```python
"""


4 - A Python script to download logs from Azure Resource group
"""


import os
import json
import requests
from azure.identity import DefaultAzureCredential


def get_azure_resource_group_activity_log_events(subscription_id: str,
                                                 resource_group:str,
                                                 start_time: str,
                                                 end_time: str)->json:
    """
    Returns an Azure resource group activity log.
    This code is adapted from the one found in the documentation.

    Arguments
    ---------
        subscription_id: Azure subscription id
        resource_group: The resource group name
        start_time: The start date for the log
        end_time: The end day for the log

    Returns
    -------
        return log_payload (json): The log event as a json object


    Documentation link
    ------------------
    https://learn.microsoft.com/en-us/azure/azure-monitor/essentials/rest-activity-log


    Usage:
    -----
    subscription_id = os.environ["AZURE_SUBSCRIPTION_ID"]
    resource_group = os.environ["RESOURCE_GROUP_NAME"]
    start_time = "2024-09-19T20:00:00Z"
    end_time = "2024-09-20T20:00:00Z"
    events = get_azure_resource_group_activity_log_events(subscription_id, resource_group, start_time, end_time)
    if events:
        print(events)
    """
    credential = DefaultAzureCredential()
    token = credential.get_token("https://management.azure.com/.default")
    access_token = token.token
    endpoint = f"https://management.azure.com/subscriptions/{subscription_id}/providers/microsoft.insights/eventtypes/management/values"
    params = {
        "api-version": "2015-04-01",
        "$filter": f"eventTimestamp ge '{start_time}' and eventTimestamp le '{end_time}' and resourceGroupName eq '{resource_group}'"
    }
    headers = {
        "Authorization": f"Bearer {access_token}"
    }
    try:
        response = requests.get(endpoint, headers=headers, params=params)
        if response.status_code == 200:
            log_payload = response.json()
            return log_payload
        else:
            print(f"Error: {response.status_code} - {response.text}")
            return None
    except Exception as e:
        print(e)



if __name__ =="__main__":
    subscription_id = os.environ["AZURE_SUBSCRIPTION_ID"]
    resource_group = os.environ["RESOURCE_GROUP_NAME"]
    start_time = "2024-09-19T20:00:00Z"
    end_time = "2024-09-20T20:00:00Z"
    events = get_azure_resource_group_activity_log_events(subscription_id, resource_group, start_time, end_time)
    if events:
        print(events)
```