## PROJETO INTEGRADOR

**Senac:** Administrador de Banco de Dados

**Professora:** Josenilda Aguiar

**Aluno:** Aluizio Monteiro

**Sumário**

[PROJETO INTEGRADOR 1](#_Toc53398202)

[Mini mundo 4](#_Toc53398203)

[Pagamento 4](#_Toc53398204)

[Parcelas 4](#_Toc53398205)

[Plano 4](#_Toc53398206)

[Aluno 4](#_Toc53398207)

[Aula 4](#_Toc53398208)

[Tópico 5](#_Toc53398209)

[Forum 5](#_Toc53398210)

[Respostas 5](#_Toc53398211)

[Curso 5](#_Toc53398212)

[Professor 5](#_Toc53398213)

[Carreira 5](#_Toc53398214)

[Exercicio 5](#_Toc53398215)

[Alternativa 5](#_Toc53398216)

[Modelo Conceitual 6](#_Toc53398217)

[Modelo Lógico 7](#_Toc53398218)

[Script 8](#_Toc53398219)

[/\*Criando o schema\*/ 8](#_Toc53398220)

[/\*Tabela plano\*/ 8](#_Toc53398221)

[/\*Tabela curso\*/ 8](#_Toc53398222)

[/\*Tabela aula\*/ 8](#_Toc53398223)

[/\*Tabela aluno\*/ 8](#_Toc53398224)

[/\*Tabela aluno\_aula\*/ 8](#_Toc53398225)

[/\*Tabela carreira\*/ 9](#_Toc53398226)

[/\*Tabela curso\_carreira\*/ 9](#_Toc53398227)

[/\*Tabela professor\*/ 9](#_Toc53398228)

[/\*Tabela forum\*/ 9](#_Toc53398229)

[/\*Tabela pagamento\*/ 9](#_Toc53398230)

[/\*Tabela respostas\*/ 10](#_Toc53398231)

[/\*Tabela professor\_curso\*/ 10](#_Toc53398232)

[/\*Tabela parcelas\*/ 10](#_Toc53398233)

[/\*Tabela topico\*/ 10](#_Toc53398234)

[/\*Tabela exercicio\*/ 11](#_Toc53398235)

[/\*Tabela alternativa\*/ 11](#_Toc53398236)

[Inserts 12](#_Toc53398237)

[/\*Insert curso\*/ 12](#_Toc53398238)

[/\*Insert curso\*/ 12](#_Toc53398239)

[/\*Insert aula\*/ 12](#_Toc53398240)

[/\*aluno\*/ 12](#_Toc53398241)

[/\*Insert aluno\_aula\*/ 13](#_Toc53398242)

[/\*Insert carreira\*/ 13](#_Toc53398243)

[/\*Inser curso\_carreira\*/ 13](#_Toc53398244)

[/\*Insert professor\*/ 13](#_Toc53398245)

[/\*Insert forum\*/ 13](#_Toc53398246)

[/\*Insert pagamento\*/ 13](#_Toc53398247)

[/\*Insert respostas\*/ 14](#_Toc53398248)

[/\*Insert professor\_curso\*/ 14](#_Toc53398249)

[/\*Insert parcelas\*/ 14](#_Toc53398250)

[/\*Insert tópico\*/ 15](#_Toc53398251)

[/\*Insert exercicio\*/ 15](#_Toc53398252)

[/\*Insert alternativa\*/ 15](#_Toc53398253)

[Selects 16](#_Toc53398254)

[/\*Quais cursos cada aluno está matriculado\*/ 16](#_Toc53398255)

[/\*Quais aulas cada aluno está matriculado\*/ 16](#_Toc53398256)

[/\*Quais aulas,curso e carreira cada aluno está matriculado\*/ 16](#_Toc53398257)

[/\*Exibe grade curricular completa\*/ 16](#_Toc53398258)

[/\*Busca aluno e o plano que cada um assinou\*/ 16](#_Toc53398259)

[/\*Todos os alunos que tem parcelas à pagar\*/ 16](#_Toc53398260)

[/\*Busca tudo do forum\*/ 17](#_Toc53398261)

[/\*Busca as perguntas não resolvidas do forum\*/ 17](#_Toc53398262)

## Mini mundo

Trata-se de uma escola on line que oferece vários cursos.

Cada curso é composto de video-aulas, textos e exercicios.

Para acessar os cursos o usuário precisa fazer uma assinatura anual.

Cada aula possui um fórum para tirar dúvidas com os professores e com outros alunos.

Os alunos podem escolher tres tipos de planos diferentes, basico, intermediário e avançado e eles darão acesso de 3 meses, 6 meses e 12 meses respectivamente.

Quando o aluno concluír todas as aulas de um determinado curso, ele recebe o certificado do curso.

