**NAME:M.SHARMIKA**

**DEPT/YEAR:CSE/III**

**CLG:2129-ST.JOSEPH COLLEGE OF ENGINEERING**

**REG N0:212921104048**

**PHASE 4:CHATBOT USING PYTHON**

**How To Make A chatbot In Python?**

Now we are going to build the chatbot using Python but first, let us see the file structure and the type of files we will be creating:

* **Intents.json** – The data file which has predefined patterns and responses.
* **train\_chatbot.py** – In this Python file, we wrote a script to build the model and train our chatbot.
* **Words.pkl** – This is a pickle file in which we store the words Python object that contains a list of our vocabulary.
* **Classes.pkl** – The classes pickle file contains the list of categories.
* **Chatbot\_model.h5** – This is the trained model that contains information about the model and has weights of the neurons.
* **Chatgui.py** – This is the Python script in which we implemented GUI for our chatbot. Users can easily interact with the bot.

**Here are the 5 steps to create a chatbot in Python from scratch**:

1. **Import and load the data file**
2. **Preprocess data**
3. **Create training and testing data**
4. **Build the model**
5. **Predict the response**

1. **Import and load the data file**

First, make a file name as train\_chatbot.py. We import the necessary packages for our chatbot and initialize the variables we will use in our Python project.

The data file is in JSON format so we used the json package to parse the JSON file into Python.

import nltk

from nltk.stem import WordNetLemmatizer

lemmatizer = WordNetLemmatizer()

import json

import pickle

import numpy as np

from keras.models import Sequential

from keras.layers import Dense, Activation, Dropout

from keras.optimizers import SGD

from tensorflow.keras.optimizers import SGD

import random

words=[]

classes = []

documents = []

ignore\_words = ['?', '!']

data\_file = open('intents.json').read()

intents = json.loads(data\_file)

2. **Preprocess data**

When working with text data, we need to perform various preprocessing on the data before we make a machine learning or a deep learning model. Based on the requirements we need to apply various operations to preprocess the data.

Tokenizing is the most basic and first thing you can do on text data. Tokenizing is the process of breaking the whole text into small parts like words.

for intent in intents['intents']:  
for pattern in intent['patterns']:  
#tokenize each word  
w = nltk.word\_tokenize(pattern)  
words.extend(w)  
#add documents in the corpus  
documents.append((w, intent['tag']))  
# add to our classes list  
if intent['tag'] not in classes:  
classes.append(intent['tag'])

Now we will lemmatize each word and remove duplicate words from the list. Lemmatizing is the process of converting a word into its lemma form and then creating a pickle file to store the Python objects which we will use while predicting.