import discord  
from discord.ext import commands  
import sys  
import os  
import traceback  
import asyncio  
import requests  
  
intents = discord.Intents(members=True)  
client = discord.Client(intents=intents)  
client = discord.Client  
  
client = commands.Bot(command\_prefix=">")  
  
  
def get\_aqi(city):  
 url = "https://air-quality-by-api-ninjas.p.rapidapi.com/v1/airquality"  
 querystring = {"city": city}  
 headers = {  
 'x-rapidapi-host': "air-quality-by-api-ninjas.p.rapidapi.com",  
 'x-rapidapi-key': "6be9d4e810msh73e4a0ee5ddc0aep11de96jsn6e35aa8f1fbe"  
 }  
 response = requests.request("GET", url, headers=headers, params=querystring)  
 data = response.json()  
 if "error" in data.keys():  
 return "Couldn't find the place :("  
 else:  
 aqi = data['overall\_aqi']  
 return f"AQI in {city.upper()} is {aqi}"  
  
  
@client.event  
async def on\_ready():  
 print("Bot is ready")  
  
  
@client.command()  
async def about\_hariyali\_bot(ctx):  
 await ctx.send("I see a future where getting to work or to school or to the store does not have to cause pollution."  
 "\n\n Hello user I am Hariyali Bot which aims at predicting the air quality index and identifying potential land for afforestation."  
 "\nUse '>aqi\_help' to know more about Air Quality Index")  
  
  
  
@client.command()  
async def clear(ctx, amount=10):  
 await ctx.channel.purge(limit=amount)  
  
@client.command(name='aqi\_help', description='aqi\_command', pass\_context=True)  
async def send\_aqi\_commands(ctx):  
 embed = discord.Embed(  
 title=">aqi\_help: Air Quality Index",  
 description="The list of commands to find information about Air Quality Index",  
 color=discord.Color.blurple())  
 embed.add\_field(  
 name=">aqi\_brief",  
 value="Brief Information about the Air Quality",  
 inline=False)  
 embed.add\_field(  
 name=">aqi\_act",  
 value=  
 ' Resources about the effects of Air Pollution',  
 inline=False)  
  
 await ctx.send(ctx.message.channel, embed=embed)  
  
@client.command()  
async def aqi\_brief(ctx):  
 await ctx.send(  
 "The air quality index (AQI) is an index for reporting air quality on a daily basis. "  
 "It is a measure of how air pollution affects one's health within a short time period. "  
 "The purpose of the AQI is to help people know how the local air quality impacts their health. "  
 "The Environmental Protection Agency (EPA) calculates the AQI for five major air pollutants, "  
 "for which national air quality standards have been "  
 "established to safeguard public health."  
 "\n 1. Ground-level ozone"  
 "\n2. Particle pollution/particulate matter (PM2.5/pm 10)"  
 "\n3. Carbon Monoxide"  
 "\n4. Sulfur dioxide"  
 "\n5. Nitrogen dioxide"  
 "\n\n\n For more Information you can visit : https://en.wikipedia.org/wiki/Air\_quality\_index")  
  
@client.command()  
async def aqi\_act(ctx):  
  
 await ctx.send(  
 "Effects of Air Pollution:"  
 "\n\n 1.Smog and soot: These are the two most prevalent types of air pollution."  
 "Both come from cars and trucks, factories, power plants, incinerators, "  
 "engines, generally anything that combusts fossil fuels such as coal, gas, "  
 "or natural gas. "  
 "\n\n 2.Hazardous air pollutants:A number of air pollutants pose severe health risks "  
 "and can sometimes be fatal even in small amounts"  
 "Mercury attacks the central nervous system. In large amounts, lead can damage "  
 "children’s brains and kidneys, and even minimal exposure can affect children’s IQ and ability to learn"  
 "\n\n 3.Greenhouse gases: By trapping the earth’s heat in the atmosphere, greenhouse gases lead to warmer "  
 "temperatures, which in turn lead to the hallmarks of climate change: rising sea levels, more extreme weather, "  
 "heat-related deaths, and the increased transmission of infectious diseases"  
 "\n\n 4.Pollen and mold:Mold and allergens from trees, weeds, and grass are also carried in the air, are "  
 "exacerbated by climate change, and can be hazardous to health."  
 "\n\n\n\nAir pollution is now the world’s fourth-largest risk factor for early death"  
 "For more information on effects of air pollution "  
 "visit: https://www.nationalgeographic.org/encyclopedia/air-pollution/#:~:text=Long-term%20health%20effects%20from,air%20pollutants%20cause%20birth%20defects.")  
  