Os cursos são organizados por carreira. Isso para dar direção ao aluno sobre quais assuntos ele deve dominar.

Cada carreira tem uma lista pré definida de cursos.

Cada curso é compoosto por diversas aulas e as aulas serão dividas em tópicos.

Cada tópico pode ou não conter exercicios de multipla escolha.

Tabelas e atributos a serem armazenados:

### Pagamento

Idpagamento

Numcartao

Nometitular

Sobrenometitular

Vencimentodia

Vencimentomes

Vencimentoano

Datacompra

Qtdparcelas

### Parcelas

Idparcela

Valor

Vencimento

Status

### Plano

Idplano

Nomeplano

dataInicio

dataTermino

### Aluno

Idaluno

Nomealuno

Emailaluno

Cpf

Idplano

Senha

Imagem

Pontuação

dataentrada

### Aula

Idaula

Titulo

Pontos

Status

### Tópico

idtopico

Video

Transcricao

Observacao

Pontos

Status

### Forum

Idaluno

Pergunta

DataPostagem

status

### Respostas

Idresposta

Textoresposta

Status

### Curso

Idcurso

Nomecurso

Datacadastro

Ultimaatualizacao

### Professor

Idprofessor

Nomeprofessor

Email

Senha

CPF

### Carreira

Idcarreira

Nomecarreira

### Exercicio

Idexercicio

Questão

AlternativaCorreta

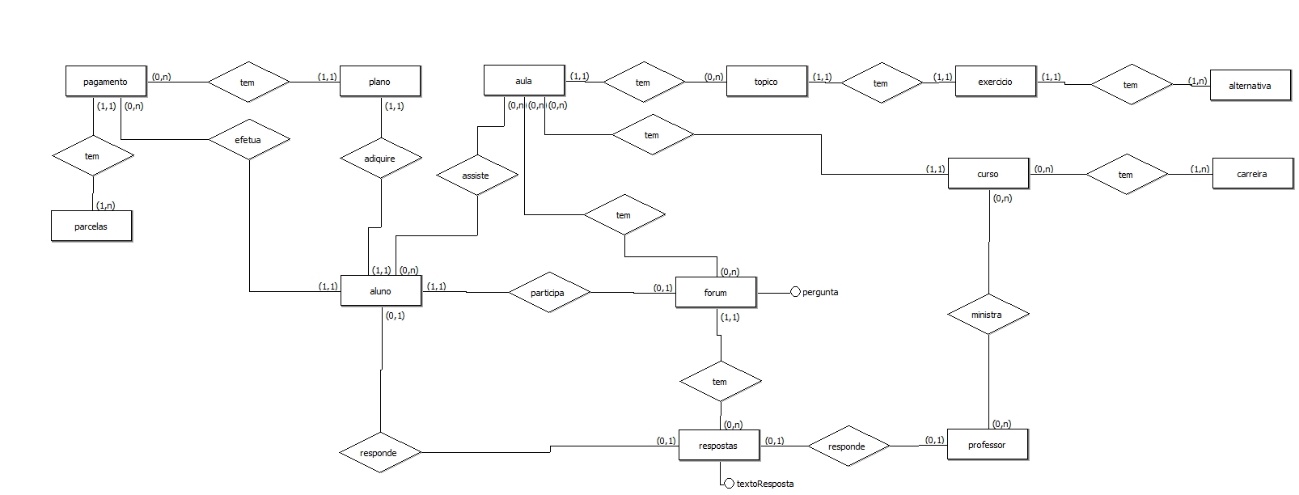
Pontos

### Alternativa

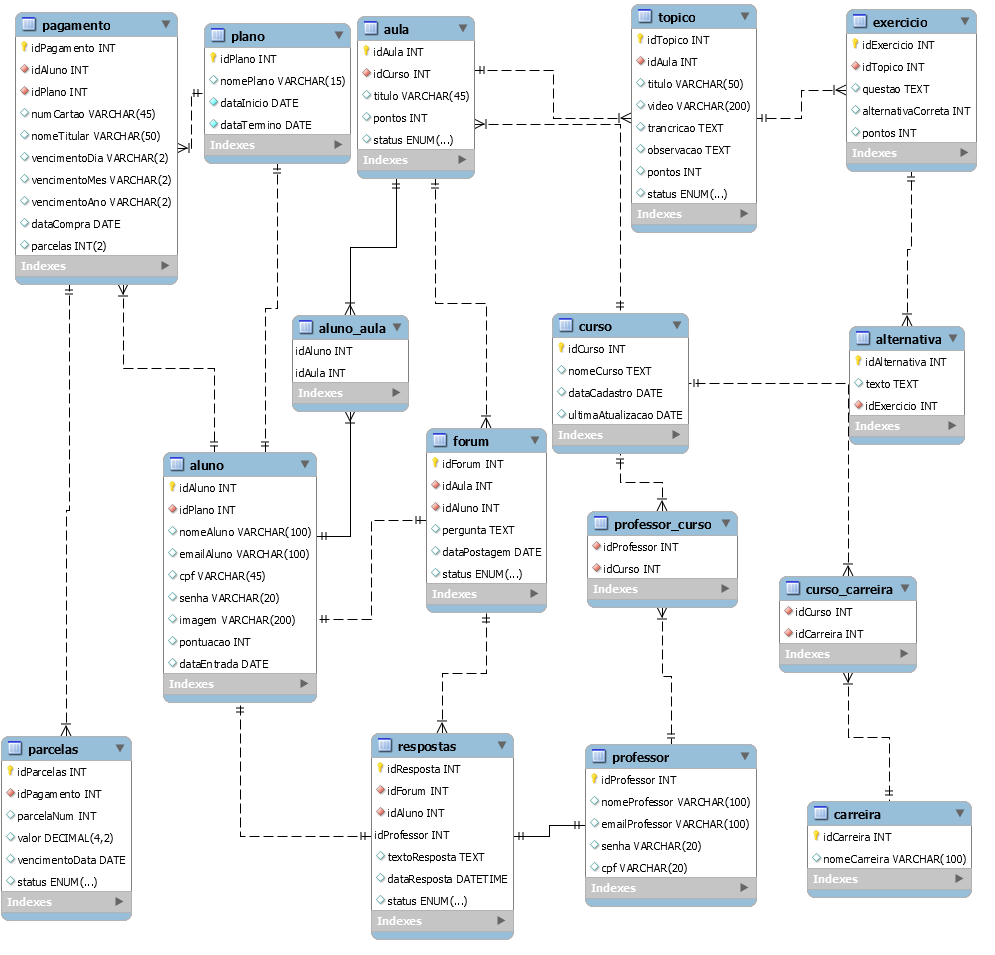
idAlternativa

Texto

## Modelo Conceitual



## Modelo Lógico



## Script

/\*\*/

### /\*Criando o schema\*/

CREATE SCHEMA escola\_online;

USE escola\_online ;

### /\*Tabela plano\*/

CREATE TABLE plano (

idPlano INT NOT NULL PRIMARY KEY AUTO\_INCREMENT COMMENT 'Chave Primaria',

nomePlano VARCHAR(15) COMMENT 'Nome do Plano',

dataInicio DATE COMMENT 'Data de início do plano',

dataTermino DATE COMMENT 'Data de termino do plano'

