**CREATE A CHATBOT IN PYTHON**

TEAM MEMBERS

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

**Phase 2 – 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:**

* This is the sample 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: ')))

* Here mentioned program is called in the above program.

import random

R\_EATING = "I don't like eating anything because I'm a bot obviously!"

def unknown():

    response = ['Could you please re-phase that?',

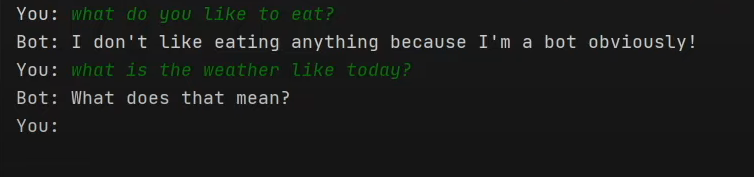
                "...",

                "sounds about right",

                "what does that mean?"][random.randrange(4)]

    return response

**OUTPUT:**



* **STEP 1:** We have developed a website in order to implement the chatbot. This is our website link - <https://technerdscentre.neocities.org/chatbot/> . This website is created by using the basic knowledge of HTML, CSS, JavaScript. and it is published with the help of [neocities](https://neocities.org/) website.
* **STEP 2:** By learning some important features of **Machine Learning,** For Example Data wrangling, neural networks, natural language processing,etc. And also some of the basic libraries we have installed the **Visual Studio Code** , They are, JSON,string, random, nltk, pytorch, flask app, request.
* **STEP 3:**  We are using Visual Studio code plateform to implement this code
* **STEP 4:** With the help of flask app and javascript , we are able to connect the python chatbot in our website.
* **STEP 5:** Here we will test our project