

Mãos na massa: Arrumando a casa

Agora começaremos a fazer a divisão de arquivos baseado em suas finalidades. Para isso, execute os passos listados abaixo

Se você quiser o projeto completo feito no treinamento anterior, pode baixá-lo [aqui \(https://caelum-online-public.s3.amazonaws.com/739-python-flask2/04/aula4.zip\)](https://caelum-online-public.s3.amazonaws.com/739-python-flask2/04/aula4.zip).

1) Em **jogoteca.py**, recorte as linhas de configuração da aplicação:

```
app.secret_key = 'alura'
app.config['MYSQL_HOST'] = "0.0.0.0"
app.config['MYSQL_USER'] = "root"
app.config['MYSQL_PASSWORD'] = "admin"
app.config['MYSQL_DB'] = "jogoteca"
app.config['MYSQL_PORT'] = 3306
app.config['UPLOAD_PATH'] = os.path.dirname(os.path.abspath(__file__)) + '/uploads'
```

2) E no seu lugar, chame o novo arquivo de configuração que será criado:

```
app.config.from_pyfile('config.py')
```

3) Então, crie o arquivo **config.py** dentro dele, importe a biblioteca **os**, e adicione as linhas de configuração da aplicação que foram recortadas acima:

```
import os

SECRET_KEY = 'alura'
MYSQL_HOST = "0.0.0.0"
MYSQL_USER = "root"
MYSQL_PASSWORD = "admin"
MYSQL_DB = "jogoteca"
MYSQL_PORT = 3306
UPLOAD_PATH = os.path.dirname(os.path.abspath(__file__)) + '/uploads'
```

Com isso temos a configuração em somente um arquivo.

4) Agora recorte as funções **recupera_imagem** e **deleta_arquivo** de **jogoteca.py**. Crie o arquivo **helpers.py** e cole essas funções dentro dele:

```
import os
from jogoteca import app

def recupera_imagem(id):
    for nome_arquivo in os.listdir(app.config['UPLOAD_PATH']):
        if f'capa{id}' in nome_arquivo:
```

```
return nome_arquivo
```

```
def deleta_arquivo(id):
    arquivo = recupera_imagem(id)
    os.remove(os.path.join(app.config['UPLOAD_PATH'], arquivo))
```

5) Agora vamos isolar nossas *views*. Para isso, recorte as rotas de **jogoteca.py**, crie o arquivo **views.py**, e cole-as dentro dele. Não esqueça de ajustar os *imports*. O arquivo ficará do seguinte jeito:

```
from flask import render_template, request, redirect, session, flash, url_for, send_from_directory
from models import Jogo
from dao import JogoDao, UsuarioDao
import time
from helpers import deleta_arquivo, recupera_imagem
from jogoteca import db, app
```

```
jogo_dao = JogoDao(db)
usuario_dao = UsuarioDao(db)
```

```
@app.route('/')
def index():
    lista = jogo_dao.listar()
    return render_template('lista.html', titulo='Jogos', jogos=lista)
```

```
@app.route('/novo')
def novo():
    if 'usuario_logado' not in session or session['usuario_logado'] == None:
        return redirect(url_for('login', proxima=url_for('novo')))
    return render_template('novo.html', titulo='Novo Jogo')
```

```
@app.route('/criar', methods=['POST',])
def criar():
    nome = request.form['nome']
    categoria = request.form['categoria']
    console = request.form['console']
    jogo = Jogo(nome, categoria, console)
    jogo = jogo_dao.salvar(jogo)

    arquivo = request.files['arquivo']
    upload_path = app.config['UPLOAD_PATH']
    timestamp = time.time()
    arquivo.save(f'{upload_path}/capa{jogo.id}-{timestamp}.jpg')
    return redirect(url_for('index'))
```

```
@app.route('/editar/<int:id>')
def editar(id):
    if 'usuario_logado' not in session or session['usuario_logado'] == None:
        return redirect(url_for('login', proxima=url_for('editar')))
    jogo = jogo_dao.busca_por_id(id)
    nome_imagem = recupera_imagem(id)
    return render_template('editar.html', titulo='Editando Jogo', jogo=jogo,
                           , capa_jogo=nome_imagem or 'capa_padrao.jpg')
```

```
@app.route('/atualizar', methods=['POST',])
def atualizar():
    nome = request.form['nome']
    categoria = request.form['categoria']
    console = request.form['console']
    jogo = Jogo(nome, categoria, console, id=request.form['id'])

    arquivo = request.files['arquivo']
    upload_path = app.config['UPLOAD_PATH']
    timestamp = time.time()
    deleta_arquivo(jogo.id)
    arquivo.save(f'{upload_path}/capa{jogo.id}-{timestamp}.jpg')
    jogo_dao.salvar(jogo)
    return redirect(url_for('index'))

@app.route('/deletar/<int:id>')
def deletar(id):
    jogo_dao.deletar(id)
    flash('O jogo foi removido com sucesso!')
    return redirect(url_for('index'))

@app.route('/login')
def login():
    proxima = request.args.get('proxima')
    return render_template('login.html', proxima=proxima)

@app.route('/autenticar', methods=['POST', ])
def autenticar():
    usuario = usuario_dao.buscar_por_id(request.form['usuario'])
    if usuario:
        if usuario.senha == request.form['senha']:
            session['usuario_logado'] = usuario.id
            flash(usuario.nome + ' logou com sucesso!')
            proxima_pagina = request.form['proxima']
            return redirect(proxima_pagina)
        else:
            flash('Não logado, tente novamente!')
            return redirect(url_for('login'))

@app.route('/logout')
def logout():
    session['usuario_logado'] = None
    flash('Nenhum usuário logado!')
    return redirect(url_for('index'))

@app.route('/uploads/<nome_arquivo>')
def imagem(nome_arquivo):
    return send_from_directory('uploads', nome_arquivo)
```

6) Em seguida remova todos os *imports* que não são mais necessários (os que estão cinza) em **jogoteca.py**.

7) Agora finalize a **jogoteca.py**, resolvendo os *imports*, para poder iniciar a aplicação:

```
from flask import Flask
from flask_mysqlldb import MySQL

app = Flask(__name__)
app.config.from_pyfile('config.py')

db = MySQL(app)

from views import *

if __name__ == '__main__':
    app.run(debug=True)
```