);

### /\*Tabela curso\*/

CREATE TABLE curso (

idCurso INT NOT NULL PRIMARY KEY AUTO\_INCREMENT COMMENT 'Chave primária da tabela',

nomeCurso TEXT COMMENT 'Nome do curso',

dataCadastro DATE COMMENT 'Data de cadastramento do curso no sistema',

ultimaAtualizacao DATE COMMENT 'Data da última atualização do curso'

);

### /\*Tabela aula\*/

CREATE TABLE aula (

idAula INT NOT NULL PRIMARY KEY AUTO\_INCREMENT COMMENT 'Chave primária da tabela',

idCurso INT COMMENT 'Chave extrangeira da tabela Curso',

titulo VARCHAR(45) COMMENT 'Título da aula',

pontos INT COMMENT 'Valor da aula em pontos',

status ENUM('INCONCLUIDO', 'CONCLUIDO') NULL,

FOREIGN KEY (idCurso) REFERENCES curso (idCurso) ON DELETE NO ACTION ON UPDATE NO ACTION

);

### /\*Tabela aluno\*/

CREATE TABLE aluno (

idAluno INT NOT NULL PRIMARY KEY AUTO\_INCREMENT COMMENT 'Chave primária da tabela',

idPlano INT COMMENT 'Chave extrangeira da tabela Plano',

nomeAluno VARCHAR(100) COMMENT 'Nome do aluno',

emailAluno VARCHAR(100) COMMENT 'Email do aluno',

cpf VARCHAR(45) COMMENT 'Cpf do aluno',

senha VARCHAR(200) COMMENT 'Senha do aluno',

imagem VARCHAR(200) COMMENT 'Caminho da imágem de perfi do aluno',

pontuacao INT COMMENT 'Pontos de participação do aluno',

dataEntrada DATE COMMENT 'Data de entrada do aluno',

FOREIGN KEY (idPlano) REFERENCES plano (idPlano) ON DELETE NO ACTION ON UPDATE NO ACTION

);

### /\*Tabela aluno\_aula\*/

CREATE TABLE aluno\_aula (

idAula INT NOT NULL COMMENT 'Chave extrangeira da tabela aula',

idAluno INT NOT NULL COMMENT 'Chave extrangeira da tabela aluno',

FOREIGN KEY (idAula) REFERENCES aula (idAula) ON DELETE NO ACTION ON UPDATE NO ACTION,

