PA1

socket_server

TODO_1 : calculate_expression

可以執行 + - * / 和 **(平方) %(取餘數) //(取商) sqrt(開根號) abs(取絕對值)

TODO_2 : 設定host 和 port

TODO_3 : 創建和綁定socket

TODO_4 : 監聽請求

TODO_5 : 接收新請求並同意連接

TODO_6 : 接收客戶端數據

TODO_7 : 傳計算結果給客戶端

client_server

TODO_1 : 設定server IP 和 port 並創建socket 連結到server

TODO_2 : 傳東東給server

TODO_3 : 接收server傳回的

TODO_4 : 發問題給server

TODO_5 : 從server拿答案

Result

tastcase_b0 : 加減乘除

tastcase_b1 : 次方

tastcase_b2 : 取mod

tastcase_b3 : 取商

tastcase_b4 : 開根號

tastcase_b5 : 取絕對值

PA2

TODO_1 : 在localhost:2040設定socket並監聽請求和取得client資訊

● 和PA1 server一樣

TODO_2 : 從請求中取出所需檔案

● 如果沒有指定 Default to index.html

● 如果有，要記得去掉開頭的 " / "

TODO_3: 把所求傳回瀏覽器

● 記得要/n/n來區別 真的內容

● 記得要utf-8去解析

TODO_4: 如果找不到那就回傳404

TODO_5: 簡單建立html

● index.html   helloword.html

Result

result_1 : access HTML files through a local machine.

result_2 : show 404 NOT FOUND on the browser

result_3 : access helloworld.html through index.html

PA3

PA1

socket_server

TODO_1 : calculate_expression

可以執行 + - * / 和 **(平方) %(取餘數) //(取商) sqrt(開根號) abs(取絕對值)

```python
def calculate_expression(expression):
    operands = expression.split()
    if len(operands) == 3:
        num1, operator, num2 = operands
        num1 = float(num1)
        num2 = float(num2)

        if operator == '+':
            return str(num1 + num2)
        elif operator == '-':
            return str(num1 - num2)
        elif operator == '*':
            return str(num1 * num2)
        elif operator == '/':
            if num2 == 0:
                return "Error: Division by zero."
            return str(num1 / num2)
        elif operator == '**':
            return str(num1 ** num2)
        elif operator == '%':
            return str(num1 % num2)
        elif operator == '//':
            return str(num1 // num2)
        else:
            return "Error: Invalid operator."
    elif len(operands) == 2:
        operator, num = operands
        num = float(num)
```

先把輸入依據空格分開 (*重點* 輸入必須有空格 像 1 + 1)

判斷被空白區分的長度
- 如果是 3 代表是 + - * / ** // % (1 ** 1 分三段)
- 如果是 2 代表是 sqrt abs (sqrt 2 分兩段)

接者處理錯誤
- 像是除以0
- 對負數開根號
- 分出來長度不是3 或 2
- 沒有定義的oprerater

利用py內建的功能達成 (沒有import math)

TODO_2 : 設定host 和 port

```python
# TODO Start
HOST, PORT = "127.0.0.1", 2040
# TODO end
```

TODO_3 : 創建和綁定socket

```python
# TODO Start
serverSocket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
serverSocket.bind((HOST, PORT))
# TODO End
```

TODO_4 : 監聽請求

```python
# TODO Start
serverSocket.listen()
# TODO End
```

TODO_5：接收新請求並同意連接

```python
# TODO Start
client, address = serverSocket.accept()
# TODO End
```

TODO_6：接收客戶端數據

```python
# TODO Start
question = client.recv(1024).decode('utf-8')
# TODO End
```

TODO_7：傳計算結果給客戶端

```python
# TODO Start
client.sendall(message.encode('utf-8'))
# TODO End
```

client_server

TODO_1：設定server IP 和 port 並創建socket 連結到server

```python
# TODO Start
if select_server == 'TA':
    HOST, PORT = '140.112.42.104' , 7777
else:
    HOST, PORT = '127.0.0.1', 2040
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect((HOST, PORT))
# TODO End
```

TODO_2：傳東東給server

```python
# TODO Start
s.sendall(response.encode('utf-8'))
# TODO End
```

TODO_3：接收server傳回的

```python
# TODO Start
server_message = s.recv(1024).decode('utf-8')
# TODO End
```

TODO_4：發問題給server

```python
# TODO Start
s.sendall(question.encode('utf-8'))
# TODO End
```

