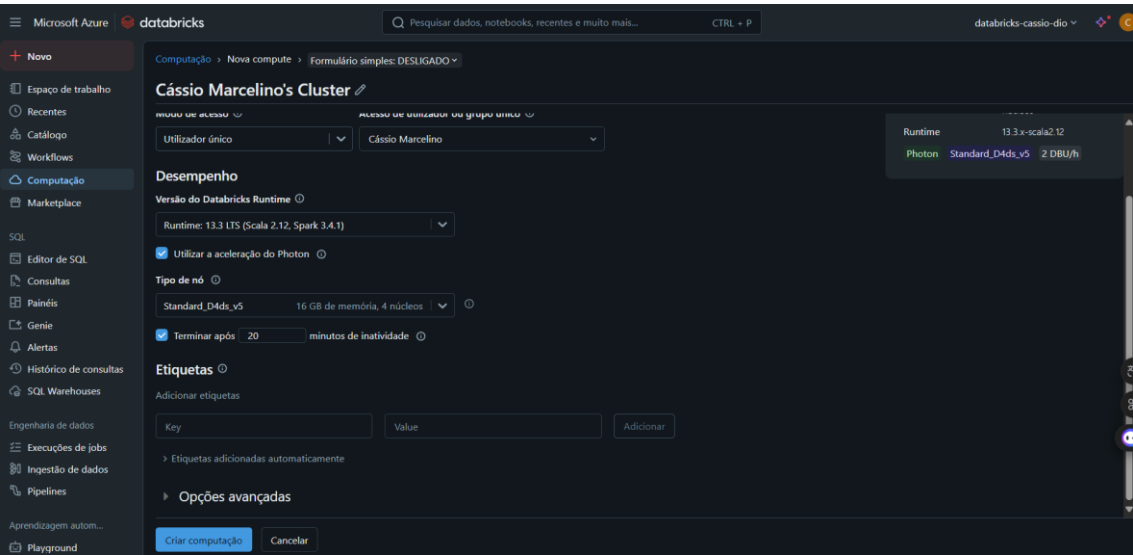
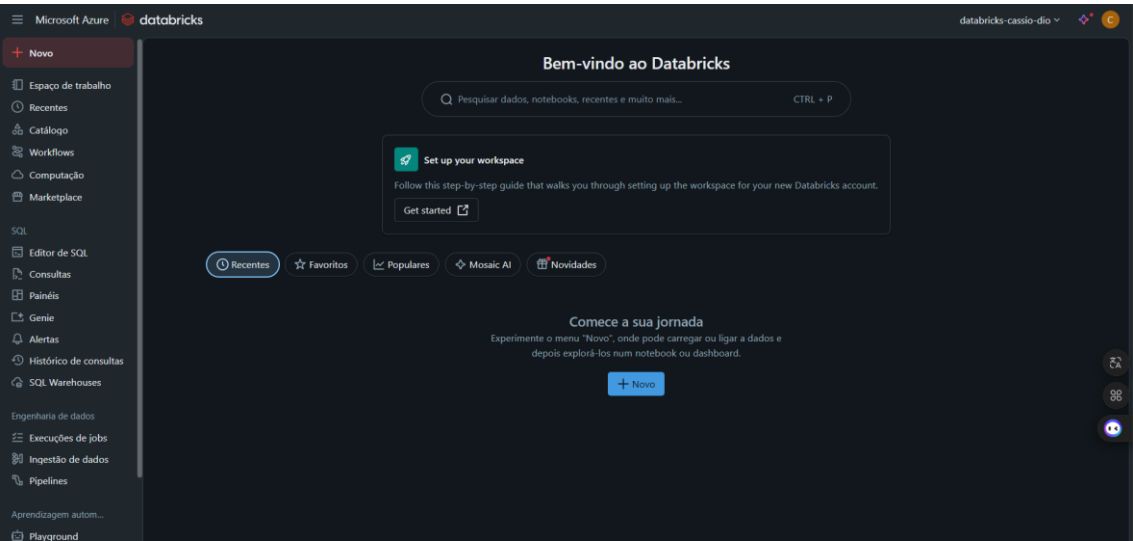
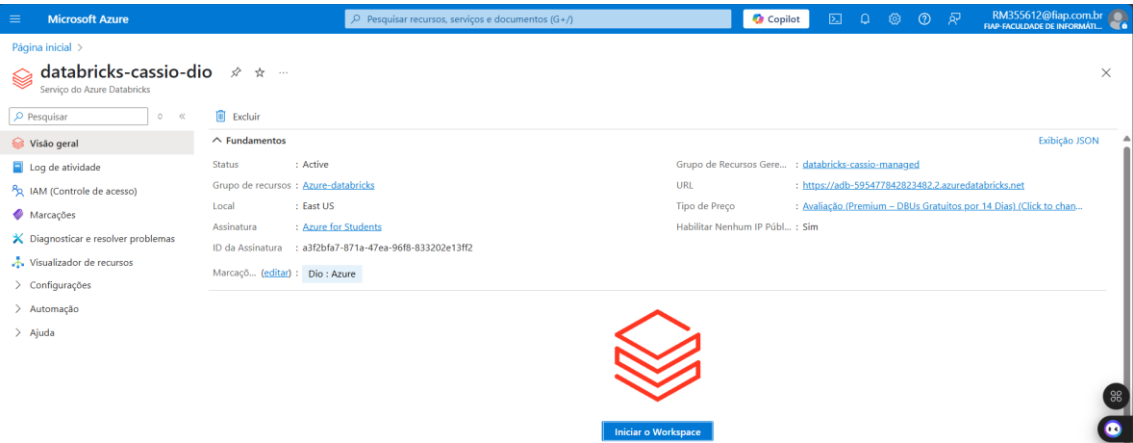


