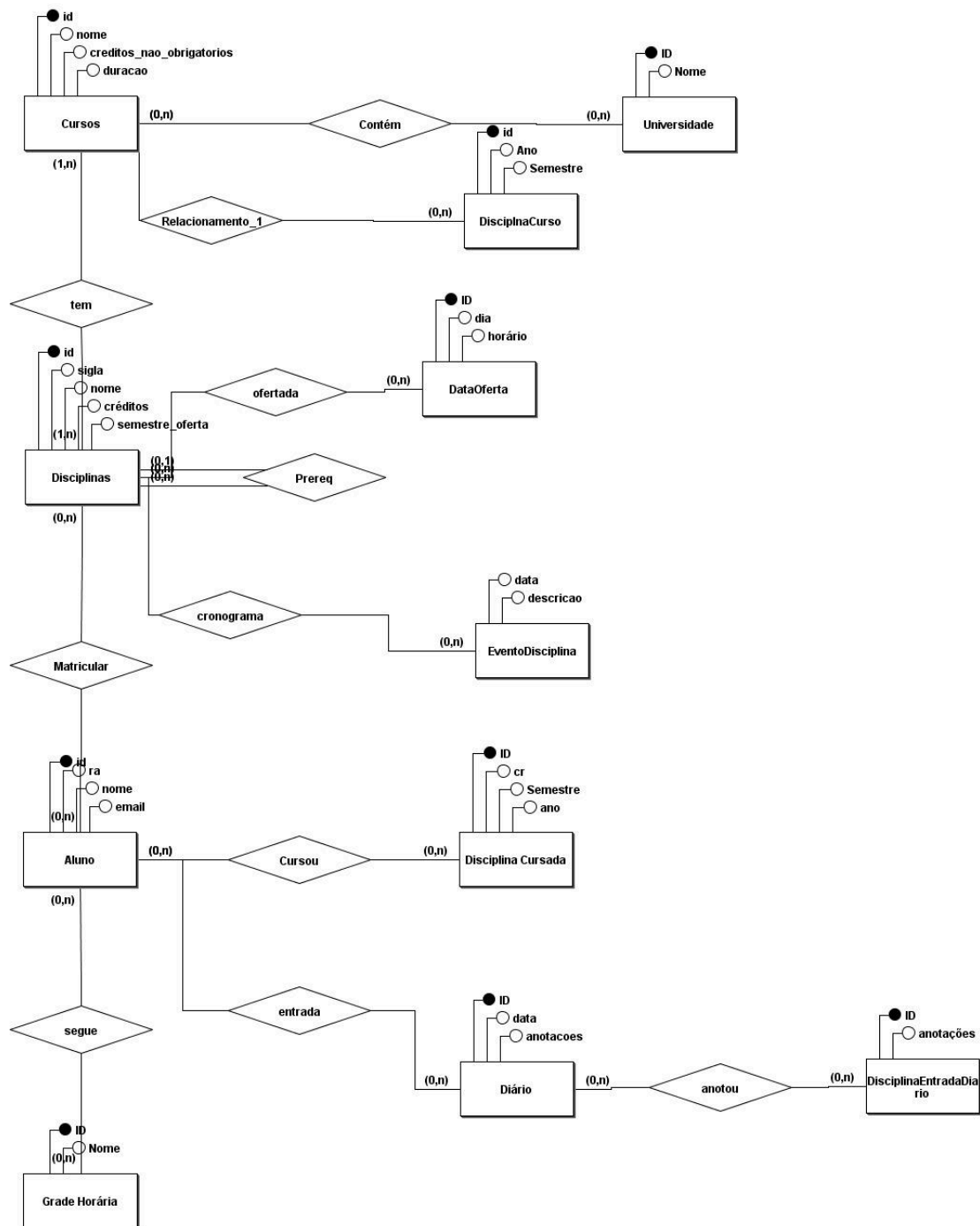
O banco de dados deverá contemplar detalhes essenciais, como a existência de pré-requisitos para determinadas disciplinas, a oferta das mesmas em semestres pares, ímpares ou ambos, e os créditos associados a cada uma. Algumas disciplinas podem contar com um cronograma composto por eventos que possuem data e descrição, facilitando o acompanhamento do andamento do curso.

Além disso, o sistema armazenará informações pessoais e acadêmicas dos estudantes, como número de matrícula, data de entrada (definida pelo semestre e ano), universidade, curso e a jornada acadêmica. Este registro inclui as disciplinas cursadas, com o coeficiente de rendimento (CR) obtido e o semestre/ano em que foram realizadas – considerando apenas aquelas disciplinas aprovadas ($CR \geq 6$).

No período entre semestres, quando as disciplinas podem ser ofertadas para matrícula, o sistema gerenciará essas ofertas, associando cada uma a um professor, a uma turma e aos respectivos horários (definidos pelo par horário e dia da semana). A partir das ofertas de um determinado ano e semestre, os estudantes terão a possibilidade de montar diversas grades horárias, preparando-se para cenários como o indeferimento de alguma disciplina na rematrícula.

