

1. Please describe the architecture of the current chatbot system. Identify the components and check where they are running now.

Ans:

use telegram to show our chatbot.

use hkgupt to understand and answer their questions.

Use redis to store these information

2. Explain how do your chatbot handle the special command. You need to trace the code and explain that.

Ans:

Original code is : `chatgpt_handler = MessageHandler(Filters.text & (~Filters.command), equipped_chatgpt)`

New code is : `chatgpt_handler = MessageHandler(Filters.text, equipped_chatgpt)`

`Filters.text & (~Filters.command)` means we will handle those messages users typed into but we won't handle the command.

Ps: command in telegram starts with /, so the easiest way to settle this problem is delete the second part.

This is a chatbot, I don't think it need to deal with these order so I simply delete it .

3. Update your code so that when user type /hello Kevin , it will reply Good day, Kevin! . Write down the change you have made.

Ans:

Original code:

```
def equipped_chatgpt(update, context):
    global chatgpt
    reply_message = chatgpt.submit(update.message.text)
    print(update)
    logging.info("Update: " + str(update))
    logging.info("context: " + str(context))
    context.bot.send_message(chat_id=update.effective_chat.id,
                             text=reply_message)
```

New code:

```
def equipped_chatgpt(update, context):
    global chatgpt
    reply_message = chatgpt.submit(update.message.text)
```

```

print(update)
if update.message.text=='/hello Kevin':
    reply_message='Good day, Kevin!'
logging.info("Update: " + str(update))
logging.info("context: " + str(context))
context.bot.send_message(chat_id=update.effective_chat.id,
text=reply_message)

```

I catch the text inputted in telegram and change the output

4. Make a few screen caps to prove that you have applied your own Redis account, used it in your chatbot, and push the code on GitHub (at least 2 commits - lab3/lab4)

Ans:

I have used redis.

The screenshot shows a code editor with four tabs: main.py, chatbot.py, test.py, and ChatGPT\_HKBU.py. The main.py file contains the following Python code:

```

r = redis.Redis(
    host='redis-15051.c292.ap-southeast-1-1.ec2.cloud.redislabs.com',
    port=15051,
    password='Ymb9U20meJ3czx5eG6XFZ9WH7rT0oMOB')

keys = r.keys()
print("Keys:")
for key in keys:
    print(key.decode())

# 获取指定键的值
key = 'your_key'
value = r.get(key)
print("here:", r.get("{www:we}"))
print(f"Value of key '{key}': {value.decode() if value else None}")

```

Below the code editor is a terminal window with the following output:

```

C:\Users\10851\AppData\Local\Programs\Python\Python310\python.exe D:\self_study\hkbu\S2\7940-cloud\code\main.py
Keys:
www:we
dad
qq
("Ty", "11")
{www:we}
www.
here: b'1'
Value of key 'your_key': None

```

Submit my code to github

Redis Clou... HKBU Chi... bilibili - 翻... 哔哩哔哩... git-哔哩哔哩... 查看本批... DeepL翻... github - 翻... History f... cmd切换... +

https://github.com/xsm-bit/chatbot-repo/commits/main/chatbot.py

github.com / xsm-bit / chatbot-repo

Type / to search

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

## Commits

History for chatbot-repo / chatbot.py on main

Commits on Feb 17, 2024

second commit

Eric Xiao committed now

1f87bdd

<>

first commit

Eric Xiao committed 10 minutes ago

dd682c9

<>

End of commit history for this file