FOREIGN KEY (idAluno) REFERENCES aluno (idAluno) ON DELETE NO ACTION ON UPDATE NO ACTION

);

### /\*Tabela carreira\*/

CREATE TABLE carreira (

idCarreira INT NOT NULL PRIMARY KEY AUTO\_INCREMENT COMMENT 'Chave primaria da tabela',

nomeCarreira VARCHAR(100) NULL COMMENT 'Nome da carreira'

);

### /\*Tabela curso\_carreira\*/

CREATE TABLE curso\_carreira (

idCarreira INT NOT NULL COMMENT 'Chave extrangeira da tabela carreira',

idCurso INT NOT NULL COMMENT 'Chave extrangeira da tabela curso',

FOREIGN KEY (idCurso) REFERENCES curso (idCurso) ON DELETE NO ACTION ON UPDATE NO ACTION,

FOREIGN KEY (idCarreira) REFERENCES carreira (idCarreira) ON DELETE NO ACTION ON UPDATE NO ACTION

);

### /\*Tabela professor\*/

CREATE TABLE professor (

idProfessor INT NOT NULL PRIMARY KEY AUTO\_INCREMENT COMMENT 'Chave primária da tabela',

nomeProfessor VARCHAR(100) NULL COMMENT 'Nome do professor',

emailProfessor VARCHAR(100) NULL COMMENT 'Email do professor',

senha VARCHAR(200) NULL COMMENT 'Senha do professor',

cpf VARCHAR(20) NULL COMMENT 'Cpf do professor'

);

### /\*Tabela forum\*/

CREATE TABLE forum (

idForum INT NOT NULL PRIMARY KEY AUTO\_INCREMENT COMMENT 'Chave primária da tabela',

idAula INT NOT NULL COMMENT 'Chave extrangeira da tabela aula',

idAluno INT NOT NULL COMMENT 'Chave extrangeira da tabela aluno',

pergunta TEXT NULL COMMENT 'Texto da pergunta',

dataPostagem DATETIME NULL COMMENT 'Data da postágem',

status ENUM('ABERTO', 'RESOLVIDA') NULL COMMENT 'Status da pergunta',

FOREIGN KEY (idAula) REFERENCES aula (idAula) ON DELETE NO ACTION ON UPDATE NO ACTION,

FOREIGN KEY (idAluno) REFERENCES aluno (idAluno) ON DELETE NO ACTION ON UPDATE NO ACTION

);

### /\*Tabela pagamento\*/

CREATE TABLE pagamento (

idPagamento INT NOT NULL PRIMARY KEY AUTO\_INCREMENT COMMENT 'Chave Primária',

idAluno INT NOT NULL COMMENT 'Chave Extrangeira da Tabela Aluno',

idPlano INT NOT NULL COMMENT 'Chave Extrangeira da Tabela Plano',

numCartao VARCHAR(45) COMMENT 'Número do Cartão',

nomeTitular VARCHAR(50) COMMENT 'Nome completo do titular',

vencimentoDia VARCHAR(2) COMMENT 'Dia do vencimento do cartão',

vencimentoMes VARCHAR(2) COMMENT 'Mês do vencimento do cartão',

vencimentoAno VARCHAR(2) COMMENT 'Ano de vencimento do cartão',

dataCompra DATETIME COMMENT 'Data da compra',

parcelas INT(2) COMMENT 'Quantidade de parcelas',

FOREIGN KEY (idAluno) REFERENCES aluno (idAluno) ON DELETE NO ACTION ON UPDATE NO ACTION,

FOREIGN KEY (idPlano) REFERENCES plano (idPlano) ON DELETE NO ACTION ON UPDATE NO ACTION

);

### /\*Tabela respostas\*/

CREATE TABLE respostas (

idResposta INT NOT NULL PRIMARY KEY AUTO\_INCREMENT COMMENT 'Chave primária da tabela',

idForum INT NOT NULL COMMENT 'Chave extrangeira da tabela Forum',

idAluno INT NULL COMMENT 'Chave extrangeira da tabela aluno',

idProfessor INT NULL COMMENT 'Chave extrangeira da tabela professor',

textoResposta TEXT COMMENT 'Conteúdo da resposta',

dataResposta DATETIME COMMENT 'Data da resposta',

status ENUM('certa', 'errada') NULL COMMENT 'Resultado se é ou não a resposta correta da questão',

FOREIGN KEY (idForum) REFERENCES forum (idForum) ON DELETE NO ACTION ON UPDATE NO ACTION,

FOREIGN KEY (idAluno) REFERENCES aluno (idAluno) ON DELETE NO ACTION ON UPDATE NO ACTION,

