

# ABEX III

Bruno Cardoso da Silva

João Luiz de Oliveira Todescato

João Arthur Gonsalez Soares

Repositório GitHub: <https://github.com/Ney-Pedreiro/Abex-III->

Models.py:

```
from django.db import models

# Create your models here.
class Jogador(models.Model):
    idjogador = models.IntegerField(db_column='idJogador',
primary_key=True)
    nome = models.CharField(max_length=25)
    login = models.CharField(max_length=25)
    senha = models.CharField(max_length=25)

    class Meta:
        managed = False
        db_table = 'jogador'
```

Views.py

```
from django.shortcuts import render, redirect
from .models import Jogador
from .forms import JogadorForm
from django.http import HttpResponse
from django.template import loader
```

```
def index(request):
    template = loader.get_template('index.html')
    context = {}
    return HttpResponse(template.render(context, request))

def jogadorList(request):
    jogadores = Jogador.objects.all()
    return render(request, "jogador-
list.html", {'jogadores': jogadores})

def jogadorCreate(request):
    if request.method == "POST":
        form = JogadorForm(request.POST)
        if form.is_valid():
            try:
                form.save()
                model = form.instance
                return redirect('jogador-list')
            except:
                pass
        else:
            form = JogadorForm()
            return render(request, 'jogador-
create.html', {'form': form})

def jogadorDelete(request, idjogador):
    jogador = Jogador.objects.get(idjogador = idjogador)
    try:
```

```

        jogador.delete()
    except:
        pass
    return redirect('jogador-list')

def jogadorUpdate(request, idjogador):
    jogador = Jogador.objects.get(idjogador=idjogador)
    form = JogadorForm(initial={
        'idjogador': jogador.idjogador,
        'nome': jogador.nome,
        'login': jogador.login,
        'senha': jogador.senha
    })
    if request.method == "POST":
        form = JogadorForm(request.POST, instance=jogador)
        if form.is_valid():
            try:
                form.save()
                return redirect('jogador-list')
            except Exception as e:
                return render(request, 'jogador-
update.html', {'form': form, 'error': str(e)})
        return render(request, 'jogador-update.html', {'form':
form})

```

#### Script Banco de Dados:

```

CREATE TABLE Banco( idBanco INTEGER NOT NULL, valor_emprestimo FLOAT,
valor_investimento FLOAT, saldo FLOAT, Personagem_idPersonagem INTEGER NOT NULL,

```

PRIMARY KEY(idBanco), CONSTRAINT Banco\_Personagem\_idPersonagem\_fkey FOREIGN KEY (Personagem\_idPersonagem) REFERENCES Personagem (idPersonagem) );

CREATE TABLE Compra( idCompra INTEGER NOT NULL, Personagem\_idPersonagem INTEGER NOT NULL, Quantidade\_comprada INTEGER NOT NULL, PRIMARY KEY(idCompra), CONSTRAINT Compra\_Personagem\_idPersonagem\_fkey FOREIGN KEY (Personagem\_idPersonagem) REFERENCES Personagem (idPersonagem) );

CREATE TABLE Estado\_Personagem( idEstadoPersonagem INTEGER NOT NULL, status\_fome INTEGER, status\_sede INTEGER, status\_felicidade INTEGER, Personagem\_idPersonagem INTEGER NOT NULL, PRIMARY KEY(idEstadoPersonagem), CONSTRAINT Estado\_Personagem\_Personagem\_idPersonagem\_fkey FOREIGN KEY (Personagem\_idPersonagem) REFERENCES Personagem (idPersonagem) );

CREATE TABLE Jogador( idJogador INTEGER NOT NULL, nome VARCHAR NOT NULL, login VARCHAR NOT NULL, senha VARCHAR NOT NULL, PRIMARY KEY(idJogador) );

CREATE TABLE Loja( idLoja INTEGER NOT NULL, nome\_produto VARCHAR NOT NULL, preco FLOAT NOT NULL, quantidade INTEGER, Compra\_idCompra INTEGER NOT NULL, PRIMARY KEY(idLoja), CONSTRAINT Loja\_Compra\_idCompra\_fkey FOREIGN KEY (Compra\_idCompra) REFERENCES Compra (idCompra) );

CREATE TABLE MiniJogo( idMiniJogo INTEGER NOT NULL, minijogo VARCHAR NOT NULL, Personagem\_idPersonagem INTEGER NOT NULL, PRIMARY KEY(idMiniJogo), CONSTRAINT MiniJogo\_Personagem\_idPersonagem\_fkey FOREIGN KEY (Personagem\_idPersonagem) REFERENCES Personagem (idPersonagem) );

CREATE TABLE Personagem( idPersonagem INTEGER NOT NULL, nome VARCHAR NOT NULL, genero CHARACTER, necessidades INTEGER NOT NULL, Jogador\_idJogador INTEGER NOT NULL, PRIMARY KEY(idPersonagem), CONSTRAINT Personagem\_Jogador\_idJogador\_fkey FOREIGN KEY (Jogador\_idJogador) REFERENCES Jogador (idJogador) );

CREATE TABLE Profissao( idProfissao INTEGER NOT NULL, profissao VARCHAR, salario FLOAT, MiniJogo\_idMiniJogo INTEGER NOT NULL, PRIMARY KEY(idProfissao), CONSTRAINT Profissao\_MiniJogo\_idMiniJogo\_fkey FOREIGN KEY (MiniJogo\_idMiniJogo) REFERENCES MiniJogo (idMiniJogo) );