Por fim, o sistema possibilitará que os estudantes registrem, diariamente, os assuntos discutidos em cada disciplina e armazenem anotações gerais, funcionando como um diário acadêmico que complementa o planejamento e o acompanhamento de sua trajetória.

Com isso, montamos o seguinte modelo de entidade relacional:



Além disso, montamos as seguintes tabelas que contêm todos os dados necessários para o sistema:

```
CREATE TABLE Universidade (  
    codigo_universidade INT PRIMARY KEY,  
    nome VARCHAR(100) NOT NULL  
);
```

```
CREATE TABLE Curso (  
    codigo_curso INT,  
    nome VARCHAR(100) NOT NULL,  
    creditos_extras INT NOT NULL,  
    codigo_universidade INT,  
    PRIMARY KEY (codigo_curso, codigo_universidade),  
    FOREIGN KEY (codigo_universidade) REFERENCES Universidade(codigo_universidade)  
ON DELETE CASCADE  
);
```

```
CREATE TABLE Disciplina (  
    codigo_disciplina INT PRIMARY KEY,  
    nome VARCHAR(100) NOT NULL,  
    creditos INT NOT NULL,  
    semestre_recomendado INT,  
    ano_recomendado INT,  
    oferta_semestre OFERTA NOT NULL,  
    codigo_curso INT,  
    codigo_universidade INT,  
    FOREIGN KEY (codigo_curso, codigo_universidade) REFERENCES Curso(codigo_curso,  
codigo_universidade) ON DELETE CASCADE  
);
```

```
CREATE TABLE PreRequisito (  
    codigo_disciplina INT,  
    codigo_pre_requisito INT,  
    PRIMARY KEY (codigo_disciplina, codigo_pre_requisito),  
    FOREIGN KEY (codigo_disciplina) REFERENCES Disciplina(codigo_disciplina) ON  
DELETE CASCADE,  
    FOREIGN KEY (codigo_pre_requisito) REFERENCES Disciplina(codigo_disciplina) ON  
DELETE CASCADE  
);
```

```
CREATE TABLE Evento (  
    codigo_evento INT PRIMARY KEY,  
    codigo_disciplina INT,  
    data DATE NOT NULL,  
    descricao TEXT,  
    FOREIGN KEY (codigo_disciplina) REFERENCES Disciplina(codigo_disciplina) ON  
DELETE CASCADE
```

);

```
CREATE TABLE Aluno (  
  ra INT PRIMARY KEY,  
  nome VARCHAR(100) NOT NULL,  
  semestre_entrada INT,  
  ano_entrada INT,  
  codigo_curso INT,  
  codigo_universidade INT,  
  FOREIGN KEY (codigo_curso, codigo_universidade) REFERENCES Curso(codigo_curso,  
  codigo_universidade) ON DELETE CASCADE  
);
```

```
CREATE TABLE DisciplinaCursada (  
  ra INT,  
  codigo_disciplina INT,  
  semestre_cursado INT,  
  ano_cursado INT,  
  cr DECIMAL(3,1),  
  PRIMARY KEY (ra, codigo_disciplina),  
  FOREIGN KEY (ra) REFERENCES Aluno(ra) ON DELETE CASCADE,  
  FOREIGN KEY (codigo_disciplina) REFERENCES Disciplina(codigo_disciplina) ON  
  DELETE CASCADE  
);
```

```
CREATE TABLE Professor (  
  codigo_professor INT PRIMARY KEY,  
  nome VARCHAR(100) NOT NULL  
);
```

```
CREATE TABLE OfertaMatricula (  
  codigo_oferta INT PRIMARY KEY,  
  codigo_disciplina INT,  
  codigo_professor INT,  
  semestre INT,  
  ano INT,  
  turma VARCHAR(10),  
  FOREIGN KEY (codigo_disciplina) REFERENCES Disciplina(codigo_disciplina) ON  
  DELETE CASCADE,  
  FOREIGN KEY (codigo_professor) REFERENCES Professor(codigo_professor) ON  
  DELETE CASCADE  
);
```

```
CREATE TABLE HorarioAula (  
  codigo_horario INT PRIMARY KEY,  
  codigo_oferta INT,  
  dia_semana DIASEMANA NOT NULL,  
  horario_inicio TIME,
```

```
    horario_fim TIME,  
    FOREIGN KEY (codigo_oferta) REFERENCES OfertaMatricula(codigo_oferta) ON  
DELETE CASCADE  
);
```

```
CREATE TABLE GradeHoraria (  
    codigo_grade INT PRIMARY KEY,  
    ra INT,  
    semestre INT,  
    ano INT,  
    descricao VARCHAR(200),  
    FOREIGN KEY (ra) REFERENCES Aluno(ra)  
);
```

```
CREATE TABLE DisciplinaGrade (  
    codigo_grade INT,  
    codigo_oferta INT,  
    PRIMARY KEY (codigo_grade, codigo_oferta),  
    FOREIGN KEY (codigo_grade) REFERENCES GradeHoraria(codigo_grade) ON DELETE  
CASCADE,  
    FOREIGN KEY (codigo_oferta) REFERENCES OfertaMatricula(codigo_oferta) ON  
DELETE CASCADE  
);
```

```
CREATE TABLE AnotacaoDiaria (  
    codigo_anotacao INT PRIMARY KEY,  
    ra INT,  
    data DATE,  
    codigo_disciplina INT,  
    assunto VARCHAR(200),  
    anotacao TEXT,  
    FOREIGN KEY (ra) REFERENCES Aluno(ra) ON DELETE CASCADE,  
    FOREIGN KEY (codigo_disciplina) REFERENCES Disciplina(codigo_disciplina) ON  
DELETE CASCADE  
);
```

```
INSERT INTO Universidade (codigo_universidade, nome) VALUES  
(1, 'Universidade Federal de São Paulo'),  
(2, 'Universidade de São Paulo'),  
(3, 'Universidade de São Paulo'),  
(4, 'Instituto Federal de São Paulo'),  
(5, 'Universidade do Vale do Paraíba'),  
(6, 'Universidade Federal de Minas Gerais'),  
(7, 'Universidade de Curitiba'),  
(8, 'Universidade de Cuiabá'),
```