TODO_5 : 從server拿答案

```
# TODO Start
ans = s.recv(1024).decode('utf-8')
# TODO End
```

Result

tastcase_b0：加減乘除

```
The Client is running..
Question: 1 + 1
Received the message from server:
Please input a question for calculation
Get the answer from server:
2.0
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: 2 - 4
Received the message from server:
Please input a question for calculation
Get the answer from server:
-2.0
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: 3 * 5
Received the message from server:
Please input a question for calculation
Get the answer from server:
15.0
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: 5 / 0
Received the message from server:
Please input a question for calculation
Get the answer from server:
Error: Division by zero.
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: 4 / 2
Received the message from server:
Please input a question for calculation
```

tastcase_b1： 次方

```
The Client is running..
Question: 2 ** 3
Received the message from server:
Please input a question for calculation
Get the answer from server:
8.0
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: 10.0 ** 3.1
Received the message from server:
Please input a question for calculation
Get the answer from server:
1258.9254117941675
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: 0 ** 10
Received the message from server:
Please input a question for calculation
Get the answer from server:
0.0
```

tastcase_b2： 取mod

```
The Client is running..
Question: 10 % 3
Received the message from server:
Please input a question for calculation
Get the answer from server:
1.0
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: 3 % 10
Received the message from server:
Please input a question for calculation
Get the answer from server:
3.0
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: 0.1 % 3
Received the message from server:
Please input a question for calculation
Get the answer from server:
0.1
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: 20 % 3.5
Received the message from server:
Please input a question for calculation
Get the answer from server:
```

tastcase_b3： 取商

```
The Client is running..
Question: 9 // 2
Received the message from server:
Please input a question for calculation
Get the answer from server:
4.0
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: 3 // 10
Received the message from server:
Please input a question for calculation
Get the answer from server:
0.0
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: 10 // 3.5
Received the message from server:
Please input a question for calculation
Get the answer from server:
2.0
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: 100 // 2
Received the message from server:
Please input a question for calculation
Get the answer from server:
50.0
```

tastcase_b4： 開根號

```
The Client is running..
Question: sqrt 16
Received the message from server:
Please input a question for calculation
Get the answer from server:
4.0
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: sqrt 1024
Received the message from server:
Please input a question for calculation
Get the answer from server:
32.0
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: sqrt 2
Received the message from server:
Please input a question for calculation
Get the answer from server:
1.4142135623730951
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: sqrt -2
Received the message from server:
Please input a question for calculation
Get the answer from server:
Error: Negative number cannot be square rooted.
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: sqrt -999
Received the message from server:
Please input a question for calculation
```

tastcase_b5： 取絕對值

```
The Client is running..
Question: abs 16
Received the message from server:
Please input a question for calculation
Get the answer from server:
16.0
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: abs -16
Received the message from server:
Please input a question for calculation
Get the answer from server:
16.0
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: abs 2.2
Received the message from server:
Please input a question for calculation
Get the answer from server:
2.2
Do you wish to continue? (Y/N)
Response to server prompt: Y
Question: abs -2.2
Received the message from server:
Please input a question for calculation
Get the answer from server:
2.2
```

PA2

TODO_1 : 在localhost:2040設定socket並監聽請求和取得client資訊

- 和PA1 server一樣

TODO_2 : 從請求中取出所需檔案

```
# TODO Start
filename = message.split()[1]
if filename == "/":
    filename = "index.html"  #
else:
    filename = filename[1:]  #
# TODO End
```

- 如果沒有指定 Default to index.html
- 如果有, 要記得去掉開頭的 " / "

TODO_3: 把所求傳回瀏覽器

```
# TODO Start
header = 'HTTP/1.1 200 OK\nContent-Type: text/html\n\n'
connectionSocket.send(header.encode("utf-8"))
connectionSocket.send(outputdata.encode('utf-8'))
connectionSocket.close()
# TODO End
```

- 記得要/n/n來區別 真的內容
- 記得要utf-8去解析

TODO_4: 如果找不到那就回傳404

```
# TODO Start
response = "HTTP/1.1 404 Not Found\n\n404 Not Found"
connectionSocket.send(response.encode())
connectionSocket.close()
# TODO End
```