FOREIGN KEY (idProfessor) REFERENCES professor (idProfessor) ON DELETE NO ACTION ON UPDATE NO ACTION

);

### /\*Tabela professor\_curso\*/

CREATE TABLE professor\_curso (

idProfessor INT NOT NULL,

idCurso INT NOT NULL,

FOREIGN KEY (idProfessor) REFERENCES professor (idProfessor) ON DELETE NO ACTION ON UPDATE NO ACTION,

FOREIGN KEY (idCurso) REFERENCES curso (idCurso) ON DELETE NO ACTION ON UPDATE NO ACTION

);

### /\*Tabela parcelas\*/

CREATE TABLE parcelas (

idParcelas INT NOT NULL PRIMARY KEY AUTO\_INCREMENT COMMENT 'Chave primária da tabela',

idPagamento INT NOT NULL,

parcelaNum INT NOT NULL,

valor DECIMAL(6,2) NULL COMMENT 'Valor da parcela',

vencimentoData DATE NULL COMMENT 'Data de vencimento da parcela',

status ENUM('A\_PAGAR', 'QUITADO', 'ATRASADO') NULL COMMENT 'Status do pagamento da parcela',

FOREIGN KEY (idPagamento) REFERENCES pagamento (idPagamento) ON DELETE NO ACTION ON UPDATE NO ACTION

);

### /\*Tabela topico\*/

CREATE TABLE topico (

idTopico INT NOT NULL PRIMARY KEY AUTO\_INCREMENT COMMENT 'Chave primária da tabela',

idAula INT NOT NULL,

titulo VARCHAR(50) COMMENT 'Título do tópico',

video VARCHAR(200) COMMENT 'Endereço do vídeo',

trancricao TEXT COMMENT 'Trancrição do vídeo',

observacao TEXT,

pontos INT COMMENT 'Valor do tópico por pontos',

status ENUM('INCONCLUIDO', 'CONCLUIDO') NULL,

FOREIGN KEY (idAula) REFERENCES aula (idAula) ON DELETE NO ACTION ON UPDATE NO ACTION

);

### /\*Tabela exercicio\*/

CREATE TABLE exercicio (

idExercicio INT NOT NULL PRIMARY KEY AUTO\_INCREMENT COMMENT 'Chave primária da tabela',

idTopico INT NOT NULL COMMENT 'Chave extrangeira da tabela topico',

questao TEXT NULL COMMENT 'Texto da questão',

alternativaCorreta INT NULL COMMENT 'O id da alternativa correta',

pontos INT NULL COMMENT 'Valor da questão em pontos',

FOREIGN KEY (idTopico) REFERENCES topico (idTopico) ON DELETE NO ACTION ON UPDATE NO ACTION

);

### /\*Tabela alternativa\*/

CREATE TABLE alternativa (

idAlternativa INT NOT NULL PRIMARY KEY AUTO\_INCREMENT COMMENT 'Chave primária da tabela',

texto TEXT NULL COMMENT 'Texto da alternativa',

idExercicio INT NOT NULL COMMENT 'Chave extrangeira da tabela exercicio',

FOREIGN KEY (idExercicio) REFERENCES exercicio (idExercicio) ON DELETE NO ACTION ON UPDATE NO ACTION

);

## Inserts

### /\*Insert curso\*/

INSERT INTO plano(nomePlano,dataInicio,dataTermino) VALUE('Basico','2020-01-01','2020-3-01');

INSERT INTO plano(nomePlano,dataInicio,dataTermino) VALUE('Intermediario','2020-01-01','2020-6-01');

INSERT INTO plano(nomePlano,dataInicio,dataTermino) VALUE('Avançado','2020-01-01','2020-12-01');

### /\*Insert curso\*/

INSERT INTO curso(nomeCurso,dataCadastro,ultimaAtualizacao) VALUES('Logica Básica','2020-09-01','2020-10-01');

INSERT INTO curso(nomeCurso,dataCadastro,ultimaAtualizacao) VALUES('Orientação a Objetos','2020-09-01','2020-10-01');

INSERT INTO curso(nomeCurso,dataCadastro,ultimaAtualizacao) VALUES('Tipo de Listas','2020-09-01','2020-10-01');

INSERT INTO curso(nomeCurso,dataCadastro,ultimaAtualizacao) VALUES('Trabalhando com Exceções','2020-09-01','2020-10-01');

INSERT INTO curso(nomeCurso,dataCadastro,ultimaAtualizacao) VALUES('Criando Thread','2020-09-01','2020-10-01');

### /\*Insert aula\*/

INSERT INTO aula(idCurso,titulo,pontos) VALUES ( 1, "Introdução", 2);

INSERT INTO aula(idCurso,titulo,pontos) VALUES ( 1, "Historia", 2);

INSERT INTO aula(idCurso,titulo,pontos) VALUES ( 1, "Evolução", 2);

