```
!pip3 install fastapi uvicorn scikit-learn matplotlib seaborn pyngrok streamlit
```

```
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (2.8.2)
Requirement already satisfied: pandas>=1.2 in /usr/local/lib/python3.11/dist-packages (from seaborn) (2.2.2)
Requirement already satisfied: PyYAML>=5.1 in /usr/local/lib/python3.11/dist-packages (from pyngrok) (6.0.2)
Requirement already satisfied: altair<6,>=4.0 in /usr/local/lib/python3.11/dist-packages (from streamlit) (5.5.0)
Requirement already satisfied: blinker<2,>=1.0.0 in /usr/local/lib/python3.11/dist-packages (from streamlit) (1.9.0)
Requirement already satisfied: cachetools<6,>=4.0 in /usr/local/lib/python3.11/dist-packages (from streamlit) (5.5.1)
Requirement already satisfied: protobuf<6,>=3.20 in /usr/local/lib/python3.11/dist-packages (from streamlit) (4.25.6)
Requirement already satisfied: pyarrow>=7.0 in /usr/local/lib/python3.11/dist-packages (from streamlit) (17.0.0)
Requirement already satisfied: requests<3,>=2.27 in /usr/local/lib/python3.11/dist-packages (from streamlit) (2.32.3)
Requirement already satisfied: rich<14,>=10.14.0 in /usr/local/lib/python3.11/dist-packages (from streamlit) (13.9.4)
Requirement already satisfied: tenacity<10,>=8.1.0 in /usr/local/lib/python3.11/dist-packages (from streamlit) (9.0.0)
Requirement already satisfied: toml<2,>=0.10.1 in /usr/local/lib/python3.11/dist-packages (from streamlit) (0.10.2)
Collecting watchdog<7,>=2.1.5 (from streamlit)
  Downloading watchdog-6.0.0-py3-none-manylinux2014_x86_64.whl.metadata (44 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 44.3/44.3 kB 2.6 MB/s eta 0:00:00
Requirement already satisfied: gitpython!=3.1.19,<4,>=3.0.7 in /usr/local/lib/python3.11/dist-packages (from streamlit)
Collecting pydeck<1,>=0.8.0b4 (from streamlit)
  Downloading pydeck-0.9.1-py2.py3-none-any.whl.metadata (4.1 kB)
Requirement already satisfied: tornado<7,>=6.0.3 in /usr/local/lib/python3.11/dist-packages (from streamlit) (6.4.2)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.11/dist-packages (from altair<6,>=4.0->streamlit) (3.1.5
Requirement already satisfied: jsonschema>=3.0 in /usr/local/lib/python3.11/dist-packages (from altair<6,>=4.0->streamli
Requirement already satisfied: narwhals>=1.14.2 in /usr/local/lib/python3.11/dist-packages (from altair<6,>=4.0->streaml
Requirement already satisfied: gitdb<5,>=4.0.1 in /usr/local/lib/python3.11/dist-packages (from gitpython!=3.1.19,<4,>=3
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas>=1.2->seaborn) (2025
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas>=1.2->seaborn) (20
Requirement already satisfied: annotated-types>=0.6.0 in /usr/local/lib/python3.11/dist-packages (from pydantic!=1.8,!=1
Requirement already satisfied: pydantic-core==2.27.2 in /usr/local/lib/python3.11/dist-packages (from pydantic!=1.8,!=1.
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.7->matplotli
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests<3,>=2.
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests<3,>=2.27->streamli
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests<3,>=2.27->st
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests<3,>=2.27->st
Requirement already satisfied: markdown-it-py>=2.2.0 in /usr/local/lib/python3.11/dist-packages (from rich<14,>=10.14.0-
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in /usr/local/lib/python3.11/dist-packages (from rich<14,>=10.14.
Requirement already satisfied: anyio<5,>=3.6.2 in /usr/local/lib/python3.11/dist-packages (from starlette<0.46.0,>=0.40.
