Cadeia de conexões para o python;

host=unictec.postgres.database.azure.com port=5432

dbname={your\_database}

user=bolinho

password={your\_password}

sslmode=require

SET DATESTYLE TO POSTGRES, DMY ;

--Definindo a prioridade de ordem

**--TABELA CURSO (EXECUTÁVEL)**

DROP TABLE IF EXISTS curso CASCADE;

CREATE TABLE curso(

cod\_curso VARCHAR(6) NOT NULL PRIMARY KEY,

nome\_curso VARCHAR(50) NOT NULL,

qntd\_semestres CHAR(2) NOT NULL,

tipo VARCHAR(30) CHECK (tipo IN ('Bacharelado', 'Tecnologo')),

turno\_curso VARCHAR(30) CHECK (turno\_curso IN ('Matutino','Vespertino', 'Noturno')),

campus VARCHAR(30) NOT NULL,

data\_criacao timestamp NOT NULL DEFAULT CURRENT\_TIMESTAMP);

**-- TABELA ENDERECO (EXECUTÁVEL)**

DROP TABLE IF EXISTS endereco CASCADE;

CREATE TABLE endereco (

cep CHAR(7) NOT NULL PRIMARY KEY,

n\_residencia INT NOT NULL,

logradouro VARCHAR (40) NOT NULL)

**-- TABELA ALUNO (EXECUTÁVEL)**

DROP TABLE IF EXISTS aluno CASCADE;

CREATE TABLE aluno (

ra INT PRIMARY KEY,

cpf VARCHAR(11) NOT NULL,

data\_nasc DATE NOT NULL,

situacao VARCHAR(11) CHECK (situacao IN ('CURSANDO','TRANCADO','CONCLUIDO')),

nome\_aluno VARCHAR(100) NOT NULL,

email VARCHAR(32) NOT NULL,

data\_matricula DATE NOT NULL,

cod\_curso CHAR(6) NOT NULL,

cep CHAR(7) NOT NULL,

FOREIGN KEY (cod\_curso) REFERENCES curso(cod\_curso),

FOREIGN KEY (cep) REFERENCES endereco(cep));

**-- TABELA PROFESSOR**

DROP TABLE IF EXISTS professor CASCADE;

CREATE TABLE professor (

registro\_prof INT PRIMARY KEY,

cpf VARCHAR(11) NOT NULL,

email VARCHAR(32) NOT NULL,

cod\_curso CHAR(6) NOT NULL,

cep CHAR(7) NOT NULL,

FOREIGN KEY (cod\_curso) REFERENCES curso(cod\_curso),

FOREIGN KEY (cep) REFERENCES endereco(cep));

**-- DISCIPLINA (EXECUTÁVEL)**

DROP TABLE IF EXISTS disciplina CASCADE;

CREATE TABLE disciplina (

cod\_disciplina VARCHAR(25) NOT NULL PRIMARY KEY,

nome\_disciplina VARCHAR(100) NOT NULL,

carga\_horaria INT NOT NULL,

periodo CHAR(2) NOT NULL,

pre\_requisito VARCHAR(15),

cod\_curso VARCHAR(6) NOT NULL,

FOREIGN KEY (cod\_curso) REFERENCES curso(cod\_curso));

**-- MATRÍCULA**

DROP TABLE IF EXISTS matricula CASCADE;

CREATE TABLE matrícula (

ra INT NOT NULL,

cod\_disciplina VARCHAR(25) NOT NULL,

semestre CHAR(2) NOT NULL,

status VARCHAR NOT NULL,

media DOUBLE NOT NULL,

FOREIGN KEY (ra) REFERENCES aluno(ra),

FOREIGN KEY (cod\_disciplina) REFERENCES disciplina(cod\_disciplina));

**-- PROVA**

DROP TABLE IF EXISTS prova CASCADE;

CREATE TABLE prova (

num\_prova SERIAL NOT NULL PRIMARY KEY,

ra INT NOT NULL,

cod\_disciplina VARCHAR(25) NOT NULL,

semestre CHAR(2) NOT NULL,

nota\_tirada DOUBLE NOT NULL,

peso DOUBLE NOT NULL,

FOREIGN KEY (ra) REFERENCES aluno(ra),

FOREIGN KEY (cod\_disciplina) REFERENCES disciplina(cod\_disciplina),

FOREIGN KEY (semestre) REFERENCES matricula(semestre));

**-- AULA**

DROP TABLE IF EXISTS aula CASCADE;

CREATE TABLE aula(

id\_aula INT NOT NULL,

cod\_disciplina VARCHAR(25) NOT NULL,

data TIMESTAMP NOT NULL,

conteudo VARCHAR NOT NULL,

FOREIGN KEY (cod\_disciplina) REFERENCES disciplina(cod\_disciplina));

**-- FREQUENCIA**

DROP TABLE IF EXISTS frequencia CASCADE;

CREATE TABLE frequencia (

id\_aula INT NOT NULL PRIMARY KEY,

cod\_disciplina VARCHAR(25) NOT NULL,

ra INT NOT NULL,

Presente BOOLEAN NOT NULL,

FOREIGN KEY (cod\_disciplina) REFERENCES disciplina(cod\_disciplina),

FOREIGN KEY (ra) REFERENCES aluno(ra));

-- turma (removida por questões internas)

DROP TABLE IF EXISTS turma CASCADE;

CREATE TABLE turma (

cod\_disciplina VARCHAR(25) NOT NULL PRIMARY KEY,

período DATE NOT NULL PRIMARY KEY,

registro professor INT NOT NULL,

FOREIGN KEY (resgistro\_prof) REFERENCES professor (registro\_prof))

FOREIGN KEY (cod\_disciplina) REFERENCES disciplina(cod\_disciplina))