# Anime Trivia Console App

# Anime Trivia -API: https://opentdb.com/api.php?amount=10&category=31

import requests

from pprint import pprint as pp # This module beautifies the data when printing and especially when we use dictionary.

# It makes them more readable from the user.

api = 'https://opentdb.com/api.php?amount=10&category=31'

response = requests.get(api) # Making a Get request (calling the API)

print(response.status\_code) # If the connection status code is 200, it means that the request worked!

data = response.json() # let's see the anime trivia data

pp(data) # Prints the data on the python console

# Using functions with return so that we create a console environment where user will read each question with the possible answers,

# and then input the answer they feel it's right!

# Function to get the data (Please see the complete answer below)

def get\_data(data):

questions = [] # Creating a list to store the questions from the api

for item in data['results']:

question = { # This block of code should show to the user the question and the possible right answers.

# Every time the append method adds each question and answers to the list we initiated.

'question': item['question'],

'options': item['incorrect\_answers'] + [item['correct\_answer']],

'correct\_answer': item['correct\_answer']

}

questions.append(question)

return questions

questions = get\_data(data)

# Function to interact with data

def console\_questions(questions):

correct = [] # Creating two empty lists so that we store all the correct and incorrect answers

# the user gives while interacting with the console.

incorrect = []

for question in questions:

print(question['question'])

for option in question['options']:

print(option)

answer = input('Your answer: ') # Using the in-built function for the user to interact with the console by giving an answer

if answer == question['correct\_answer']:

correct.append(question['question'])

else:

incorrect.append(question['question'])

return correct, incorrect

correct, incorrect = console\_questions(questions)

print('You had {} correct answers'.format(len(correct))) # Using in-built function that adds the number of the right answers in the list correct=[]

# and the number of the wrong answers in the incorrect = []

print('You had {} incorrect answers'.format(len(incorrect)))

# String Slicing to find the category that the questions fall in

line = "Entertainment: Japanese Anime &amp; Manga"

s\_line = line[35:] # Slices from 35 to the end of the string

print(s\_line)

# As the questions include some special characters which affects the understanding of the questions,

# I will do some cleaning and then write the questions into a file

# by adding only the "?" from the special characters at the end of the questions.

# Writes the questions in the text file "F\_Results"

with open('F\_Results.txt', 'w') as text\_file:

for item in data['results']: # Using a for loop

# Removing special characters from the questions

translator = str.maketrans('', '', string.punctuation)

s = item['question']

question = s.translate(translator)

if '?' in s:

question += '?' # Equivalent to question = question + '?'

correct\_answer = item['correct\_answer'].translate(translator)

incorrect\_answers = [answer.translate(translator) for answer in item['incorrect\_answers']]

# Writing the questions and answers to the file

text\_file.write(question + '\n') # Use of '\n' to start on a new line

text\_file.write(f"Correct Answer: {correct\_answer}\n")

text\_file.write("Incorrect Answers:\n")

for answer in incorrect\_answers:

text\_file.write(f"- {answer}\n") # Each incorrect question will appear with a "-" character and in a new line ('\n')

text\_file.write("\n")

# Reads the questions that are saved in the text file "F\_Results"

with open("F\_Results.txt", "r") as text\_file:

contents = text\_file.read()

print(contents)

# Import the datetime module where it shows the time and date to the user and

# adding it in the text file "F\_Results by using append mode"

from datetime import datetime

with open("F\_Results.txt", "a") as text\_file:

now = datetime.now()

text\_file.write(str(now)) # This will write the datetime.now() in the text file "F\_Results"

# str() in-built function converts an object to a string

print("The current date & time is the following: {}".format(now)) # This will appear in Python Console

import requests

import random

# Fetch data from the API

response = requests.get('https://opentdb.com/api.php?amount=10&category=31')

# Check if the request was successful

if response.status\_code != 200:

print("Failed to fetch questions")

else:

data = response.json()

def format\_questions(api\_data):

questions = []

for item in api\_data['results']:

question = {

'question': item['question'],

'options': item['incorrect\_answers'] + [item['correct\_answer']],

'correct\_answer': item['correct\_answer']

}

random.shuffle(question['options'])

questions.append(question)

return questions

def console\_questions(questions):

correct = []

incorrect = []

for question in questions:

print(question['question'])

for option in question['options']:

print(option)

answer = input('Your answer: ')

if answer == question['correct\_answer']:

correct.append(question['question'])

else:

incorrect.append(question['question'])

return correct, incorrect

questions = format\_questions(data)

correct, incorrect = console\_questions(questions)

print(f'You had {len(correct)} correct answers')

print(f'You had {len(incorrect)} incorrect answers')