(9, 'Universidade Federal Fluminense'),
(10, 'Universidade Estadual do Rio de Janeiro');

```
INSERT INTO Curso (codigo_curso, nome, creditos_extras, codigo_universidade) VALUES  
(101, 'Ciência da Computação', 20, 1),  
(102, 'Engenharia de Software', 18, 1),  
(103, 'Engenharia de Computação', 22, 2),  
(104, 'Sistemas de Informação', 18, 3),  
(105, 'Análise e Desenvolvimento de Sistemas', 16, 4),  
(106, 'Ciência de Dados', 20, 5),  
(107, 'Segurança da Informação', 18, 6),  
(107, 'Segurança da Informação', 18, 7),  
(107, 'Segurança da Informação', 18, 8),  
(107, 'Segurança da Informação', 18, 9);
```

```
INSERT INTO Disciplina (codigo_disciplina, nome, creditos, semestre_recomendado,  
ano_recomendado, oferta_semestre, codigo_curso) VALUES  
(1001, 'Banco de Dados', 4, 3, 2, 'ambos', 101),  
(1002, 'Algoritmos', 4, 1, 1, 'impar', 101),  
(1003, 'Redes', 4, 4, 2, 'par', 101),  
(1004, 'Inteligência Artificial', 4, NULL, NULL, 'ambos', NULL),  
(1005, 'Segurança da Informação', 4, NULL, NULL, 'par', NULL),  
(1006, 'Programação Web', 4, 5, 3, 'ambos', 101),  
(1007, 'Machine Learning', 4, 6, 3, 'par', 103),  
(1008, 'Desenvolvimento Mobile', 4, 4, 2, 'impar', 104),  
(1009, 'Cloud Computing', 4, 5, 3, 'ambos', 105),  
(1010, 'Ética em TI', 2, 3, 2, 'par', 106);
```

```
INSERT INTO PreRequisito (codigo_disciplina, codigo_pre_requisito) VALUES  
(1001, 1002),  
(1006, 1001),  
(1007, 1002),  
(1008, 1003),  
(1009, 1001),  
(1003, 1002),  
(1007, 1010),  
(1009, 1007),  
(1005, 1010),  
(1010, 1002);
```

```
INSERT INTO Evento (codigo_evento, codigo_disciplina, data, descricao) VALUES  
(1, 1001, '2024-03-15', 'Prova 1 - Modelo Relacional'),  
(2, 1002, '2024-04-20', 'Entrega do Trabalho Prático'),  
(3, 1002, '2024-03-20', 'Prova 1 - Complexidade de Algoritmos'),  
(4, 1003, '2024-03-25', 'Aula prática de configuração de redes'),  
(5, 1006, '2024-04-01', 'Entrega do projeto de front-end'),  
(6, 1007, '2024-04-10', 'Apresentação de modelos de ML'),
```

```
(7, 1008, '2024-04-15', 'Demonstração de aplicativo mobile'),  
(8, 1001, '2024-04-20', 'Entrega do trabalho final'),  
(9, 1007, '2024-04-12', 'Entrega do relatório'),  
(10, 1008, '2024-05-07', 'Prova final');
```

```
INSERT INTO Aluno (ra, nome, semestre_entrada, ano_entrada, codigo_curso,  
codigo_universidade) VALUES  
(156465, 'Eduardo Paulo Goes', 1, 2023, 101, 1),  
(163978, 'Rafael Pires', 2, 2023, 101, 1),  
(163851, 'José Vanderlei', 1, 2023, 101, 1),  
(170001, 'Maria Oliveira', 1, 2024, 103, 2),  
(170002, 'João Santos', 1, 2024, 104, 3),  
(170003, 'Ana Costa', 1, 2024, 105, 4),  
(170004, 'Pedro Almeida', 1, 2024, 106, 5),  
(170005, 'Carla Souza', 1, 2024, 107, 6),  
(170006, 'Luide Mattos', 1, 2024, 106, 5),  