INSERT INTO aula(idCurso,titulo,pontos) VALUES ( 1, "Mão na massa", 2);

INSERT INTO aula(idCurso,titulo,pontos) VALUES ( 5, "Herança e Polimorfismo", 2);

INSERT INTO aula(idCurso,titulo,pontos) VALUES ( 2, "Interface", 2);

INSERT INTO aula(idCurso,titulo,pontos) VALUES ( 2, "Sobrescrevendo Métodos", 2);

### /\*aluno\*/

INSERT INTO aluno(idPlano,nomeAluno,emailAluno,CPF,senha,imagem,pontuacao,dataEntrada)

VALUES( 1, 'Fafa de Belem', 'fafa@gmail.com', '11231223312', sha('14523'),"#","20034","2019-01-01");

INSERT INTO aluno(idPlano,nomeAluno,emailAluno,CPF,senha,imagem,pontuacao,dataEntrada)

VALUES( 1, 'Reginaldo Rossi', 'reg@gmail.com', '13123123212', sha('34125'),"#","3462","2019-01-01");

INSERT INTO aluno(idPlano,nomeAluno,emailAluno,CPF,senha,imagem,pontuacao,dataEntrada)

VALUES( 2, 'Mc Marcinho', 'marc@gmail.com', '12312123123', sha('14523'),"#","211","2019-01-01");

INSERT INTO aluno(idPlano,nomeAluno,emailAluno,CPF,senha,imagem,pontuacao,dataEntrada)

VALUES( 2, 'Zeca Pagodinho', 'ze@gmail.com', '12231231231', sha('31245'),"#","0","2019-01-01");

INSERT INTO aluno(idPlano,nomeAluno,emailAluno,CPF,senha,imagem,pontuacao,dataEntrada)

VALUES( 3, 'Axl Rose', 'rose@gmail.com', '12312312231', sha('12345'),"#","8765443","2019-01-01");

INSERT INTO aluno(idPlano,nomeAluno,emailAluno,CPF,senha,imagem,pontuacao,dataEntrada)

VALUES( 3, 'Pepê e Nenêm', 'pene@gmail.com', '33312333331', sha('BCD45'),"#","2356","2019-01-01");

### /\*Insert aluno\_aula\*/

INSERT INTO aluno\_aula(idAula, idAluno) VALUES(1,1);

INSERT INTO aluno\_aula(idAula, idAluno) VALUES(4,2);

INSERT INTO aluno\_aula(idAula, idAluno) VALUES(1,3);

INSERT INTO aluno\_aula(idAula, idAluno) VALUES(5,4);

INSERT INTO aluno\_aula(idAula, idAluno) VALUES(2,4);

INSERT INTO aluno\_aula(idAula, idAluno) VALUES(1,5);

### /\*Insert carreira\*/

INSERT INTO carreira(nomeCarreira) VALUE("Desenvolvedor Java I");

INSERT INTO carreira(nomeCarreira) VALUE("Desenvolvedor Java II");

### /\*Inser curso\_carreira\*/

INSERT INTO curso\_carreira(idCarreira,idCurso) VALUES(1,1);

INSERT INTO curso\_carreira(idCarreira,idCurso) VALUES(1,2);

INSERT INTO curso\_carreira(idCarreira,idCurso) VALUES(2,3);

INSERT INTO curso\_carreira(idCarreira,idCurso) VALUES(2,4);

INSERT INTO curso\_carreira(idCarreira,idCurso) VALUES(2,5);

### /\*Insert professor\*/

INSERT INTO professor(nomeProfessor,emailProfessor,senha,cpf) VALUES ('Zé Ramalho', 'ze@gmail.com', sha('12345'),'12312312312');

INSERT INTO professor(nomeProfessor,emailProfessor,senha,cpf) VALUES ('Jô Soares', 'jo@gmail.com', sha('55555'),'88888888888');

### /\*Insert forum\*/

INSERT INTO forum(idAula,idAluno,pergunta,dataPostagem,status)

VALUES (1,1,"Como instalar o JDK?","2020-02-01 12:31:33", 'ABERTO');

INSERT INTO forum(idAula,idAluno,pergunta,dataPostagem,status)

VALUES (1,2,"Não conosigo fazer o Helo World?","2020-07-04 13:40:02", "ABERTO");

INSERT INTO forum(idAula,idAluno,pergunta,dataPostagem,status)

VALUES (1,3,"Como pegar os dados da lista?","2020-01-07 10:20:09", "RESOLVIDA");

INSERT INTO forum(idAula,idAluno,pergunta,dataPostagem,status)

VALUES (1,4,"O que é Map?","2020-03-05 10:20:00", "RESOLVIDA");

INSERT INTO forum(idAula,idAluno,pergunta,dataPostagem,status)

