**Conexão com o banco e listagem de dados – PyMongo**

**pip install Flask-PyMongo**

**database.py:**

*from* flask\_pymongo *import* PyMongo

*from* bson *import* ObjectId

mongo = PyMongo()

# *Classe responsável por criar a entidade "Games" com seus atributos.*

class Game():

    def \_\_init\_\_(self, titulo, ano, categoria, plataforma, preco, quantidade):

        self.titulo = titulo

        self.ano = ano

        self.categoria = categoria

        self.plataforma = plataforma

        self.preco = preco

        self.quantidade = quantidade

    def *save*(self):

        mongo.db.games.*insert\_one*({

            'titulo': self.titulo,

            'ano': self.ano,

            'categoria': self.categoria,

            'plataforma': self.plataforma,

            'preco': self.preco,

            'quantidade': self.quantidade

        })

**app.py:**

*from* flask *import* Flask, *render\_template*

*from* controllers *import* routes

*from* models.database *import* mongo, Game

app = Flask(\_\_name\_\_, template\_folder='views')

routes.*init\_app*(app)

app.config['MONGO\_URI'] = 'mongodb://localhost:27017/games'

*if* \_\_name\_\_ *==* '\_\_main\_\_':

    mongo.*init\_app*(app=app)

    # *Adicione um documento à coleção ao iniciar o aplicativo*

*with* app.*app\_context*():

        # *Supondo que 'games' seja o nome da sua coleção*

*if* 'games' not in mongo.db.*list\_collection\_names*():

            # *Insira um documento para criar a coleção*

            game = Game(

                titulo='',

                ano=0,

                categoria='',

                plataforma='',

                preco=0,

                quantidade=0

            )

            game.*save*()

    app.*run*(host='localhost', port=5000, debug=True)

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

**database.py:**

    @staticmethod

    def *get\_all*():

*return* list(mongo.db.games.*find*())

**routes.py:**

# *CRUD - LISTAGEM, CADASTRO E EXCLUSÃO*

    @app.*route*('/estoque', methods=['GET', 'POST'])

gamesestoque = Game.*get\_all*()

*return* *render\_template*('estoque.html', gamesestoque=gamesestoque)

**estoque.html**

{% extends 'base.html' %}

{% block content %}

    <h2>Controle de estoque</h2><br>

    <hr>

    <h5>Jogos em estoque:</h5><br>

    <table *class*="table table-bordered table-hover">

        <thead *class*="table-dark">

            <tr>

                <th>Título:</th>

                <th>Ano:</th>

                <th>Categoria:</th>

                <th>Plataforma:</th>

                <th>Preço:</th>

                <th>Quantidade:</th>

            </tr>

        </thead>

        <tbody>

        {% for g in gamesestoque %}

            <tr>

                <td>{{g.titulo}}</td>

                <td>{{g.ano}}</td>

                <td>{{g.categoria}}</td>

                <td>{{g.plataforma}}</td>

                <td>{{g.preco}}</td>

                <td>{{g.quantidade}}</td>

            </tr>

        {% endfor %}

        </tbody>

    </table>

{% endblock content %}

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

**CADASTRO DE DADOS NO BANCO**

**estoque.html:**

{% extends 'base.html' %}

{% block content %}

    <h2>Controle de estoque</h2><br>

    <details>

        <summary>[ Cadastrar novo jogo ]</summary>

            <div *class*="col-md-4">

                <form *action*="{{url\_for('estoque')}}" *method*="POST">

                    <input *type*="text" *name*="titulo" *placeholder*="Título" *class*="form-control" *required*>

                    <input *type*="number" *name*="ano" *placeholder*="Ano" *min*="1970" *max*="2099" *class*="form-control" *required*>

                    <input *type*="text" *name*="categoria" *placeholder*="Categoria" *class*="form-control" *required*>

                    <input *type*="text" *name*="plataforma" *placeholder*="Plataforma" *class*="form-control" *required*>

                    <input *type*="number" *name*="preco" *placeholder*="Preço" *min*=0 *step*="0.01" *class*="form-control" *required*>