(170007, 'Fernando Moura Bolzano', 1, 2024, 107, 6);
```

```
INSERT INTO DisciplinaCursada (ra, codigo_disciplina, semestre_cursado, ano_cursado,  
cr) VALUES  
(156465, 1002, 1, 2023, 8.5),  
(163978, 1001, 1, 2023, 7.5),  
(163851, 1002, 1, 2023, 8.0),  
(156465, 1003, 2, 2023, 9.0),  
(170001, 1002, 1, 2024, 7.0),  
(170002, 1003, 1, 2024, 8.5),  
(170003, 1001, 1, 2024, 9.0),  
(170004, 1006, 1, 2024, 8.0),  
(170006, 1007, 1, 2024, 2.5),  
(170007, 1007, 1, 2024, 6.3);
```

```
INSERT INTO Professor (codigo_professor, nome) VALUES  
(1, 'Carlos Silva'),  
(2, 'Ana Paula Rocha'),  
(3, 'Fernando Torres'),  
(4, 'Roberto Alves'),  
(5, 'Patrícia Lima'),  
(6, 'Marcos Rocha'),  
(7, 'Juliana Fernandes'),  
(8, 'Oliver Kolonossoki'),  
(9, 'Gabriel Haeser'),  
(10, 'Ricardo Oliveira');
```

```
INSERT INTO OfertaMatricula (codigo_oferta, codigo_disciplina, codigo_professor,  
semestre, ano, turma) VALUES  
(101, 1001, 1, 1, 2024, 'T01'),  
(102, 1002, 2, 1, 2024, 'T02'),  
(103, 1003, 3, 1, 2024, 'T03'),
```



```
(104, 1006, 4, 1, 2024, 'T04'),  
(105, 1007, 5, 1, 2024, 'T05'),  
(106, 1008, 6, 1, 2024, 'T06'),  
(107, 1009, 7, 1, 2024, 'T07'),  
(108, 1010, 8, 1, 2024, 'T08'),  
(109, 1004, 9, 1, 2024, 'T09'),  
(110, 1005, 10, 1, 2024, 'T10');
```

```
INSERT INTO HorarioAula (codigo_horario, codigo_oferta, dia_semana, horario_inicio,  
horario_fim) VALUES
```

```
(201, 101, 'seg', '14:00', '16:00'),  
(202, 101, 'qua', '14:00', '16:00'),  
(203, 102, 'ter', '08:00', '10:00'),  
(204, 103, 'qui', '10:00', '12:00'),  
(205, 104, 'seg', '10:00', '12:00'),  
(206, 105, 'ter', '14:00', '16:00'),  
(207, 106, 'qua', '08:00', '10:00'),  
(208, 107, 'qui', '16:00', '18:00'),  
(209, 108, 'sex', '09:00', '11:00'),  
(210, 109, 'sex', '10:00', '12:00');
```

```
INSERT INTO GradeHoraria (codigo_grade, ra, semestre, ano, descricao) VALUES
```

```
(1, 156465, 1, 2024, 'Grade Principal'),  
(2, 163978, 1, 2024, 'Grade Alternativa'),  
(3, 170001, 1, 2024, 'Grade Principal - Engenharia'),  
(4, 170002, 1, 2024, 'Grade Alternativa - Sistemas'),  
(5, 170003, 1, 2024, 'Grade Principal - ADS'),  
(6, 170004, 1, 2024, 'Grade Principal - Ciência de Dados'),  
(7, 170005, 1, 2024, 'Grade Principal - Segurança'),  
(8, 170006, 1, 2024, 'Grade Principal - ADS'),  
(9, 170007, 1, 2024, 'Grade Alternativa - Sistemas')  
(10, 170006, 1, 2024, 'Grade Principal - Redes');
```

```
INSERT INTO DisciplinaGrade (codigo_grade, codigo_oferta) VALUES
```

```
(1, 101),  
(1, 102),  
(2, 103),  
(3, 104),  
(4, 105),  
(5, 106),  
(6, 107),  
(7, 108),  
(8, 109),  
(9, 110);
```

```
INSERT INTO AnotacaoDiaria (codigo_anotacao, ra, data, codigo_disciplina, assunto,  
anotacao) VALUES
```