# Controle e Versionamento de Código no Notebook da Azure



Microsoft Azure databricks

hive

databricks-cassio-dio

Adicionar dados

### Criar ou modificar tabela a partir de upload de ficheiro

products.csv carregados 14.37KB

Modo de pré-visualização

Catálogo

Esquema

Nome da tabela

hive\_metastore

default

products

Atributos avançados

A pré-visualizar 50 linhas, 4 colunas

ProductID	ProductName	Category	ListPrice
771	Mountain-100 Silver, 38	Mountain Bikes	3399.99
772	Mountain-100 Silver, 42	Mountain Bikes	3399.99
773	Mountain-100 Silver, 44	Mountain Bikes	3399.99
774	Mountain-100 Silver, 48	Mountain Bikes	3399.99
775	Mountain-100 Black, 38	Mountain Bikes	3374.99
776	Mountain-100 Black, 42	Mountain Bikes	3374.99
777	Mountain-100 Black, 44	Mountain Bikes	3374.99
778	Mountain-100 Black, 48	Mountain Bikes	3374.99
779	Mountain-200 Silver, 38	Mountain Bikes	2319.99
780	Mountain-200 Silver, 42	Mountain Bikes	2319.99

Microsoft Azure databricks

hive

databricks-cassio-dio

### Explore hive\_metastore.default.products 2025-04-28 00:13:30

Python

Arquivo Editar Ver Executar Ajuda

Executar tudo

Ligar

Agenda

Partilhar

O teste grátis termina dentro de 14 dias. Faça upgrade para a versão Premium no Azure Portal

1

```
%sql
SELECT * FROM 'hive_metastore'.default.products;
```

[Shift+Enter] para executar e passar para a célula seguinte  
[Ctrl+Shift+P] para abrir a paleta de comandos  
[Esc H] para ver todos os atalhos do teclado

Microsoft Azure databricks

hive

databricks-cassio-dio

### Explore hive\_metastore.default.products 2025-04-28 00:13:30

Python

Arquivo Editar Ver Executar Ajuda Última edição foi há 10 minutos

Interromper

A iniciar

Agenda

Partilhar

2

```
%sql
SELECT * FROM 'hive_metastore'.default.products;
```

Em espera Finding instances for new nodes, acquiring more instances if necessary

```
df = spark.sql("SELECT * FROM products")
df = df.filter(Category == 'Road Bikes')
display(df)
```

[Shift+Enter] para executar e passar para a célula seguinte  
[Ctrl+Shift+P] para abrir a paleta de comandos  
[Esc H] para ver todos os atalhos do teclado

## **Resumo do Projeto**

Ao longo deste projeto, explorei a fundo o Azure Databricks e como ele pode ser utilizado para versionamento e organização de notebooks em ambientes de dados. Inicialmente, eu tinha uma visão básica sobre análise de dados, mas conforme fui avançando na configuração, comecei a perceber o quão essencial é ter um ambiente estruturado para garantir colaboração, segurança e escalabilidade no trabalho.

O primeiro passo foi configurar o Databricks, criando um cluster para processamento distribuído. Isso permitiu que as tarefas fossem executadas de forma paralela, melhorando a velocidade das análises. Depois, precisei importar arquivos e dados para dentro do ambiente, garantindo que as informações pudessem ser utilizadas nos notebooks interativos.

A parte mais interessante foi começar a executar os notebooks com suporte à inteligência artificial. Foi incrível ver como o Databricks pode gerar código automaticamente para otimizar processos. Pude criar notebooks organizados, com filtros, sumarizações e visualizações interativas, o que facilitou muito a interpretação dos dados.

Outro ponto essencial do processo foi a integração com Azure DevOps, que me permitiu manter controle de versão dos notebooks. Isso foi um grande aprendizado, pois percebi que ter um histórico de mudanças evita problemas como perda de progresso e permite colaboração eficiente entre equipes.

Ao longo do projeto, alguns pontos me chamaram atenção:

- Automação e inteligência artificial realmente fazem diferença! Com a IA integrada ao Databricks, consegui gerar código Python e Spark sem precisar escrever tudo manualmente, o que acelerou meu trabalho.
- Ter um ambiente bem organizado melhora muito a produtividade. Antes, meus notebooks estavam bagunçados, sem um fluxo de trabalho claro. Após estruturar em camadas e aplicar boas práticas, tudo ficou mais intuitivo.
- Versionamento é fundamental para projetos maiores. No começo, eu não via tanta necessidade de manter controle de versão, mas percebi que, ao integrar com Azure DevOps, ficou muito mais seguro e fácil administrar as edições dos notebooks.
- O Databricks é muito mais poderoso do que eu imaginava! Inicialmente, eu achava que era apenas uma ferramenta de processamento, mas vi que ele pode ser personalizado e otimizado para colaboração em engenharia de dados e machine learning.

Esse projeto me ensinou muito sobre organização, versionamento e automação de notebooks dentro do Azure Databricks. Sair da teoria e colocar tudo em prática foi essencial para entender como grandes volumes de dados podem ser gerenciados de forma eficiente e colaborativa.

Agora que conheço essas ferramentas, percebo como elas podem facilitar projetos reais, seja na área de análise de dados, engenharia de dados ou machine learning. As possibilidades são enormes, e vejo que dominar essas práticas pode abrir muitas portas no mercado de tecnologia!