Requirement already satisfied: sniffio>=1.1 in /usr/local/lib/python3.11/dist-packages (from anyio<5,>=3.6.2->starlette<
Requirement already satisfied: smmap<6,>=3.0.1 in /usr/local/lib/python3.11/dist-packages (from gitdb<5,>=4.0.1->gitpyth
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.11/dist-packages (from jinja2->altair<6,>=4.0->
Requirement already satisfied: attrs>=22.2.0 in /usr/local/lib/python3.11/dist-packages (from jsonschema>=3.0->altair<6,
Requirement already satisfied: jsonschema-specifications>=2023.03.6 in /usr/local/lib/python3.11/dist-packages (from jso
Requirement already satisfied: referencing>=0.28.4 in /usr/local/lib/python3.11/dist-packages (from jsonschema>=3.0->alt
Requirement already satisfied: rpds-py>=0.7.1 in /usr/local/lib/python3.11/dist-packages (from jsonschema>=3.0->altair<6
Requirement already satisfied: mdurl~=0.1 in /usr/local/lib/python3.11/dist-packages (from markdown-it-py>=2.2.0->rich<1
Downloading fastapi-0.115.8-py3-none-any.whl (94 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 94.8/94.8 kB 8.6 MB/s eta 0:00:00
Downloading uvicorn-0.34.0-py3-none-any.whl (62 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 62.3/62.3 kB 6.1 MB/s eta 0:00:00
Downloading pyngrok-7.2.3-py3-none-any.whl (23 kB)
Downloading streamlit-1.42.1-py2.py3-none-any.whl (9.6 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 9.6/9.6 MB 95.3 MB/s eta 0:00:00
Downloading pydeck-0.9.1-py2.py3-none-any.whl (6.9 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 6.9/6.9 MB 88.3 MB/s eta 0:00:00
Downloading starlette-0.45.3-py3-none-any.whl (71 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 71.5/71.5 kB 5.9 MB/s eta 0:00:00
Downloading watchdog-6.0.0-py3-none-manylinux2014_x86_64.whl (79 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 79.1/79.1 kB 7.9 MB/s eta 0:00:00
Installing collected packages: watchdog, uvicorn, pyngrok, starlette, pydeck, fastapi, streamlit
Successfully installed fastapi-0.115.8 pydeck-0.9.1 pyngrok-7.2.3 starlette-0.45.3 streamlit-1.42.1 uvicorn-0.34.0 watch
```

```python
import streamlit as st
import pandas as pd
import numpy as np
from fastapi import FastAPI, HTTPException, Query
from typing import Optional
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
import uvicorn
import threading


try:
    data = pd.read_csv("/content/dataset.csv")
except FileNotFoundError:
    print("Error: File Not Found")
    exit()


element_mapping = {
    "Emissions Share (CH4)": "Emissions_Share_CH4",
    "Emissions Share (CO2)": "Emissions_Share_CO2",
    "Emissions Share (CO2eq) (AR5)": "Emissions_Share_CO2eq_AR5",
    "Emissions Share (CO2eq) (AR5) (F-gases)": "Emissions_Share_CO2eq_AR5_F_gases",
    "Emissions Share (N2O)": "Emissions_Share_N2O",
    "Emissions per area of agricultural land": "Emissions_per_area",
    "Emissions per capita": "Emissions_per_capita",
    "Emissions per value of agricultural production": "Emissions_per_value",
}
```

```
print(data.columns)
```

```
Index(['Domain Code', 'Domain', 'Area Code (M49)', 'Area', 'Element Code',
       'Element', 'Item Code', 'Item', 'Year Code', 'Year', 'Unit', 'Value',
       'Flag', 'Flag Description'],
      dtype='object')
```

```
print(data.head())
```

```
  Domain Code                Domain  Area Code (M49)   Area  Element Code  \
0          EM  Emissions indicators              356  India        726313   
1          EM  Emissions indicators              356  India        726313   
2          EM  Emissions indicators              356  India        726313   
3          EM  Emissions indicators              356  India        726313   
4          EM  Emissions indicators              356  India        726313   

                         Element  Item Code       Item  Year Code  Year Unit  \
0  Emissions Share (CO2eq) (AR5)       6996  Farm gate       1990  1990    %   
1  Emissions Share (CO2eq) (AR5)       6996  Farm gate       1991  1991    %   
2  Emissions Share (CO2eq) (AR5)       6996  Farm gate       1992  1992    %   
3  Emissions Share (CO2eq) (AR5)       6996  Farm gate       1993  1993    %   