(1, 156465, '2024-03-10', 1001, 'Normalização', 'Revisar formas normais até 3FN'),
(2, 163978, '2024-03-11', 1003, 'TCP/IP', 'Praticar configuração de redes locais'),
(3, 170001, '2024-03-12', 1002, 'Algoritmos de ordenação', 'Revisar QuickSort e MergeSort'),
(4, 170002, '2024-03-13', 1003, 'Protocolos de rede', 'Estudar TCP/IP e UDP'),
(5, 170003, '2024-03-14', 1001, 'Modelo relacional', 'Praticar normalização'),
(6, 170004, '2024-03-15', 1006, 'HTML/CSS', 'Criar layout responsivo'),
(7, 170005, '2024-03-16', 1007, 'Regressão linear', 'Implementar em Python'),
(8, 170006, '2024-03-12', 1007, 'SVM', 'Treinar o modelo'),
(9, 163851, '2024-02-20', 1001, 'SQL', 'Terminar o trabalho'),
(10, 163978, '2024-02-20', 1001, 'SQL', 'Terminar o trabalho');

tabela alunos

	ra [PK] integer	nome character varying (100)	semestre_entrada integer	ano_entrada integer	codigo_curso integer	codigo_universidade integer
1	156465	Eduardo Paulo Goes	1	2023	101	1
2	163851	José Vanderlei	1	2023	101	1
3	163978	Rafael Pires	2	2023	101	1
4	170001	Maria Oliveira	1	2024	103	2
5	170002	João Santos	1	2024	104	3
6	170003	Ana Costa	1	2024	105	4
7	170004	Pedro Almeida	1	2024	106	5
8	170005	Carla Souza	1	2024	107	6
9	170006	Luide Mattos	1	2024	106	5
10	170007	Fernando Moura Bolzano	1	2024	107	6

tabela anotação diaria

	codigo_anotacao [PK] integer	ra integer	data date	codigo_disciplina integer	assunto character varying (200)	anotacao text
1	1	156465	2024-03-10	1001	Normalização	Revisar formas normais até 3FN
2	2	163978	2024-03-11	1003	TCP/IP	Praticar configuração de redes locais
3	3	170001	2024-03-12	1002	Algoritmos de ordenação	Revisar QuickSort e MergeSort
4	4	170002	2024-03-13	1003	Protocolos de rede	Estudar TCP/IP e UDP
5	5	170003	2024-03-14	1001	Modelo relacional	Praticar normalização
6	6	170004	2024-03-15	1006	HTML/CSS	Criar layout responsivo
7	7	170005	2024-03-16	1007	Regressão linear	Implementar em Python
8	8	170006	2024-03-12	1007	SVM	Treinar o modelo
9	9	163851	2024-02-20	1001	SQL	Terminar o trabalho
10	10	163978	2024-02-20	1001	SQL	Terminar o trabalho

tabela curso

	codigo_curso [PK] integer	nome character varying (100)	creditos_extras integer	codigo_universidade [PK] integer
1	101	Ciência da Computação	20	1
2	102	Engenharia de Software	18	1
3	103	Engenharia de Computação	22	2
4	104	Sistemas de Informação	18	3
5	105	Análise e Desenvolvimento de Sistemas	16	4
6	106	Ciência de Dados	20	5
7	107	Segurança da Informação	18	6
8	107	Segurança da Informação	18	7
9	107	Segurança da Informação	18	8
10	107	Segurança da Informação	18	9

tabela disciplina

	codigo_disciplina [PK] integer	nome character varying (100)	creditos integer	semestre_recomendado integer	ano_recomendado integer	oferta_semestre oferta	codigo_curso integer
1	1001	Banco de Dados	4	3	2	ambos	101
2	1002	Algoritmos	4	1	1	impar	101
3	1003	Redes	4	4	2	par	101
4	1004	Inteligência Artificial	4	[null]	[null]	ambos	[null]
5	1005	Segurança da Informação	4	[null]	[null]	par	[null]
6	1006	Programação Web	4	5	3	ambos	101
7	1007	Machine Learning	4	6	3	par	103
8	1008	Desenvolvimento Mobile	4	4	2	impar	104
9	1009	Cloud Computing	4	5	3	ambos	105
10	1010	Ética em TI	2	3	2	par	106

tabela disciplinacursada

	ra [PK] integer	codigo_disciplina [PK] integer	semestre_cursado integer	ano_cursado integer	cr numeric (3,1)
1	156465	1002	1	2023	8.5
2	156465	1003	2	2023	9.0
3	163851	1002	1	2023	8.0
4	163978	1001	1	2023	7.5
5	170001	1002	1	2024	7.0
6	170002	1003	1	2024	8.5
7	170003	1001	1	2024	9.0
8	170004	1006	1	2024	8.0
9	170006	1007	1	2024	2.5
10	170007	1007	1	2024	6.3

tabela disciplinagrade

	codigo_grade [PK] integer	codigo_oferta [PK] integer
1	1	101
2	1	102
3	2	103
4	3	104
5	4	105
6	5	106
7	6	107
8	7	108
9	8	109
10	9	110