VALUES (1,5,"Como iterar na lista?","2020-02-02 11:02:33", "RESOLVIDA");

### /\*Insert pagamento\*/

INSERT INTO pagamento(idAluno,idPlano,numCartao, nomeTitular, vencimentoDia, vencimentoMes, vencimentoAno, dataCompra, parcelas)

VALUES(1,1,1234567890, "Muçum", "2", "5", "20", '2020-12-01 10:34:20', 12);

INSERT INTO pagamento(idAluno,idPlano,numCartao, nomeTitular, vencimentoDia, vencimentoMes, vencimentoAno, dataCompra, parcelas)

VALUES(2,3,6781234590, "Milton Nascimento", "6", "5", "20", '2020-11-01 20:12:11', 12);

INSERT INTO pagamento(idAluno,idPlano,numCartao, nomeTitular, vencimentoDia, vencimentoMes, vencimentoAno, dataCompra, parcelas)

VALUES(3,3,6712345890, "Ozzy Osbourne", "3", "12", "20", '2020-05-21 05:10:55', 12);

INSERT INTO pagamento(idAluno,idPlano,numCartao, nomeTitular, vencimentoDia, vencimentoMes, vencimentoAno, dataCompra, parcelas)

VALUES(4,2,4123567890, "Falcão", "7", "9", "20", '2020-11-23 09:12:43', 12);

INSERT INTO pagamento(idAluno,idPlano,numCartao, nomeTitular, vencimentoDia, vencimentoMes, vencimentoAno, dataCompra, parcelas)

VALUES(5,1,9123456780, "John Francis Bongiovi", "5", "4", "20", '2020-07-03 10:20:21', 12);

### /\*Insert respostas\*/

INSERT INTO respostas(idForum, idAluno, idProfessor, textoResposta,dataResposta, status)

VALUES(1, NULL, 1, "Procure no site oficial","2020-02-01 13:40:24", "certa");

INSERT INTO respostas(idForum, idAluno, idProfessor, textoResposta,dataResposta, status)

VALUES(2, NULL, 2, "Reveja a aula","2020-05-10 11:20:10", "certa");

INSERT INTO respostas(idForum, idAluno, idProfessor, textoResposta,dataResposta, status)

VALUES(2, NULL, 2, "Reveja a aula","2020-05-10 11:20:10", "certa");

INSERT INTO respostas(idForum, idAluno, idProfessor, textoResposta,dataResposta, status)

VALUES(2, 3, NULL, "Tenta fazer um getText()","2020-06-23 11:12:10", "errada");

INSERT INTO respostas(idForum, idAluno, idProfessor, textoResposta,dataResposta, status)

VALUES(3, 5, NULL, "Deve ser o Google Map","2020-06-23 11:40:30", "errada");

INSERT INTO respostas(idForum, idAluno, idProfessor, textoResposta,dataResposta, status)

VALUES(3, 2, NULL, "Use um forEach ou lambda","2020-06-07 09:23:55", "certa");

### /\*Insert professor\_curso\*/

INSERT INTO professor\_curso(idProfessor,idCurso) VALUES(1,1);

INSERT INTO professor\_curso(idProfessor,idCurso) VALUES(2,2);

INSERT INTO professor\_curso(idProfessor,idCurso) VALUES(1,3);

### /\*Insert parcelas\*/

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(1,1,100.0,"2020-12-01","QUITADO");

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(1,2,100.0,"2021-01-01","A\_PAGAR");

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(1,3,100.0,"2021-02-01","A\_PAGAR");

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(1,4,100.0,"2021-03-01","A\_PAGAR");

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(1,5,100.0,"2021-04-01","A\_PAGAR");

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(1,6,100.0,"2021-05-01","A\_PAGAR");

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(1,7,100.0,"2021-06-01","A\_PAGAR");

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(1,8,100.0,"2021-07-01","A\_PAGAR");

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(1,9,100.0,"2021-08-01","A\_PAGAR");

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(1,10,100.0,"2021-09-01","A\_PAGAR");

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(1,11,100.0,"2021-10-01","A\_PAGAR");

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(1,12,100.0,"2021-11-01","A\_PAGAR");

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(2,1,1000.0,"2021-11-01","QUITADO");

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(3,1,1000.0,"2021-11-01","QUITADO");

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(4,1,1000.0,"2021-11-01","QUITADO");

INSERT INTO parcelas(idPagamento,parcelaNum,valor,vencimentoData,status) VALUES(5,1,1000.0,"2021-11-01","QUITADO");

### /\*Insert tópico\*/

INSERT INTO topico(idAula,titulo,video,trancricao,observacao,pontos,status)

VALUES(1,"Apresentação","#","Lorem Impsu","Lorem Impsu","10","INCONCLUIDO");

INSERT INTO topico(idAula,titulo,video,trancricao,observacao,pontos,status)