@client.command()  
async def how\_to\_be\_safe(ctx):  
 embed = discord.Embed()  
 embed.set\_image(url="https://www.epa.gov/sites/default/files/2014-09/aqiguidepm.png")  
 await ctx.send(embed=embed)  
  
  
@client.command()  
async def feedback(ctx):  
 await ctx.send(  
 "Hello all,"  
  
"\n\n\n\n\n I hope you are receiving this message in good Mental and Physical health."  
  
  
"We see a Future where getting to work or to school or to the store does not have to "  
 "cause pollution."  
"There are very harmful effects of pollution and if we dont eradicate the problem we have "  
 "created it will cause us the big trouble."  
"\n\n\n\n\n\n\n Team Hariyali is trying to take initiative to aware and empower people "  
 "as well to protect vulnerable communities. " \  
"We want to include all the genders, Youth communities and local stake holders in our "  
 "mission. " \  
"We have made a discord bot HARIYALI BOT which allow the people of our country to know "  
 "the AQI of their place and " \  
"the overall geography of the place through Hariyali’s afforestation land suggestion feature."  
"\n\n\n\n\n\n\n Help us by filling this survey form ( https://forms.gle/aLSm3rWZznURrwhi9 )"  
"\n\nThank You!")  
  
@client.command(name='afforestation', description='aqi\_command', pass\_context=True)  
async def send\_aqi\_commands(ctx):  
 embed = discord.Embed(  
 title=">afforestation\_city",  
 description="The list of commands to find information about last 10 years data of your city",  
 color=discord.Color.blurple())  
 embed.add\_field(  
 name=">planting\_trees",  
 value="Brief Information planting trees in your area",  
 inline=False)  
  
 await ctx.send(ctx.message.channel, embed=embed)  
  
@client.command()  
async def afforestation\_delhi(ctx):  
 embed = discord.Embed()  
 embed.set\_image(url="https://www.skymetweather.com/content/wp-content/uploads/2017/11/Delhi-AQI-Graph-600.jpg")  
 await ctx.send(embed=embed)  
 await ctx.send("As suggested by the research, delhi's air pollution is far worse during the winter than in summer. "  
 "Therefore, each and every person needs to plant trees around their houses, workplaces, parks etc. "  
 "Trees act as the earth’s purification system by absorbing airborne chemicals and releasing oxygen. "  
 "To tackle global air pollution, we need to halt deforestation and plant billions of trees. AQI over "  
 "100 is considered to be unhealthy and here, delhi's AQI ranges in its 300's that is too dangerous for "  
 "human health as well as environment. With COVID-19 around, people are at high risk ofgetting sick, "  
 "which results in their immunity being lowered. In such a high pollution area, they are bound to get "  
 "trapped in diseases very easily therefore, each and every person should try their best to help in purifying "  
 "air. One simple and best way is to plant a tree. One tree makes a thousand faces healthy and happy.")  
@client.command()  
async def afforestation\_chennai(ctx):  
 embed = discord.Embed()  
 embed.set\_image(url="https://lh3.googleusercontent.com/izNsQIDQVHfu8UNJjnA5KlMaOPzilT64e-jkePkLRm7HGL4ov4a3UAKHdNWfw7Dr\_fCItADBuwt4fM6D9VdHauTUstgDiq-XFJ3wsuVnvBCa1iZbHKp25FBiw\_0l5vsTRToLVD6g")  
 await ctx.send(embed=embed)  
 await ctx.send("As suggested by the research, chennai's air pollution is far worse during the winter than in summer. "  
 "Therefore, each and every person needs to plant trees around their houses, workplaces, parks etc. "  
 "Trees act as the earth’s purification system by absorbing airborne chemicals and releasing oxygen. "  
 "To tackle global air pollution, we need to halt deforestation and plant billions of trees. AQI over "  
 "100 is considered to be unhealthy and here, chennai's AQI ranges in its 300's that is too dangerous for "  
 "human health as well as environment. With COVID-19 around, people are at high risk of getting sick, "  
 "which results in their immunity being lowered. In such a high pollution area, they are bound to get "  
 "trapped in diseases very easily therefore, each and every person should try their best to help in purifying "  
 "air. One simple and best way is to plant a tree. One tree makes a thousand faces healthy and happy.")  
  
@client.command()  
async def afforestation\_bhopal(ctx):   
 embed = discord.Embed()  
 embed.set\_image(url="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQktM-tGj\_8goI8Nu7971S2wAnYqbAOLO-hiQ&usqp=CAU")  
 await ctx.send(embed=embed)  
 await ctx.send("As suggested by the research, bhopal's air pollution is far worse during the winter than in summer. "  
 "Therefore, each and every person needs to plant trees around their houses, workplaces, parks etc. "  
 "Trees act as the earth’s purification system by absorbing airborne chemicals and releasing oxygen. "  
 "To tackle global air pollution, we need to halt deforestation and plant billions of trees. AQI over "  
 "100 is considered to be unhealthy and here, bhopal's AQI ranges in its 300's that is too dangerous for "  
 "human health as well as environment. With COVID-19 around, people are at high risk of getting sick, "  
 "which results in their immunity being lowered. In such a high pollution area, they are bound to get "  
 "trapped in diseases very easily therefore, each and every person should try their best to help in purifying "  
 "air. One simple and best way is to plant a tree. One tree makes a thousand faces healthy and happy.")  
  
