Air University Multan campus Department of Computer Science and Engineering

**Random Wikipedia Article Generator**

# **Prepared by:** Group # 3

# **Group Members:**

# Kainat Zahra 183159

# Momina Iqbal 183173

# Muhammad Furqan Amjad 183205

# Muhammad Anas Malik 183183

Sixth Semester

Spring 2021

**TABLE OF CONTENTS**

1. [Introduction………………………………………………………………………………......](#_gjdgxs) 3
2. The Problem………………………………………………………………………………….3
3. [Specifications](#_3znysh7)…………………………………………………………………………………….3
4. Process flow………………………………………………………………………………….3
5. Code…………………………………………………………………………………………..4
6. Output…………………………………………………………………………………………7
7. Conclusion……………………………………………………………………………………8

8. [References……………………………………………………………………………………](#_1fob9te) 8

#### **INTRODUCTION**

The project Wikipedia article fetches random articles from Wikipedia and user can then read the summary of articles, get new articles or open the complete articles in the web browser. The user can read as many articles as he wants of his choice until and unless he/she presses the quit option.

#### **THE PROBLEM**

The project is used to get a random article from Wikipedia and then ask the user if he wants to read the article or not. If the answer is yes, then the article is shown on the screen otherwise another random article is fetched from Wikipedia.

1. **Specifications**

In this random article generator, we used ‘pip install Wikipedia’ command to install Wikipedia API. Then we used a library ‘import Wikipedia’ library to get access to Wikipedia webpage.

In Wikipedia and other sites running on Media-Wiki, we used wikipedia.random ( ) can be used to access a random article in the main namespace; this feature is useful as a tool to generate a random article. Similarly we used wikipedia.page( ) to view a random article webpage.

### Process flow

1. First of all, the program chooses a random Wikipedia article.
2. The application asks the user if he/she wants to read the article or not.
3. If he/she enters yes, the app displays a brief summary about the article.
4. If the user enters no, the program displays another random article and asks the user if he/she wants to read it or not and continues this process until the user gets an article of his choice or he/she enters the quit option.
5. After the summary is displayed, the program asks the user if he/she wants to open the complete article in the web browser.
6. If the user enters yes, the control moves to the web browser and the complete article is displayed there.
7. It the user enters no the program asks the user if he/she wants to read about another article or not.
8. The user can read as many articles as he wants by pressing a yes.
9. There is a quit option with yes/no at each step in the program. The program terminates as the user enters quit.

### Code

### After starting a project and staring an app named Wikipedia article, we added the app name in the settings.py file of our project in the installed apps.

### 

### After that we included the URL of the app into the urls.py file of the project. The code after adding the required URL is shown below:

### 

### We added the python code in views.py file of the app. we defined the function getpage() in the code. This function is responsible for all the functionalities being performed in the process flow. In the following code we used different functions, loops, conditions, print statements and libraries to ensure the required output.