tabela evento

	codigo_evento [PK] integer	codigo_disciplina integer	data date	descricao text
1	1	1001	2024-03-15	Prova 1 - Modelo Relacional
2	2	1002	2024-04-20	Entrega do Trabalho Prático
3	3	1002	2024-03-20	Prova 1 - Complexidade de Algoritm...
4	4	1003	2024-03-25	Aula prática de configuração de redes
5	5	1006	2024-04-01	Entrega do projeto de front-end
6	6	1007	2024-04-10	Apresentação de modelos de ML
7	7	1008	2024-04-15	Demonstração de aplicativo mobile
8	8	1001	2024-04-20	Entrega do trabalho final
9	9	1007	2024-04-12	Entrega do relatório
10	10	1008	2024-05-07	Prova final

tabela gradehoraria

	codigo_grade [PK] integer	ra integer	semestre integer	ano integer	descricao character varying (200)
1	1	156465	1	2024	Grade Principal
2	2	163978	1	2024	Grade Alternativa
3	3	170001	1	2024	Grade Principal - Engenharia
4	4	170002	1	2024	Grade Alternativa - Sistemas
5	5	170003	1	2024	Grade Principal - ADS
6	6	170004	1	2024	Grade Principal - Ciência de Dados
7	7	170005	1	2024	Grade Principal - Segurança
8	8	170006	1	2024	Grade Principal - ADS
9	9	170007	1	2024	Grade Alternativa - Sistemas
10	10	170006	1	2024	Grade Principal - Redes

tabela horarioaula

	codigo_horario [PK] integer	codigo_oferta integer	dia_semana diasemana	horario_inicio time without time zone	horario_fim time without time zone
1	201	101	seg	14:00:00	16:00:00
2	202	101	qua	14:00:00	16:00:00
3	203	102	ter	08:00:00	10:00:00
4	204	103	qui	10:00:00	12:00:00
5	205	104	seg	10:00:00	12:00:00
6	206	105	ter	14:00:00	16:00:00
7	207	106	qua	08:00:00	10:00:00
8	208	107	qui	16:00:00	18:00:00
9	209	108	sex	09:00:00	11:00:00
10	210	109	sex	10:00:00	12:00:00

tabela ofertametricula

	codigo_oferta [PK] integer	codigo_disciplina integer	codigo_professor integer	semestre integer	ano integer	turma character varying (10)
1	101	1001	1	1	2024	T01
2	102	1002	2	1	2024	T02
3	103	1003	3	1	2024	T03
4	104	1006	4	1	2024	T04
5	105	1007	5	1	2024	T05
6	106	1008	6	1	2024	T06
7	107	1009	7	1	2024	T07
8	108	1010	8	1	2024	T08
9	109	1004	9	1	2024	T09
10	110	1005	10	1	2024	T10
11	111	1010	10	1	2024	T11

tabela prerequisite

	codigo_disciplina [PK] integer	codigo_pre_requisito [PK] integer
1	1001	1002
2	1003	1002
3	1005	1010
4	1006	1001
5	1007	1002
6	1007	1010
7	1008	1003
8	1009	1001
9	1009	1007
10	1010	1002

tabela professor



	codigo_professor [PK] integer	nome character varying (100)
1	1	Carlos Silva
2	2	Ana Paula Rocha
3	3	Fernando Torres
4	4	Roberto Alves
5	5	Patrícia Lima
6	6	Marcos Rocha
7	7	Juliana Fernandes
8	8	Oliver Kolonossoki
9	9	Gabriel Haeser
10	10	Ricardo Oliveira

tabela universidade

	codigo_universidade [PK] integer	nome character varying (100)
1	1	Universidade Federal de São Paulo
2	2	Universidade de São Paulo
3	3	Universidade de São Paulo
4	4	Instituto Federal de São Paulo
5	5	Universidade do Vale do Paraíba
6	6	Universidade Federal de Minas Gerais
7	7	Universidade de Curitiba
8	8	Universidade de Cuiabá
9	9	Universidade Federal Fluminense
10	10	Universidade Estadual do Rio de Janeiro

-- 1. Alunos e seus cursos

```
SELECT a.nome, c.nome AS curso
FROM Aluno a
INNER JOIN Curso c
ON a.codigo_curso = c.codigo_curso
AND a.codigo_universidade = c.codigo_universidade;
```

	nome character varying (100) 	curso character varying (100) 
1	Eduardo Paulo Goes	Ciência da Computação
2	Rafael Pires	Ciência da Computação
3	José Vanderlei	Ciência da Computação
4	Maria Oliveira	Engenharia de Computação
5	João Santos	Sistemas de Informação
6	Ana Costa	Análise e Desenvolvimento de Sistemas
7	Pedro Almeida	Ciência de Dados
8	Carla Souza	Segurança da Informação
9	Luide Mattos	Ciência de Dados
10	Fernando Moura Bolzano	Segurança da Informação