4  Emissions Share (CO2eq) (AR5)       6996  Farm gate       1994  1994    %   

   Value Flag Flag Description  
0  51.99    E  Estimated value  
1  50.97    E  Estimated value  
2  49.92    E  Estimated value  
3  50.01    E  Estimated value  
4  49.00    E  Estimated value  
```

```
data = data.groupby(['Area', 'Year', 'Element'])['Value'].mean().unstack().reset_index()
data.rename(columns=element_mapping, inplace=True)


data.dropna(inplace=True)


print(data)
```

```
15                 2.953333
16                 2.843333
17                 2.626667
18                 2.626667
19                 2.676667
20                 2.466667
21                 2.306667
22                 2.273333
23                 2.253333
24                 2.223333
25                 2.203333
26                 2.140000
27                 2.036667
28                 1.990000
29                 1.973333
30                 1.950000
31                 1.856667
32                 1.846667
```

```
numeric_cols = list(element_mapping.values())
data[numeric_cols] = data[numeric_cols].apply(pd.to_numeric, errors='coerce')
data.dropna(inplace=True)


data['Total_Emissions_Share'] = data[[col for col in data.columns if col.startswith('Emission_Share_')]].sum(axis=1)


X = data.drop(["Emissions_per_area", "Area", "Year"], axis=1)
y = 1 / (data["Emissions_per_area"] + 1e-9)


numerical_features = X.select_dtypes(include=np.number).columns.tolist()
categorical_features = [col for col in X.columns if col not in numerical_features]


preprocessor = ColumnTransformer(
    transformers=[
        ('num', StandardScaler(), numerical_features),
        ('cat', OneHotEncoder(handle_unknown='ignore'), categorical_features)
    ],
    remainder='passthrough' #Handles any unexpected column more robustly
)


pipeline = Pipeline(steps=[('preprocessor', preprocessor),
                           ('regressor', RandomForestRegressor(random_state=42))])


X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
pipeline.fit(X_train, y_train)
y_pred = pipeline.predict(X_test)


mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print(f"MSE : {mse} ")
print(f"r2 score : {r2}")
```

```
MSE : 1.0402906987608268e-05
r2 score : 0.9606877851486674
```

```
feature_names = pipeline.named_steps['preprocessor'].get_feature_names_out()
importances = pipeline.named_steps['regressor'].feature_importances_
feature_importances_df = pd.DataFrame({'Feature': feature_names, 'Importance': importances}).sort_values(by='Importance', as


print("\n Feature Importances:")
print(feature_importances_df)
```

```
 Feature Importances:
                                Feature  Importance
5               num__Emissions_per_capita    0.260900
2          num__Emissions_Share_CO2eq_AR5    0.253911
6                num__Emissions_per_value    0.171642
0                num__Emissions_Share_CH4    0.137198
1                num__Emissions_Share_CO2    0.103669
4                num__Emissions_Share_N2O    0.047022
3  num__Emissions_Share_CO2eq_AR5_F_gases    0.025658
7              num__Total_Emissions_Share    0.000000
```

```
app = FastAPI(title="Carbon Credits and Emissions Prediction API")
```

```
@app.get("/predict_emissions")
async def predict_emissions(
    area: str = Query(..., description="Area (e.g., country)"),
    year: int = Query(..., description="Year"),
    ch4_emissions_share: float = Query(..., description="Emissions Share (CH4)"),
    co2_emissions_share: float = Query(..., description="Emissions Share (CO2)"),
    co2eq_ar5_emissions_share: float = Query(..., description="Emissions Share (CO2eq) (AR5)"),
    co2eq_ar5_fgases_emissions_share: float = Query(..., description="Emissions Share (CO2eq) (AR5) (F-gases)"),
    n2o_emissions_share: float = Query(..., description="Emissions Share (N2O)"),
    emissions_per_capita: float = Query(..., description="Emissions per capita"),
    emissions_per_value: float = Query(..., description="Emissions per value of agricultural production"),
):
    """Predicts emissions per area using the trained model."""