                    <input *type*="number" *name*="quantidade" *placeholder*="Quantidade" *class*="form-control" *required*><br>

                    <input *type*="submit" *value*="Cadastrar" *class*="btn btn-success">

                    <input *type*="reset" *value*="Limpar" *class*="btn btn-danger">

                </form>

            </div>

    </details>

    <hr>

    <h5>Jogos em estoque:</h5><br>

**routes.py:**

    # *CRUD - LISTAGEM, CADASTRO E EXCLUSÃO*

    @app.*route*('/estoque', methods=['GET', 'POST'])

        # *Cadastra um novo jogo*

*if* request.method *==* 'POST':

            newgame = Game(request.form['titulo'], request.form['ano'], request.form['categoria'], request.form['plataforma'], request.form['preco'], request.form['quantidade'])

            Game.*save*(newgame)

*return* *redirect*(*url\_for*('estoque'))

*else*:

            gamesestoque = Game.*get\_all*()

*return* *render\_template*('estoque.html', gamesestoque=gamesestoque)

Interface gráfica do usuário, Tabela

Descrição gerada automaticamente

Tabela

Descrição gerada automaticamente com confiança média

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

**DELETANDO DADOS NO BANCO**

**database.py**

    @staticmethod

    def *delete*(id):

        mongo.db.games.*delete\_one*({'\_id': ObjectId(id)})

**estoque.html**

    <h5>Jogos em estoque:</h5><br>

    <table *class*="table table-bordered table-hover">

        <thead *class*="table-dark">

            <tr>

                <th>Título:</th>

                <th>Ano:</th>

                <th>Categoria:</th>

                <th>Plataforma:</th>

                <th>Preço:</th>

                <th>Quantidade:</th>

                <th *style*="text-align: center;" *colspan*="2">Ação:</th>

            </tr>

        </thead>

        <tbody>

        {% for g in gamesestoque %}

            <tr>

                <td>{{g.titulo}}</td>

                <td>{{g.ano}}</td>

                <td>{{g.categoria}}</td>

                <td>{{g.plataforma}}</td>

                <td>{{g.preco}}</td>

                <td>{{g.quantidade}}</td>

                <td><a *style*="text-decoration: none; color: orange;" *href*="#">Editar</a></td>

                <td><a *style*="text-decoration: none; color: red;" *href*="{{url\_for('estoque', id=g.\_id)}}" *onclick*="*return* confirm('Deseja mesmo excluir o jogo {{g.titulo}}?')">Excluir</a></td>

            </tr>

        {% endfor %}

        </tbody>

    </table>

{% endblock content %}

**routes.py:**

    # *CRUD - LISTAGEM, CADASTRO E EXCLUSÃO*

    @app.*route*('/estoque', methods=['GET', 'POST'])

    @app.*route*('/estoque/delete/<id>')

    def *estoque*(id=None):

*if* id:

            # *Deleta o cadastro pela ID*

            Game.*delete*(id)

*return* *redirect*(*url\_for*('estoque'))

        # *Cadastra um novo jogo*

*if* request.method *==* 'POST':

            newgame = Game(request.form['titulo'], request.form['ano'], request.form['categoria'], request.form['plataforma'], request.form['preco'], request.form['quantidade'])

            Game.*save*(newgame)

*return* *redirect*(*url\_for*('estoque'))

*else*:

            # *Armazena em "gamesestoque" todos os valores, como em um SELECT e encaminha para estoque.html*

            gamesestoque = Game.*get\_all*()

*return* *render\_template*('estoque.html', gamesestoque=gamesestoque)

Tela de celular com publicação numa rede social

Descrição gerada automaticamente

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

**ALTERANDO DADOS NO BANCO**

**database.py**

    @staticmethod

    def *get\_by\_id*(id):

*return* mongo.db.games.*find\_one*({'\_id': ObjectId(id)})

