**CREATE CHATEBOT IN PYTHON**

TEAM MEMBERS

1. SUYAMBULAKSHMI V
2. SHARINITHA R
3. AYESHASITHIKA S
4. PIRITHIBA M

**PHASE – 1 Document Submission**

**OBJECTIVE:**

* The Objective of this project is to create a high-quality support to users, ensuring a positive user experience and customer satisfaction chatbot in Python that provides exceptional customer service, answering user queries on a website or application.

**PROGRAM:**

import json

import string

import random

import nltk

import numpy as num

from nltk.stem import WordNetLemmatizer # It has the ability to lemmatize.

import tensorflow as tensorF # A multidimensional array of elements is represented by this symbol.

from tensorflow.keras import Sequential # Sequential groups a linear stack of layers into a tf.keras.Model

from tensorflow.keras.layers import Dense, Dropout

nltk.download("punkt")# required package for tokenization

nltk.download("wordnet")# word database

import re

import long\_responses as long

def message\_probability(user\_message, recognised\_words, single\_response=False, required\_words=[]):

  message\_certainty = 0

  has\_required\_words = True

  #counts how many words are present in each predefined message

  for word in user\_message:

    if word in recognised\_words:

      message\_certainty +=1

  #calculates the percent of recognised words in a user message

  percentage = float (message\_certainty) / float(len(recognised\_words))

  #checks that the required words are in the string

  for word in required\_words:

    if word not in user\_message:

      has\_required\_words = False

      break

  if has\_required\_words or single\_response:

    return int(percentage+100)

  else:

    return 0

def check\_all\_messages(message):

  highest\_prob\_list = {}

  def response(bot\_response, list\_of\_words, single\_response=False, required\_words=[]):

    nonlocal highest\_prob\_list

    highest\_prob\_list[bot\_response] = message\_probability(message, list\_of\_words, single\_response, required\_words)

    #response----------------------------------------

    response("Hello!",["hello","hi","sup","hey","heyo","whatsup"], single\_response=True)

    response('I\'m doing fine, and You?',["how","can","i","help","you"], required\_words=['how'])

    response("Thank You!",['i','love','doing','online','courses'], required\_words=['online','palace'])

    response(long.R\_EATING, ['what','you','eat'], required\_words=['you','eat'])

    best\_match = max(highest\_prob\_list, key=highest\_prob\_list.get)

    #print(highest\_prob\_list)

    return long.unknown() if highest\_prob\_list[best\_match] < 1 else best\_match

def get\_response(user\_input):

  split\_message = re.split(r'\s+|[,;?!.-]\s\*', user\_input.lower())

  response = check\_all\_messages(split\_message)

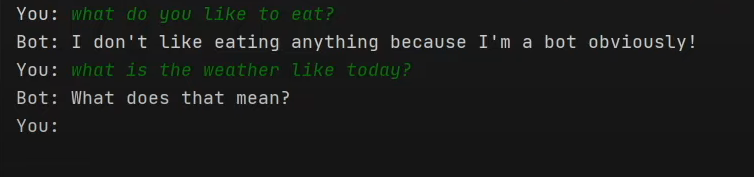
  return response

while True:

  print('Bot: ' + get\_response(input('You: ')))

**SAMPLE OUTPUT:**

Output is taken for our sample chatbot project in VISUAL STUDIO CODE,

****

**FUNTIONALITY:**

* Here we have used some libraries to create chatbot using NLP.
* They are,

1. JSON: It is possible to utilize it to work with JSON data.
2. String: Provides access to several potentially valuable constants.
3. Random: For various distributions, this module implements pseudo-random number generators.
4. WordNetLemmatizer: It can lemmatize.
5. Tensorflow: A multidimensional array of elements is represented by this symbol.
6. Sequential: Sequential groups a linear stack of layers into a tf.keras.Model.

**USER-INTERFACE:**

* This chatbot will be integrated in website using some basic skills (HTML, CSS, JAVASCRIPT).

**NATURAL LANGUAGE PROCESSING(NLP):**

* An natural language processing chatbot is a software program **that can understand and respond to human speech**. Bots powered by NLP allow people to communicate with computers in a way that feels natural and human-like — mimicking person-to-person conversations.

**RESPONSES:**

* By using NLP ,This chatbot can understand and respond naturally  such as accurate answers, suggestions, and assistance.

**INTEGRATION:**

* Here ,We have developed a sample website using HTML to integrate the chatbot .

**WEBSITE LINK:** [**https://technerdscentre.neocities.org/chatbot/**](https://technerdscentre.neocities.org/chatbot/)

**TESTING AND IMPROVEMENT:**

* Testing and Improvement can be done by using **VISUAL STUDIO CODE** or **KAGGLE NOTEBOOK**

**DATASET LINK**[**:** **https://www.kaggle.com/datasets/grafstor/simple-dialogs-for-chatbot**](file:///C:\Users\suyambu%20lakshmi\Downloads\%20https\www.kaggle.com\datasets\grafstor\simple-dialogs-for-chatbot)