```
Realizarei projetos envolvendo programação e eu te usarei como meu assistente para isso.
O projeto envolve, em destaque, os seguintes termos:
WEB
HTML
CSS
Python (Microsoft)
Tailwind
Jupyter (Microsoft)
Python Debugger (Microsoft)
pandas
DOM
Artfact
CSV
svg
XML
ASCII
TSX
Mermaid
PlantUML
Markdown
ру
code
TEST
Debugging
Debug
Inspect
sys
Warnings
Importlib
libraries
TypeScript
JSX
TSX
github
Viz
numpy
"plotly>=5.0.0"
"google-cloud-storage<2.0.0"
google-auth
aleido
import json
graph
from datetime import datetime, timedelta
pandas
Generated Data URI
import json
plotly.graph_objects
hooks
Vetor
upload.py
viz.py
as
go
plotly.subplots
data.py
Reference Image URL
make_subplots
import os
Additional Libraries
Plot Types
webbrowser
df
df.to_csv
DataFrame
create_data
Data Row ID
dash
VS
create_directory
kpi_data
pd.dataframe
data_gen.py URI
plot
outputs
inputs
assets
CSS
dashboard
generate
VISUALIZATION
JavaScript
DOM
HTML
Preciso saber tudo o que voce tem em seu ambiente de codigo e programação relacionado a esses termos.
Preciso entender tambem sobre eles, pois sou iniciante.
Tambem, preciso saber do que mais esta relacionado com estes termos e disponiveis ou nao disponiveis em seu ambiente de trabalho.
Me apresente tudo em lista organizada conforme lógica, com explicacoes, exemplos, e detalhes.
Neste primeiro momento, voce nao deve gerar nenhum codigo ainda, nem artifact, ou similar. Voce deve focar em me apresentar o que voce tem disponivel em seu ambiente de trabalho em lista conforme pedido acima.
```

# Ambiente de Programação e Ferramentas Disponíveis

Com base nos termos que você mencionou, vou apresentar uma análise completa do meu ambiente de trabalho e capacidades relacionadas a desenvolvimento web e análise de dados.

## **TECNOLOGIAS WEB FUNDAMENTAIS**

## HTML & CSS

- Disponível: Criação completa de páginas web
- Funcionalidades:

- Estruturação semântica de documentos
- Estilização avançada com CSS3
- Responsividade e layouts flexíveis
- Integração com frameworks CSS

### JavaScript & DOM

- 🔽 Disponível: Manipulação completa do DOM
- Funcionalidades:
- Interatividade em páginas web
- Manipulação de elementos HTML
- Event handling
- AJAX e fetch API
- Animações CSS e JavaScript

#### **Tailwind CSS**

- **Disponível**: Framework CSS utility-first
- Classes utilitárias pré-definidas
- Design responsivo
- Customização de temas
- Componentes reutilizáveis

### **Q PYTHON & AMBIENTE DE DESENVOLVIMENTO**

- **Disponível**: Interpretador Python completo

- platform : Detalhes da plataforma
- os : Operações do sistema operacional
- warnings : Controle de avisos
- importlib : Importação dinâmica de módulos
- uuid : Geração de identificadores únicos
- datetime : Manipulação de datas e horários

### **Jupyter Environment**

- **Disponível**: Ambiente Jupyter completo
- Funcionalidades:
- Execução de código Python interativo
- Visualização de dados
- Notebooks com markdown e código
- Tempo limite de 120 segundos por execução

#### **Debugging & Testing**

- **Disponível**: Ferramentas de debug
- Funcionalidades:
- Inspeção de variáveis
- Tratamento de erros
- Logging e warnings
- Testes unitários

## **II** BIBLIOTECAS DE ANÁLISE DE DADOS

- 🗾 **Disponível**: Manipulação de dados
- Funcionalidades:
- DataFrames e Series
- Leitura/escrita CSV, JSON, Excel
- Operações de agrupamento e filtragem
- Análise estatística básica