### 

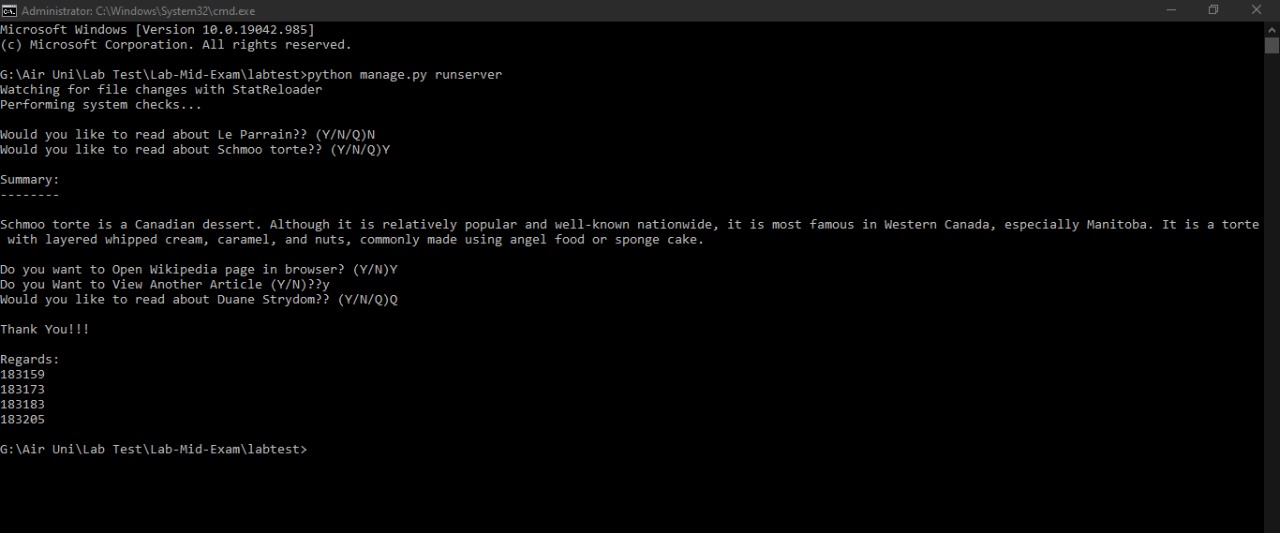
### 

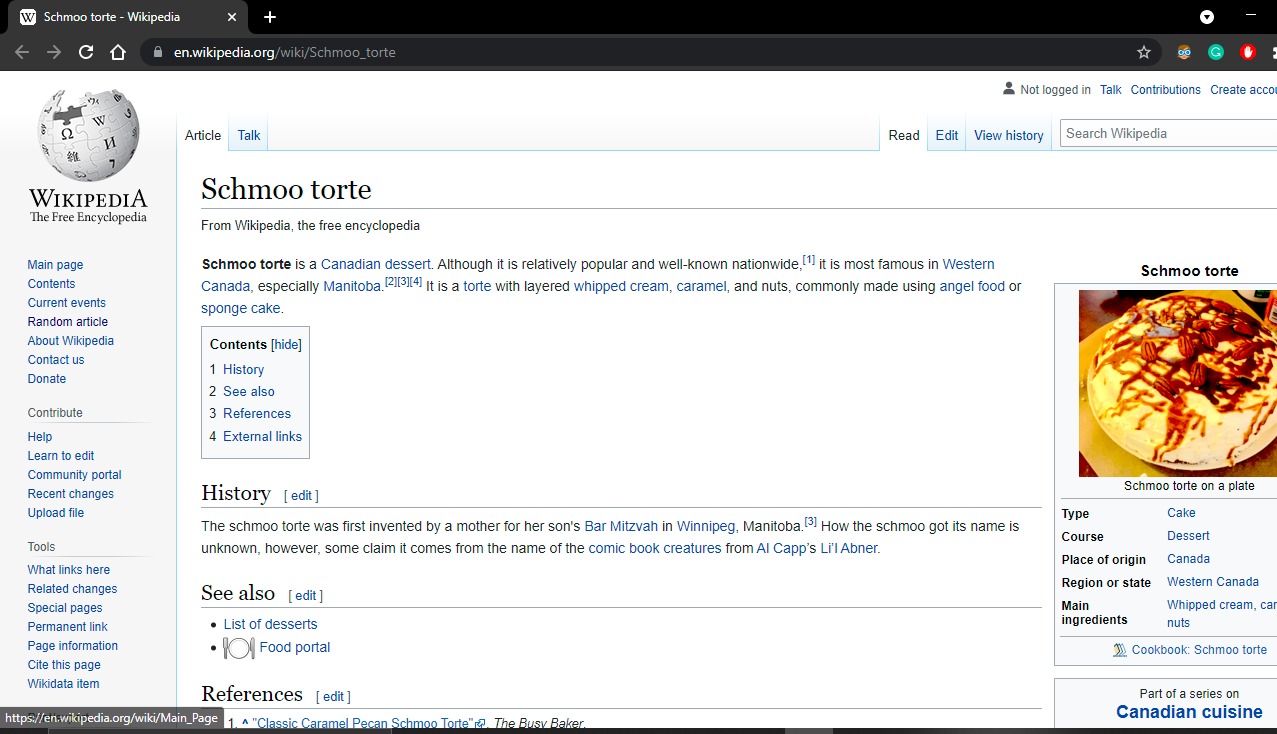
### 

### At last we called the defined function getpage() in urls.py file of the app. This is done by defining the path for function in urlpatterns.

### 

1. **Output**

****

****

### Conclusion

Wikipedia is an online free-content encyclopedia project helping to create a world in which everyone can freely share in the sum of all knowledge. Basically Wikipedia Article is a random article generator, In which a user can read an article or change the article according to his interest.

### REFERENCES

<https://www.w3schools.com/> <https://docs.python.org/3/tutorial/>

<https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django> <https://www.scribd.com/doc/61315817/Intra-Mailing-System> <http://www.ijesrt.com/issues%20pdf%20file/Archive-2017/April-2017/18.pdf> <https://www.academia.edu/36614854/Spam_Filtering_Using_ML_Algorithms>