    try:
        future_data = pd.DataFrame({
            'Area': [area],
            'Year': [year],
            'Emissions_Share_CH4': [ch4_emissions_share],
            'Emissions_Share_CO2': [co2_emissions_share],
            'Emissions_Share_CO2eq_AR5': [co2eq_ar5_emissions_share],
            'Emissions_Share_CO2eq_AR5_F_gases': [co2eq_ar5_fgases_emissions_share],
            'Emissions_Share_N2O': [n2o_emissions_share],
            'Emissions_per_capita': [emissions_per_capita],
            'Emissions_per_value': [emissions_per_value],
        })

        future_data['Total_Emissions_Share'] = future_data[[col for col in future_data.columns if col.startswith('Emissions_

        X_future = future_data.drop(['Area', 'Year'], axis=1)  # Drop target

        X_future_transformed = pipeline.named_steps['preprocessor'].transform(X_future)

        feature_names = pipeline.named_steps['preprocessor'].get_feature_names_out()
        X_future_transformed = pd.DataFrame(X_future_transformed, columns=feature_names)

        future_predictions = pipeline.predict(X_future)
        predicted_emissions_per_area = 1 / (future_predictions + 1e-9)

        return {
            "area": area,
            "year": year,
            "predicted_emissions_per_area": predicted_emissions_per_area[0],
        }

    except Exception as e:
        raise HTTPException(status_code=500, detail=f"An error occurred: {e}")



@app.get("/calculate_carbon_credits")
async def calculate_carbon_credits(
    area: str = Query(..., description="Area"),
    year: int = Query(..., description="Year"),
    ch4_emissions_share: float = Query(..., description="Emissions Share (CH4)"),
    co2_emissions_share: float = Query(..., description="Emissions Share (CO2)"),
    co2eq_ar5_emissions_share: float = Query(..., description="Emissions Share (CO2eq) (AR5)"),
    co2eq_ar5_fgases_emissions_share: float = Query(..., description="Emissions Share (CO2eq) (AR5) (F-gases)"),
    n2o_emissions_share: float = Query(..., description="Emissions Share (N2O)"),
    emissions_per_capita: float = Query(..., description="Emissions per capita"),
    emissions_per_value: float = Query(..., description="Emissions per value of agricultural production"),
    baseline_emissions_per_area: Optional[float] = Query(None, description="Baseline emissions per area (optional)"),
    reduction_achieved: Optional[float] = Query(None, description="Emission reduction achieved (optional)")

):
    """Calculates carbon credits (simplified example)."""
    try:
        # 1. Get predicted emissions per area (same logic as in /predict_emissions)
        future_data = pd.DataFrame({  # Create the input DataFrame
            'Area': [area],
            'Year': [year],
            'Emissions_Share_CH4': [ch4_emissions_share],
            'Emissions_Share_CO2': [co2_emissions_share],
            'Emissions_Share_CO2eq_AR5': [co2eq_ar5_emissions_share],
            'Emissions_Share_CO2eq_AR5_F_gases': [co2eq_ar5_fgases_emissions_share],
            'Emissions_Share_N2O': [n2o_emissions_share],
            'Emissions_per_capita': [emissions_per_capita],
            'Emissions_per_value': [emissions_per_value],
        })
        future_data['Total_Emissions_Share'] = future_data[[col for col in future_data.columns if col.startswith('Emissions_

        X_future = future_data.drop(['Area', 'Year'], axis=1)
        X_future_transformed = pipeline.named_steps['preprocessor'].transform(X_future)
        feature_names = pipeline.named_steps['preprocessor'].get_feature_names_out()
        X_future_transformed = pd.DataFrame(X_future_transformed, columns=feature_names)
        future_predictions = pipeline.predict(X_future)
        predicted_emissions_per_area = 1 / (future_predictions + 1e-9)
        predicted_emissions_per_area = predicted_emissions_per_area[0]


        # 2. Calculate carbon credits
        if baseline_emissions_per_area is None:
            baseline_emissions_per_area = predicted_emissions_per_area

        if reduction_achieved is not None:
            carbon_credits = reduction_achieved
        else:
            carbon_credits = baseline_emissions_per_area - predicted_emissions_per_area

        if carbon_credits < 0:
            carbon_credits = 0

        return {
            "area": area,
            "year": year,
            "predicted_emissions_per_area": predicted_emissions_per_area,
            "baseline_emissions_per_area": baseline_emissions_per_area,
            "potential_carbon_credits": carbon_credits,
            "message": "Carbon credit calculation (simplified example). See documentation for important considerations.",
        }

    except Exception as e:
        raise HTTPException(status_code=500, detail=f"An error occurred: {e}")
```

```
tle ("Carbon Credit and Emissions Prediction App")

debar.header("Input Parameters")
= st.sidebar.text_input("Area (eg: Your Country)")
= st.sidebar.number_input("Year", min_value=1900, max_value=2100, value=2025)
missions_share = st.sidebar.number_input("Emissions Share (CH4)", min_value=0.0, value=0.0)
missions_share = st.sidebar.number_input("Emissions Share (CO2)", min_value=0.0, value=0.0)
_ar5_emissions_share = st.sidebar.number_input("Emissions Share (CO2eq) (AR5)", min_value=0.0, value=0.0)
_ar5_fgases_emissions_share = st.sidebar.number_input("Emissions Share (CO2eq) (AR5) (F-gases)", min_value=0.0, value=0.0)
missions_share = st.sidebar.number_input("Emissions Share (N2O)", min_value=0.0, value=0.0)
ions_per_capita = st.sidebar.number_input("Emissions per capita", min_value=0.0, value=0.0)
ions_per_value = st.sidebar.number_input("Emissions per value of agricultural production", min_value=0.0, value=0.0)
ine_emissions_per_area = st.sidebar.number_input("Baseline Emissions per Area (optional)", min_value=0.0, value=0.0)
tion_achieved = st.sidebar.number_input("Reduction Achieved (optional)", min_value=0.0, value=0.0)

.sidebar.button("Predict and Calculate"):
f not area:
   st.error("Please Provide Area.")