-- 2. Disciplinas com pré-requisitos

```
SELECT d.nome AS disciplina, p.nome AS pre_requisito
FROM PreRequisito pr
INNER JOIN Disciplina d ON pr.codigo_disciplina = d.codigo_disciplina
INNER JOIN Disciplina p ON pr.codigo_pre_requisito = p.codigo_disciplina;
```


	disciplina character varying (100) 🔒	pre_requisito character varying (100) 🔒
1	Banco de Dados	Algoritmos
2	Programação Web	Banco de Dados
3	Machine Learning	Algoritmos
4	Desenvolvimento Mobile	Redes
5	Cloud Computing	Banco de Dados
6	Redes	Algoritmos
7	Machine Learning	Ética em TI
8	Cloud Computing	Machine Learning
9	Segurança da Informação	Ética em TI
10	Ética em TI	Algoritmos

-- 3. Disciplinas cursadas e notas dos alunos

SELECT d.nome, cd.cr, a.nome

FROM DisciplinaCursada cd

INNER JOIN Disciplina d ON cd.codigo_disciplina = d.codigo_disciplina

INNER JOIN Aluno a ON a.ra = cd.ra;

	nome character varying (100) 🔒	cr numeric (3,1) 🔒	nome character varying (100) 🔒
1	Algoritmos	8.5	Eduardo Paulo Goes
2	Banco de Dados	7.5	Rafael Pires
3	Algoritmos	8.0	José Vanderlei
4	Redes	9.0	Eduardo Paulo Goes
5	Algoritmos	7.0	Maria Oliveira
6	Redes	8.5	João Santos
7	Banco de Dados	9.0	Ana Costa
8	Programação Web	8.0	Pedro Almeida
9	Machine Learning	2.5	Luide Mattos
10	Machine Learning	6.3	Fernando Moura Bolzano

-- 4. Ofertas com horários e nome da disciplina

SELECT o.semestre, o.ano, h.dia_semana, h.horario_inicio, d.nome

FROM OfertaMatricula o

LEFT JOIN HorarioAula h ON o.codigo_oferta = h.codigo_oferta

LEFT JOIN Disciplina d ON o.codigo_disciplina = d.codigo_disciplina;

	semestre integer	ano integer	dia_semana diasemana	horario_inicio time without time zone	nome character varying (100)
1	1	2024	seg	14:00:00	Banco de Dados
2	1	2024	qua	14:00:00	Banco de Dados
3	1	2024	ter	08:00:00	Algoritmos
4	1	2024	qui	10:00:00	Redes
5	1	2024	seg	10:00:00	Programação Web
6	1	2024	ter	14:00:00	Machine Learning
7	1	2024	qua	08:00:00	Desenvolvimento Mobile
8	1	2024	qui	16:00:00	Cloud Computing
9	1	2024	sex	09:00:00	Ética em TI
10	1	2024	sex	10:00:00	Inteligência Artificial
11	1	2024	[null]	[null]	Segurança da Informação
12	1	2024	[null]	[null]	Ética em TI

-- 5. Grades horárias recomendadas e matérias de todos os alunos

SELECT g.descricao, d.nome AS nome_disciplina, a.nome AS nome_aluno

FROM GradeHoraria g

INNER JOIN DisciplinaGrade dg ON g.codigo_grade = dg.codigo_grade

INNER JOIN OfertaMatricula o ON dg.codigo_oferta = o.codigo_oferta

INNER JOIN Disciplina d ON o.codigo_disciplina = d.codigo_disciplina

INNER JOIN Aluno a ON a.ra = g.ra;

	descricao character varying (200) 🔒	nome_disciplina character varying (100) 🔒	nome_aluno character varying (100) 🔒
1	Grade Principal	Banco de Dados	Eduardo Paulo Goes
2	Grade Principal	Algoritmos	Eduardo Paulo Goes
3	Grade Alternativa	Redes	Rafael Pires
4	Grade Principal - Engenharia	Programação Web	Maria Oliveira
5	Grade Alternativa - Sistemas	Machine Learning	João Santos
6	Grade Principal - ADS	Desenvolvimento Mobile	Ana Costa
7	Grade Principal - Ciência de Dados	Cloud Computing	Pedro Almeida
8	Grade Principal - Segurança	Ética em TI	Carla Souza
9	Grade Principal - ADS	Inteligência Artificial	Luide Mattos
10	Grade Alternativa - Sistemas	Segurança da Informação	Fernando Moura Bolzano

-- 6. Total de créditos por aluno

```
SELECT a.nome, a.ra, SUM(d.creditos) AS total_creditos
FROM DisciplinaCursada cd
INNER JOIN Disciplina d ON cd.codigo_disciplina = d.codigo_disciplina
INNER JOIN Aluno a ON cd.ra = a.ra
GROUP BY a.ra;
```

	nome character varying (100) ✎	ra [PK] integer ✎	total_creditos bigint 🔒
1	José Vanderlei	163851	4
2	Eduardo Paulo Goes	156465	8
3	Pedro Almeida	170004	4
4	Luide Mattos	170006	4
5	Ana Costa	170003	4
6	Rafael Pires	163978	4
7	João Santos	170002	4
8	Fernando Moura Bolzano	170007	4
9	Maria Oliveira	170001	4