Texto

Descrição gerada automaticamente

**routes.py:**

    # *CRUD - EDIÇÃO*

    @app.*route*('/edit/<id>', methods=['GET', 'POST'])

    def *edit*(id):

        g =  Game.*get\_by\_id*(id)

*return* *render\_template*('editgame.html', g=g)

**editgame.html**

{% extends 'base.html' %}

{% block content %}

<div *class*="col-md-6 m-auto">

    <h3 *style*="text-align: center;">Editar jogo:</h3><br>

    <form *action*="{{url\_for('edit', id=g.\_id)}}" *method*="POST">

        <input *type*="text" *name*="titulo" *class*="form-control" *value*="{{g.titulo}}" *required*>

        <input *type*="number" *name*="ano" *value*="{{g.ano}}" *min*="1970" *max*="2099" *class*="form-control" *required*>

        <input *type*="text" *name*="categoria" *value*="{{g.categoria}}" *class*="form-control" *required*>

        <input *type*="text" *name*="plataforma" *value*="{{g.plataforma}}" *class*="form-control" *required*>

        <input *type*="number" *name*="preco" *value*="{{g.preco}}" *min*=0 *step*="0.01" *class*="form-control" *required*>

        <input *type*="number" *name*="quantidade" *value*="{{g.quantidade}}" *class*="form-control" *required*><br>

        <a *href*="{{url\_for('edit', id=g.\_id)}}" *onclick*="*return* confirm('Confirma as alterações?')"><input *type*="submit" *value*="Alterar" *class*="btn btn-warning"></a>

        <a *href*="{{url\_for('estoque')}}"><input *type*="button" *value*="Cancelar" *class*="btn btn-danger"></a>

    </form>

</div>

{% endblock content %}

**estoque.html:**

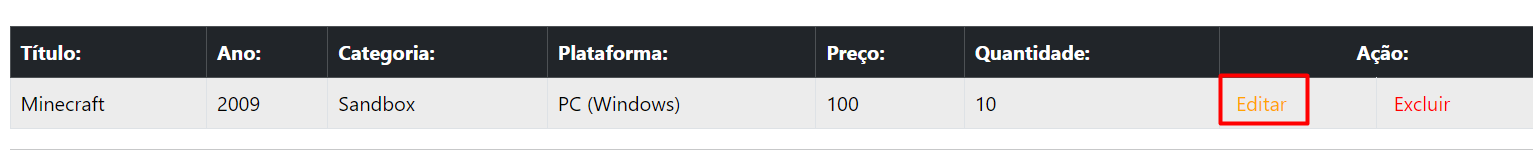
                <td><a *style*="text-decoration: none; color: orange;" *href*="{{url\_for('edit', id=g.\_id)}}">Editar</a></td>

                <td><a *style*="text-decoration: none; color: red;" *href*="{{url\_for('estoque', id=g.\_id)}}" *onclick*="*return* confirm('Deseja mesmo excluir o jogo {{g.titulo}}?')">Excluir</a></td>

            </tr>

        {% endfor %}

        </tbody>



Interface gráfica do usuário

Descrição gerada automaticamente com confiança média

**database.py:**

    def *update*(self, id):

        mongo.db.games.*update\_one*({'\_id': ObjectId(id)},

        {'$set':

            {

            'titulo': self.titulo,

            'ano': self.ano,

            'categoria': self.categoria,

            'plataforma': self.plataforma,

            'preco': self.preco,

            'quantidade': self.quantidade

            }

        })

**routes.py:**

    # *CRUD - EDIÇÃO*

    @app.*route*('/edit/<id>', methods=['GET', 'POST'])

    def *edit*(id):

        g =  Game.*get\_by\_id*(id)

        # *Edita o jogo com as informações do formulário*

*if* request.method *==* 'POST':

            editedgame = Game(request.form['titulo'], request.form['ano'], request.form['categoria'], request.form['plataforma'], request.form['preco'], request.form['quantidade'])

            Game.*update*(editedgame, id)

*return* *redirect*(*url\_for*('estoque'))

*return* *render\_template*('editgame.html', g=g)