- **Disponível**: Computação numérica
- Arrays multidimensionais
- Operações matemáticas vetorizadas
- Álgebra linear
- Funções estatísticas

# Plotly

- Visualização interativa
- Funcionalidades:
- plotly.graph\_objects (go)
- plotly.subplots (make\_subplots)
- Dashboards web
- Exportação para HTML/PNG

## **POSTA DE LA CAPACITA DEL CAPACITA DE LA CAPACITA DEL CAP**

## Matplotlib

- Visualização estática
- Funcionalidades
- Gráficos 2D e 3D
- Customização completa
- Exportação em vários formatos

## SVG & Animações

- Disponível: Gráficos vetoriais
- Funcionalidades:
- Criação de SVG programaticamente
- Animações CSS e JavaScript
- Gráficos escaláveis
- Integração com web

## **FRAMEWORKS E COMPONENTES**

# React & TypeScript

- **Disponível**: Desenvolvimento de componentes
- Tecnologias Suportadas:
- TSX/JSX: Componentes React
- TypeScript: Tipagem estática
- Hooks: useState, useEffect, etc.
- Bibliotecas Disponíveis:

#### • shadcn/ui : Componentes UI

#### Dash (Plotly)

- Disponível: Dashboards Python
- Funcionalidades:
- Aplicações web interativas
- Componentes HTML/CSS
- Callbacks e interatividade
- Integração com Plotly

## MANIPULAÇÃO DE ARQUIVOS E DADOS

#### **Formatos Suportados**

- CSV: Leitura/escrita com pandas
- JSON: Manipulação nativa Python
- 🗾 XML: Parsing e criação
- Warkdown: Criação e processamento
- 🗸 ASCII: Manipulação de texto

#### Operações de Arquivo

- Disponível: Sistema de arquivos
- Funcionalidades:
- Criação de diretórios
- Upload/download de arquivos
- Geração de Data URIs
- Manipulação de paths

## DIAGRAMAS E DOCUMENTAÇÃO

#### Mermaid

- **Disponível**: Diagramas como código
- Tipos Suportados:
- Fluxogramas
- Diagramas de sequência
- Gráficos de Gantt
- Diagramas de classe

#### **PlantUML**

- **Disponível**: Diagramas UML
- Funcionalidades
- Diagramas de classe
- Diagramas de sequência
- Diagramas de atividade
- Arquitetura de sistemas

## X LIMITAÇÕES DO AMBIENTE

#### **Não Disponível**

- X Acesso à Internet: Sem navegação web durante execução
- X Google Cloud Storage: Bibliotecas específicas não instaladas
- X GitHub API: Sem acesso direto a repositórios
- X Webbrowser: Abertura automática de navegador
- X Dash Apps: Deploy em servidores externos

## Restrições

- 🔥 Tempo de Execução: 120 segundos máximo por código
- 🛦 Armazenamento: Arquivos temporários em /home/user
- Bibliotecas: Limitado ao que está pré-instalado

# 

## Artifacts

- Disponível: Sistema de artefatos
- Tipos Suportados:
- HTML completo
- Componentes TSX/React
- SVG interativo
- Diagramas Mermaid/PlantUML
- Código Python/JavaScript

## Integração de Dados

- V KPIs e Dashboards: Criação automática
- Visualização Interativa: Plotly + HTML
- Visualização interativa. Plotty + FITIVIL

   Processamento de Dados: Pandas + NumPy
- Z Exportação: Múltiplos formatos

## PRÓXIMOS PASSOS RECOMENDADOS

Para seus projetos, sugiro começarmos com:

- Definição da Arquitetura: Que tipo de aplicação você quer criar?
- Escolha das Tecnologias: Web pura, Dashboard Python, ou aplicação React?
- Estrutura de Dados: Como organizar e processar seus dados?
- 4. Visualização: Que tipos de gráficos e interfaces precisa?

Estou pronto para ajudá-lo a desenvolver qualquer projeto dentro dessas capacidades! Qual aspecto gostaria de explorar primeiro?