TODO_5: 簡單建立html

- index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>HaHa~</title>
</head>
<body>
    <h1>My name is Dang Xu Chen and my student ID is b09901017</h1>
    <p>Click ME <a href="helloworld.html"> helloworld is here</a> </p>
</body>
</html>
```

helloword.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>hello world</title>
</head>
<body>
    <h1>Hello World!</h1>
    <p>Back to ~ <a href="index.html">index page</a>.</p>
</body>
</html>
```
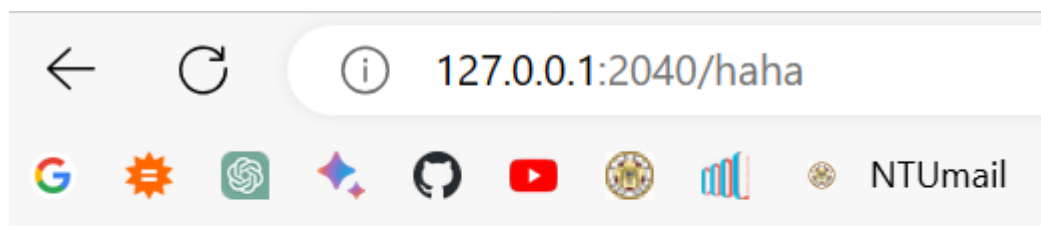
- 利用 <a> 建立超連結 可以跳轉到 helloword.html

Result

result_1 : access HTML files through a local machine.

```
Ready to serve...
('127.0.0.1', 51295) connected
client's request message:
 GET /index.html HTTP/1.1
Host: 127.0.0.1:2040
Connection: keep-alive
sec-ch-ua: "Microsoft Edge";v="123", "Not:A-Brand";v="8", "Chromium";v="123"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Windows"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/123.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,ap
=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: http://127.0.0.1:2040/helloworld.html
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: zh-TW,zh;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6


Extract the filename: index.html
```

result_2 : show 404 NOT FOUND on the browser


```
404 Not Found
```

result_3 : access helloworld.html through index.html

```
Ready to serve...
('127.0.0.1', 51296) connected
client's request message:
 GET /helloworld.html HTTP/1.1
Host: 127.0.0.1:2040
Connection: keep-alive
sec-ch-ua: "Microsoft Edge";v="123", "Not:A-Brand";v="8", "Chromium";v="123"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Windows"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHT
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,imag
=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: http://127.0.0.1:2040/index.html
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: zh-TW,zh;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6


Extract the filename: helloworld.html
```

PA3

Proxy Server TODO

TODO_1: 建立socket 等前置動作 (localhost:9999)

和前面一樣

TODO_2: check cache 裡面有沒有

```python
with open(file_path[1:], "r") as cache_file:
    output_data = cache_file.readlines()
file_exist = "true"
```

TODO_3: 如果沒有就創新socket去和server拿，然後寫入cache

創新socket

```python
proxy_server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
```

連線

```python
proxy_server_socket.connect((host_name, 2040))
```

請求

```python
request_message = f"GET {file_path} HTTP/1.1\r\n"
print(request_message)
file_obj.write(request_message)  # Write the request to the file-like object
file_obj.flush()
```

讀回傳值

```python
buffer = file_obj.read()
```

寫入cache

```python
with open(file_path[1:], "w", encoding='utf-8') as cache_file:
    cache_file.write(buffer)
```

然後回傳給client

```python
for line in buffer :
    client_socket.send(line.encode('utf-8'))
```

TODO_4: 如果有 就直接從 cache拿給client

直接給client

```python
for line in output_data:
    client_socket.send(line.encode('utf-8'))
```

TODO_5: 如果都沒有 就回傳404

```python
client_socket.send("HTTP/1.1 404 NOT FOUND\r\n".encode('utf-8'))
client_socket.send("<html><body><h1>404 Not Found</h1></body></html>".encode('utf-8'))
```

Result

Access index.html

```
index.html
/index.html
Host name is 127.0.0.1
Trying to connect to the web server
Connected successfully
GET /index.html HTTP/1.1

Sent the request to the web server successfully
Read the file from the web server successfully
Wrote the file to the cache successfully
Sent the data from the web server to the client
Ready to serve...
Received a connection from: ('127.0.0.1', 51551)
```

Access helloworld.html

```
helloworld.html
/helloworld.html
Host name is 127.0.0.1
Trying to connect to the web server
Connected successfully
GET /helloworld.html HTTP/1.1

Sent the request to the web server successfully
Read the file from the web server successfully
Wrote the file to the cache successfully
Sent the data from the web server to the client
Ready to serve...
Received a connection from: ('127.0.0.1', 51553)
```

Access a non-existing testnotfound.html

127.0.0.1:9999/notfound_test

```
404 Not Found
```