-- 7. Disciplinas que o aluno RA 156465 não cursou ainda

```
SELECT d.nome
FROM Disciplina d
```

```
LEFT JOIN DisciplinaCursada cd ON d.codigo_disciplina = cd.codigo_disciplina AND cd.ra = 156465
WHERE cd.codigo_disciplina IS NULL AND d.codigo_curso = 101;
```

	nome character varying (100) 🔒
1	Banco de Dados
2	Programação Web

-- 8. Eventos de disciplinas

```
SELECT e.data, e.descricao, d.nome
```

```
FROM Evento e
```

```
INNER JOIN Disciplina d ON d.codigo_disciplina = e.codigo_disciplina;
```

	data date 🔒	descricao text 🔒	nome character varying (100) 🔒
1	2024-03-15	Prova 1 - Modelo Relacional	Banco de Dados
2	2024-04-20	Entrega do Trabalho Prático	Algoritmos
3	2024-03-20	Prova 1 - Complexidade de Algoritm...	Algoritmos
4	2024-03-25	Aula prática de configuração de redes	Redes
5	2024-04-01	Entrega do projeto de front-end	Programação Web
6	2024-04-10	Apresentação de modelos de ML	Machine Learning
7	2024-04-15	Demonstração de aplicativo mobile	Desenvolvimento Mobile
8	2024-04-20	Entrega do trabalho final	Banco de Dados
9	2024-04-12	Entrega do relatório	Machine Learning
10	2024-05-07	Prova final	Desenvolvimento Mobile

-- 9. Anotações por disciplina

```
SELECT a.data, a.assunto, a.anotacao, d.nome
```

```
FROM AnotacaoDiaria a
```

```
INNER JOIN Disciplina d ON a.codigo_disciplina = d.codigo_disciplina;
```

	data date	assunto character varying (200)	anotacao text	nome character varying (100)
1	2024-03-10	Normalização	Revisar formas normais até 3FN	Banco de Dados
2	2024-03-11	TCP/IP	Praticar configuração de redes locais	Redes
3	2024-03-12	Algoritmos de ordenação	Revisar QuickSort e MergeSort	Algoritmos
4	2024-03-13	Protocolos de rede	Estudar TCP/IP e UDP	Redes
5	2024-03-14	Modelo relacional	Praticar normalização	Banco de Dados
6	2024-03-15	HTML/CSS	Criar layout responsivo	Programação Web
7	2024-03-16	Regressão linear	Implementar em Python	Machine Learning
8	2024-03-12	SVM	Treinar o modelo	Machine Learning
9	2024-02-20	SQL	Terminar o trabalho	Banco de Dados
10	2024-02-20	SQL	Terminar o trabalho	Banco de Dados

-- 10. Professores e ofertas

SELECT p.nome, COUNT(o.codigo_oferta) AS total_ofertas

FROM Professor p



LEFT JOIN OfertaMatricula o ON p.codigo_professor = o.codigo_professor

GROUP BY p.nome

HAVING COUNT(o.codigo_oferta) >= 1;

	nome character varying (100)	total_ofertas bigint
1	Carlos Silva	1
2	Ana Paula Rocha	1
3	Gabriel Haeser	1
4	Patrícia Lima	1
5	Marcos Rocha	1
6	Fernando Torres	1
7	Juliana Fernandes	1
8	Roberto Alves	1
9	Ricardo Oliveira	2
10	Oliver Kolonossoki	1

```
-- 11: Alunos que completaram todos os créditos obrigatórios do curso
SELECT a.ra, SUM(d.creditos) AS creditos_cursados
FROM DisciplinaCursada cd
INNER JOIN Disciplina d ON cd.codigo_disciplina = d.codigo_disciplina
INNER JOIN Aluno a ON cd.ra = a.ra
WHERE d.codigo_curso = a.codigo_curso -- Apenas disciplinas da trajetória do curso
GROUP BY a.ra
HAVING SUM(d.creditos) >= (
    SELECT SUM(d2.creditos) FROM Disciplina d2 WHERE d2.codigo_curso =
a.codigo_curso
);
```

	ra [PK] integer 	creditos_cursados bigint 
--	--	---

vazio

```
-- 12. Disciplinas mais populares
SELECT d.nome, COUNT(cd.ra) AS alunos_cursando
FROM DisciplinaCursada cd
INNER JOIN Disciplina d ON cd.codigo_disciplina = d.codigo_disciplina
GROUP BY d.nome
ORDER BY COUNT(cd.ra) DESC;
```

	nome character varying (100) 🔒	alunos_cursando bigint 🔒
1	Algoritmos	3
2	Banco de Dados	2
3	Machine Learning	2
4	Redes	2
5	Programação Web	1