@client.command(name='planting\_trees')  
async def test(context: commands.Context):  
  
 # Example dataset here!  
 DATASET = (  
  
 ('Good', ' 0-50', '1 daily'),  
 ('Satisfactory', ' 51-100', '10 daily'),  
 ('Moderate', ' 101-200', '100 daily'),  
 ('Poor', ' 2011-300', '1000 daily'),  
 ('Very Poor', ' 300+400', '10,000 daily'),  
 ('Severe', ' 401-500', '100,000 daily')  
 )# Assumes we have sorted this!  
 s = ['AQI Level AQI Range No. of Tress']  
 # This needs to be adjusted based on expected range of values or calculated dynamically  
 for data in DATASET:  
 s.append(' '.join([str(item).center(6, ' ') for item in data]))  
 # Joining up scores into a line  
 d = '```'+'\n'.join(s) + '```'  
 # Joining all lines togethor!  
 embed = discord.Embed(title = 'AQI Ranges and No. of tress to be planted', description = d)  
 await context.send(embed = embed)  
  
  
  
  
@client.command(name='most\_polluted\_city')  
async def test(context: commands.Context):  
  
 # Example dataset here!  
 DATASET = (  
  
 (1, 'Pitampura', 606),  
 (2, 'Dadri', 503),  
 (3, 'Hisar', 494),  
 (4, 'Jind', 446),  
 (5, 'Karol Bagh', 441),  
 (6, 'Nanpara', 429),  
 (7, 'Loni', 424),  
 (8, 'Sitapur', 420),  
 (9, 'Unnao', 417),  
 (10, 'Bettiah', 400),  
 )# Assumes we have sorted this!  
 s = ['No. City Name AQI Value']  
 # This needs to be adjusted based on expected range of values or calculated dynamically  
 for data in DATASET:  
 s.append(' '.join([str(item).center(6, ' ') for item in data]))  
 # Joining up scores into a line  
 d = '```'+'\n'.join(s) + '```'  
 # Joining all lines togethor!  
 embed = discord.Embed(title = 'AQI Pollution Data', description = d)  
 await context.send(embed = embed)  
  
  
@client.command(name='most\_cleanest\_city')  
async def test(context: commands.Context):  
  
 # Example dataset here!  
 DATASET = (  
  
 (1, 'Sikanadarbad', 12),  
 (2, 'Medinipur', 15),  
 (3, 'Kattivakkam', 25),  
 (4, 'Shillong', 30),  
 (5, 'Barmer', 37),  
 (6, 'Chingelput', 38),  
 (7, 'Digboi', 53),  
 (8, 'Cuttack', 54),  
 (9, 'Naspur', 55),  
 (10, 'Baloda Bazar', 56),  
 )# Assumes we have sorted this!  
 s = ['No. City Name AQI Value']  
 # This needs to be adjusted based on expected range of values or calculated dynamically  
 for data in DATASET:  
 s.append(' '.join([str(item).center(6, ' ') for item in data]))  
 # Joining up scores into a line  
 d = '```'+'\n'.join(s) + '```'  
 # Joining all lines togethor!  
 embed = discord.Embed(title = 'AQI Pollution Data', description = d)  
 await context.send(embed = embed)  
  
@client.command(name='aqi\_calculation', description='aqi\_command', pass\_context=True)  
async def send\_aqi\_commands(ctx):  
 embed = discord.Embed(  
 title=">aqi\_calculation: Air Quality Index Calculation",  
 description="Overall AQI is"  
"calculated only if data are available for minimum three pollutants out of which one should"  
"necessarily be either PM2.5 or PM10"  
"\n\n The list of commands to find Air Quality Index and Pollution markers for the place you mention",  
 color=discord.Color.blurple())  
  
  
 embed.add\_field(  
 name=">AQI<name of the place>",  
 value=  
 'Send you the list of Pollution Detection Stations in the place you pass',  
 inline=False)  
  
 await ctx.send(ctx.message.channel, embed=embed)  
  
 @client.command(name=">aqi <latitude> <longitude>")  
 async def get\_aqi(ctx, lat=None, lon=None) :  
 data\_aqi = await set\_aqi(lat=lat, lon=lon)  
 embed = discord.Embed(  
 title="AQI data", color=discord.Color.dark\_blue(), inline=False)  
 aqi\_ = data\_aqi["data"]["aqi"]  
 place = data\_aqi["data"]["city"]["name"]  
 place\_url = data\_aqi["data"]["city"]["url"]  
 dominant\_pollutant = data\_aqi["data"]["dominentpol"]  
 timestamp = data\_aqi["data"]["time"]["s"]  
 data\_pollutants = data\_aqi["data"]["iaqi"]  
  