lse:
   try:
       import requests
       prediction_url = f"{ngrok_url}/prediction_emissions?area={area}&year={year}&ch4_emissions_share={ch4_emissions_share}&c
       predict_response = requests.get(predict_url).json()

       calculate_url = f"{ngrok_url}/calculate_carbon_credits?area={area}&year={year}&ch4_emissions_share={ch4_emissions_share
       calculate_response = requests.get(calculate_url).json()

       st.subheader("Results")
       st.write(f"**Area:** {calculate_response['area']}")
       st.write(f"**Year:** {calculate_response['year']}")
       st.write(f"**Predicted Emissions per Area:** {calculate_response['predicted_emissions_per_area']:.2f}")
       st.write(f"**Baseline Emissions per Area:** {calculate_response['baseline_emissions_per_area']:.2f}")
       st.write(f"**Potential Carbon Credits:** {calculate_response['potential_carbon_credits']:.2f}")
       st.write(calculate_response['message'])

   except requests.exceptions.RequestException as e:
       st.error(f"Error communicating with the API: {e}")
   except Exception as e:
       st.error(f"An error occurred: {e}")
```

```
2025-02-18 08:22:17.002 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.004 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.008 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.010 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.014 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.015 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.016 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.018 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.020 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.021 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.023 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.026 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.027 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.031 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.032 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.033 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.035 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.038 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.039 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.040 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.045 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.046 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.047 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.048 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.049 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.053 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.054 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.055 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.055 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.056 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.060 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.061 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.062 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.065 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.066 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.067 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.068 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.068 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.069 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.070 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.075 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.076 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.077 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.078 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.079 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.079 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
2025-02-18 08:22:17.080 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare
```

```
!pip install pyngrok
from pyngrok import ngrok

ngrok.set_auth_token("2tAjmYGdSmTF7ZOoQUFFYK8o3iq_3B5YfDWqzgPR6b5MpoEzu")

tunnel = ngrok.connect(8000)
ngrok_url = tunnel.public_url
print(f"Public URL: {ngrok_url}")

import uvicorn
import threading

def run_fastapi():
    uvicorn.run(app=app, host="0.0.0.0", port=8000)

thread = threading.Thread(target=run_fastapi, daemon=True)
thread.start()
```

```
Requirement already satisfied: pyngrok in /usr/local/lib/python3.11/dist-packages (7.2.3)
Requirement already satisfied: PyYAML>=5.1 in /usr/local/lib/python3.11/dist-packages (from pyngrok) (6.0.2)
Public URL: https://9fc1-34-82-253-99.ngrok-free.app
```

```
%%writefile carbon_credits_app.py
```

```
Overwriting carbon_credits_app.py
```

```
!streamlit run /usr/local/lib/python3.11/dist-packages/colab_kernel_launcher.py carbon_credits_app.py
```

```

Collecting usage statistics. To deactivate, set browser.gatherUsageStats to false.


  You can now view your Streamlit app in your browser.

  Local URL: http://localhost:8501
  Network URL: http://172.28.0.12:8501
  External URL: http://34.82.253.99:8501
```