

TASK ASSIGNMENT-2

TASK:

Building a Chabot using python:

Building a Chabot Using python, in this you will explore various libraries and techniques to develop a responsive and interactive Chabot that can engage with users effectively. These tasks have been designed to provide you with hands-on experience in python programming and GUI Development.

INTRODUCTION:

All of you will be familiar with Chabot. Today we will learn about how to design chatbots in python. Making chat bots are very amazing. So welcome in **Python Chabot Tutorial**. So let's start without wasting time.

Now-a-days various companies, industries or individuals are using chatbots. Chatbots are very helpful tool for today's business world. They are providing great business opportunities for small and large scale industries. It reduces the response time and increases the availability of services. So now the question is- **what are chat bots, how they work and why we use them?**

CHATBOTS:

A chatbot is a service, powered by rules and sometimes artificial intelligence that you interact with via a chat interface.

Chatbots are softwares agents that converse trough a chat interface that means the softwares programs that are able to have a conversation which provides some kinds of value to the end users.

The user can interact with the chatbot by typing in their end of the conversation or simply by voice depending upon the type of chatbot is provided.

Chatbots are revolutionizing the way businesses interact with their clients. Using AI and sophisticated natural language processing, modern chatbots offer a deeper level of interaction than ever before.

SOME EXAMPLES OF CHATBOTS

- Liveperson
- LiveChat
- Amazon Lex
- Dialogflow
- IBM Watson
- bold360

WHAT CAN CHATBOTS DO

- Deliver personalized content experiences.
- Answer common customer service questions.
- Streamline product purchases.
- Cultivate connection via entertainments.
- Offer specialized services.

HOW CAN CHATBOTS IMPROVE BUSINESS:

Chatbots can help the business in many ways, like –

- Drive sales
- Immediate customer services
- Lower Acquisition costs
- Automation+Efficient
- Better Interaction with consumers
- Gather data faster

SOME USE CASES OF CHATBOTS

- YouTube celebrity/Artists
- Restaurants/cafe
- Ecommerce

- Newsletter
- Business card
- Customer service
- Quiz
- Campaign etc

Now let us know about the code and this was prepared without any libraries and a simple programming. Here we are creating the chat bot on creating and importing the modules. Here we have used “long_responses” module.

Below shown is the code for chat bot

COMPLETE CODE:

```
import re
import long_responses

def
message_probability(user_message, recognised_words, single_response=False, required_words=[]):
    message_certainty=0

    has_required_words=True
    for word in user_message:
        if word in recognised_words:has_required_words
            message_certainty+=1

    percentage=float(message_certainty)/float(len(recognised_words))

    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_wor
ds=[]):
        nonlocal highest_prob_list
        highest_prob_list[bot_response]=message_probability(message,
list_of_words, single_response, required_words)
        #response-----

response('hello!',['helloo','hi','sup','hey','heyo'],single_response=T
rue)
    response('I\'m doing fine, and you?',['how','are','you','doing'],
required_words=['how'])
    response('thank
you!',['i','love','code','palace'],required_words=['code','palace'])

    response(long_responses.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_responses.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())
    return get_response

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:')))

```

As we have already discussed about the module we have imported in the above code

The below is the internal code written in the module

```

import random

R_EATING = "I dont like eating anything because I'm a bot obviously."
def unknown():
    response = ['could you please re-phrase that?',
                "...",
                "sounds about right",
                "What does that mean?"][random.randrange(44)]
    return response

```

OUTPUT:

```
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\user> & "C:/Program Files/Python312/python.exe" "c:/Users/user/Desktop/internship/task 2/bot.py"
You:hi
Bot:hello!
You:how are you doing
Bot:I'm doing fine, and you?
You:i love code palace
Bot:thank you!
You:what do you eat
Bot:I dont like eating anything because I'm a bot obviously.
You:
```

CONCLUSION:

In this way the chat bot is created using a simple python code without using any libraries. The responses are given by the chatbot by selecting the highest probability in the given words that is given by the user.