 # co\_ = data\_aqi["data"]["iaqi"]["co"]["v"]  
 # no2\_ = data\_aqi["data"]["iaqi"]["no2"]["v"]  
 # o3\_ = data\_aqi["data"]["iaqi"]["o3"]["v"]  
 # so2\_ = data\_aqi["data"]["iaqi"]["so2"]["v"]  
 # pm\_10 = data\_aqi["data"]["iaqi"]["pm10"]["v"]  
 # pm\_25 = data\_aqi["data"]["iaqi"]["pm25"]["v"]  
 pollutants = ''  
 for k, v in data\_pollutants.items() :  
 pollutants += f"{k} {v} \n"  
 embed.add\_field(  
 name=f"Place: {place} | AQI: {aqi\_}",  
 value=f"{place\_url}",  
 inline=False)  
 embed.add\_field(  
 name=f"Dominant Pollutant: ",  
 value=f"{dominant\_pollutant}",  
 inline=False)  
 embed.add\_field(name=f"Pollutants: ", value=pollutants, inline=False)  
 embed.add\_field(  
 name=f"Data Generated At: ", value=f"{timestamp}", inline=False)  
 embed.set\_image(url='https://airmega.com/wp-content/uploads/2016/01/1.png')  
 embed.set\_thumbnail(url='https://aqicn.org/air/experiments/images/aqi.png')  
 await ctx.send(embed=embed)  
  
 @client.command(name=">aqi\_stations <name of the place>")  
 async def get\_aqi\_stations(ctx, \*, name) :  
 nearby\_list = await set\_aqi(name=name)  
 if nearby\_list :  
 embed = discord.Embed(  
 title="Pollution Detection Stations in the city",  
 inline=False,  
 color=discord.Color.blurple())  
 for i in nearby\_list :  
 embed.add\_field(  
 name=f"Place: {i['name']} | AQI: {i['AQI']} ",  
 value=f"Coordinates: {i['geo\_loc']}",  
 inline=False)  
 embed.set\_image(  
 url=  
 'https://a.scpr.org/i/e88dad6837bf40214eb2234b461cfd10/136217-full.jpg'  
 )  
 embed.set\_thumbnail(  
 url='https://aqicn.org/air/experiments/images/aqi.png')  
 else :  
 embed = discord.Embed(  
 title="Pollution Detection Stations in the city",  
 description='Regret - No stations found in the place or near it!',  
 color=discord.Color.blurple(),  
 inline=False)  
 embed.set\_thumbnail(  
 url='https://aqicn.org/air/experiments/images/aqi.png')  
 await ctx.send(embed=embed)  
  
 async def set\_aqi(lat=None, lon=None, name=None) :  
 async with aiohttp.ClientSession() as session :  
 if lat is None and lon is None and name is None :  
 api\_key = "9299675506f06d45c06124fc2baf67b36d75a73d"  
 url = "https://api.waqi.info/feed/beijing/?token=9299675506f06d45c06124fc2baf67b36d75a73d" + "9299675506f06d45c06124fc2baf67b36d75a73d"  
 async with session.get(url) as resp :  
 aqi\_data = await resp.json()  
 return aqi\_data  
  
 elif name is not None and lat is None and lon is None :  
 api\_key = "9299675506f06d45c06124fc2baf67b36d75a73d"  
 url = "https://api.waqi.info/feed/beijing/?token=9299675506f06d45c06124fc2baf67b36d75a73d" + "9299675506f06d45c06124fc2baf67b36d75a73d" + "&keyword=" + name  
 async with session.get(url) as resp :  
 aqi\_data = await resp.json()  
 nearby\_list = []  
 for i in aqi\_data["data"] :  
 aqi = i["aqi"]  
 name = i["station"]["name"]  
 geo\_loc = i["station"]["geo"]  
 nearby\_list.append({  
 "name" : name,  
 "geo\_loc" : geo\_loc,  
 "AQI" : aqi  
 })  
 return nearby\_list  
 else :  
 api\_key = "9299675506f06d45c06124fc2baf67b36d75a73d"  
 url = "https://api.waqi.info/feed/beijing:" + lat + ";" + lon + "/?token=9299675506f06d45c06124fc2baf67b36d75a73d" + "9299675506f06d45c06124fc2baf67b36d75a73d"  
 async with session.get(url) as resp :  
 aqi\_data = await resp.json()  
 return aqi\_data  
  
  
client.run("OTAzOTIzNTQ5NjgzNzMyNTAw.YX0CaA.GGDLIlQPYLWBsjGPFJsnMDr7ySg")