



GRADUAÇÃO 4D
PUCPR



■ Hora da Prática 2

Programação para Ciência de Dados



HORA DA PRÁTICA 2

Depois de todo o conhecimento compartilhado durante cada uma das semanas de estudos, chegou o momento de nos testarmos e colocarmos em prática nossos saberes, resolvendo situações que podem surgir em nosso dia a dia profissional.

Para isso, observe, com atenção, o contexto apresentado. Esteja atento aos detalhes e se dedique ao máximo às entregas.

Cenário:

Projeto: Sistema de gerenciamento de biblioteca digital em python

Você é um desenvolvedor de *software* e foi contratado por uma biblioteca universitária para criar um sistema que facilite a gestão de documentos digitais.

A biblioteca, por sua vez, possui uma vasta coleção de artigos, teses e livros em formatos digitais, como PDF, ePUB, entre outros.

Atualmente, a gestão desses arquivos é feita manualmente, o que é ineficiente e propenso a erros. 😞 Dessa forma, esta é a situação que deve ser resolvida por você.... Vamos começar?

Descrição da atividade:

1. Criação do repositório:

- Inicie criando um repositório no GitHub para o projeto.
- Documente cada passo do desenvolvimento com *commits* claros e informativos.

2. Manipulação de arquivos e diretórios:

- Implemente funções em python para listar todos os documentos digitais, organizados por tipo de arquivo e por ano de publicação.
- Crie uma interface de linha de comando que permita aos bibliotecários adicionarem, renomearem e removerem documentos.

3. Integração com Git e GitHub:

- Configure o repositório para aceitar *pull requests*.
- Adicione um guia de contribuição explicando como realizar *commits*, *pushes* e *pull requests*.

4. Testes e feedback:

- Realize testes para garantir que todas as funções estão operando corretamente.



○ ○ ○ ○

- Peça *feedback* aos bibliotecários e faça ajustes conforme necessário.

O que devo entregar:

- Um repositório no GitHub contendo todo o código-fonte do sistema.
- Documentação detalhada de cada funcionalidade implementada.
- Relatório de testes e *feedback* recebido, incluindo como o *feedback* foi incorporado ao projeto.

Você será avaliado:

- Pela qualidade e pela clareza do código.
- Pela funcionalidade das operações de manipulação de arquivos e diretórios.
- Pelo uso efetivo do Git e GitHub para versionamento e colaboração.
- Pela entrega da documentação e dos relatórios completos.

⚠ Observe na tabela abaixo os critérios pelos quais você será avaliado.



RUBRICA – CRITÉRIO DE AVALIAÇÃO

Critério	Autônomo (90% a 100%)	Capaz (70% a 89%)	Aprendiz (40% a 69%)	Iniciante (1% a 39%)	Critério de Anulação (0%)
Desenvolve um sistema que permite manusear, eficientemente, os arquivos digitais (abrir, ler, criar, renomear e remover) e gerenciar diretórios (listar, criar e remover). 35%	Desenvolve o sistema adequadamente, demonstrando domínio nas funcionalidades implementadas, as quais operam perfeitamente e sem erros, com código bem-organizado e comentado , seguindo as convenções de estilo do python.	Desenvolve o sistema com a maioria das funcionalidades, as quais operam corretamente, com poucos erros e não críticos , demonstrando código bem-organizado e comentado , seguindo, com pequenas falhas, as convenções de estilo do python.	Desenvolve um sistema em que algumas funcionalidades não operam como esperado, com erros que afetam a operação , demonstrando um código razoavelmente organizado e comentado, demonstrando várias falhas nas convenções de estilo do python.	Desenvolve um sistema em que muitas funcionalidades não operam ou o sistema é inoperante , demonstrando código desorganizado, mal comentado e não segue as convenções de estilo do python.	Não realiza a entrega.
Demonstra domínio no uso do Git e GitHub, considerando <i>commits</i> , <i>pushes</i> e <i>pull requests</i> (requisições) de código para o repositório remoto. 30%	Demonstra domínio exemplar no uso do Git e GitHub, considerando commits claros e descritivos , além de repositório bem configurado .	Demonstra bom uso do Git e GitHub, mas com alguns commits pouco claros ou repositório com configuração incompleta .	Demonstra uso básico do Git e GitHub, com commits genéricos e configuração mínima do repositório.	Demonstra falha no uso do Git e GitHub, com commits confusos e repositório mal configurado .	Não realiza a entrega.
Apresenta documentação e relatórios de acordo com a situação-problema a ser resolvida. 35%	Apresenta documentação completa e detalhada , com relatórios de testes e <i>feedbacks bem integrados</i> , demonstrando uma solução criativa e eficaz para a situação-problema, com análise e posicionamento avançados .	Apresenta documentação ok , mas menos detalhada , com relatórios de testes e <i>feedbacks razoáveis</i> , demonstrando solução adequada para a situação-problema, com boa análise e posicionamento .	Apresenta documentação básica , mas menos detalhada , com relatórios de testes e <i>feedbacks com informações mínimas</i> , demonstrando solução parcial para a situação-problema, com análise e posicionamento superficiais .	Apresenta documentação e relatórios insuficientes ou ausentes , demonstrando falha em responder adequadamente à situação-problema.	Não realiza a entrega.



GRADUAÇÃO 4D
PUCPR