VALUES(1,"História","#","Lorem Impsu","Lorem Impsu","10","INCONCLUIDO");

INSERT INTO topico(idAula,titulo,video,trancricao,observacao,pontos,status)

VALUES(1,"Instalação do ambiente","#","Lorem Impsu","Lorem Impsu","10","INCONCLUIDO");

### /\*Insert exercicio\*/

INSERT INTO exercicio(idTopico,questao,alternativaCorreta,pontos) VALUES(1,"O que é java?",4,10);

### /\*Insert alternativa\*/

INSERT INTO alternativa(texto,idExercicio) VALUES("É um animal.", 1);

INSERT INTO alternativa(texto,idExercicio) VALUES("É uma marca de café.", 1);

INSERT INTO alternativa(texto,idExercicio) VALUES("É a musica do Mr. Love Man.", 1);

INSERT INTO alternativa(texto,idExercicio) VALUES("É uma linguágem de programação.", 1);

## Selects

### /\*Quais cursos cada aluno está matriculado\*/

SELECT nomeAluno,titulo,nomeCurso

FROM aluno

LEFT JOIN aluno\_aula ON aluno.idAluno = aluno\_aula.idAluno

LEFT JOIN aula ON aula.idAula = aluno\_aula.idAula

INNER JOIN curso ON aula.idCurso = curso.idCurso;

### /\*Quais aulas cada aluno está matriculado\*/

SELECT idAluno,idPlano,nomeAluno,emailAluno,cpf,

imagem,pONtuacao,dataEntrada,idAula,idCurso,titulo,status

FROM aluno

LEFT JOIN aluno\_aula ON aluno.idAluno = aluno\_aula.idAluno

LEFT JOIN aula ON aula.idAula = aluno\_aula.idAula;

### /\*Quais aulas,curso e carreira cada aluno está matriculado\*/

SELECT aluno.idAluno,nomeAluno,aula.idAula,titulo,curso.idCurso,nomeCurso,carreira.idCarreira,nomeCarreira

FROM aluno

LEFT JOIN aluno\_aula ON aluno.idAluno = aluno\_aula.idAluno

LEFT JOIN aula ON aula.idAula = aluno\_aula.idAula

INNER JOIN curso ON aula.idCurso = curso.idCurso

INNER JOIN curso\_carreira ON curso.idCurso = curso\_carreira.idCurso

INNER JOIN carreira ON carreira.idCarreira = curso\_carreira.idCarreira;

### /\*Exibe grade curricular completa\*/

SELECT carreira.idCarreira,nomeCarreira,curso.idCurso,nomeCurso,dataCadastro,ultimaAtualizacao,titulo FROM carreira

INNER JOIN curso\_carreira ON carreira.idCarreira = curso\_carreira.idCarreira

INNER JOIN curso ON curso\_carreira.idCurso = curso.idCurso

INNER JOIN aula ON aula.idCurso = curso.idCurso;

SELECT \* FROM aluno\_aula;

### /\*Busca aluno e o plano que cada um assinou\*/

SELECT idAluno,nomeAluno,emailAluno,cpf,imagem,pONtuacao,dataEntrada,nomeplano,dataInicio,dataTermino FROM aluno

INNER JOIN plano ON aluno.idPlano = plano.idPlano;

### /\*Todos os alunos que tem parcelas à pagar\*/

SELECT \* FROM aluno

RIGHT JOIN pagamento ON aluno.idAluno = pagamento.idAluno

LEFT JOIN parcelas ON pagamento.idPagamento = parcelas.idPagamento

WHERE status = 'A\_PAGAR';

### /\*Busca tudo do forum\*/

SELECT forum.idForum,idAula,forum.idAluno AS Quem\_perguntou\_id,aluno.nomeAluno AS Quem\_perguntou\_nome,

pergunta,dataPostagem,forum.status,respostas.idAluno AS Quem\_respondeu\_aluno,

idProfessor AS Quem\_respondeu\_professor,textoResposta,dataResposta,respostas.status FROM forum

INNER JOIN respostas ON forum.idForum = respostas.idForum

INNER JOIN aluno ON aluno.idAluno = respostas.idAluno;

### /\*Busca as perguntas não resolvidas do forum\*/

SELECT forum.idForum,idAula,forum.idAluno as Quem\_perguntou\_id,aluno.nomeAluno as Quem\_perguntou\_nome,pergunta,dataPostagem,forum.status,respostas.idAluno as Quem\_respondeu\_aluno,idProfessor as Quem\_respondeu\_professor,textoResposta,dataResposta,respostas.status

FROM forum

INNER JOIN respostas ON forum.idForum = respostas.idForum

INNER JOIN aluno ON aluno.idAluno = respostas.idAluno

WHERE forum.status = 'ABERTO';