

## แบบฝึกหัดเกี่ยวกับการจัดการไฟล์ใน C++ (ifstream, ofstream, fstream)

วัตถุประสงค์ของแบบฝึกหัด

- 1) เข้าใจหลักการ อ่านและเขียนไฟล์ ใน C++
- 2) ฝึกใช้ ofstream, ifstream, fstream เพื่อจัดการไฟล์
- 3) ใช้โหมดการเปิดไฟล์ (ios::in, ios::out, ios::app, ios::trunc)
- 4) เข้าใจ พาทไฟล์ และ การใช้ไฟล์เดอร์
- 5) 🚀 สรุปแบบฝึกหัด

แบบฝึกหัด	แนวคิดหลัก	ไฟล์ที่เกี่ยวข้อง
1: เขียนและอ่านไฟล์ข้อความ	ofstream, ifstream, getline()	message.txt
2: คำนวณและบันทึกผลลัพธ์	ofstream, ifstream, คำนวณทางคณิตศาสตร์	calculation.txt
3: อ่านข้อมูล CSV	ifstream, getline(), การแยกข้อมูล (,)	data.csv
4: บันทึก Log การใช้งาน	ofstream (ios::app), time()	log.txt
5: อ่านค่า Config	ifstream, getline(), การจัดการ key=value	config.txt
6: แก้ไขข้อมูลในไฟล์	fstream, ifstream, ofstream	data.txt

ฟังก์ชัน	C++	Python
เขียนและอ่านไฟล์ข้อความ	ofstream, ifstream	open("file.txt", "w/r")
คำนวณและบันทึกผลลัพธ์	ofstream, ifstream	open("file.txt", "w/r")
อ่านไฟล์ CSV	ifstream, getline()	csv.reader()
บันทึก Log การใช้งาน	ofstream (ios::app), time()	datetime.now()
อ่านค่า Config	ifstream, getline()	open("config.txt", "r")
แก้ไขข้อมูลในไฟล์	fstream, ofstream	file.read/write()

◆ แบบฝึกหัด 1: เขียนและอ่านไฟล์ข้อความ (.txt)

ดูตัวอย่าง code ที่ LAB12\_06

📌 โจทย์

1. ให้เขียนโปรแกรมรับข้อความจากผู้ใช้
2. บันทึกข้อความลงในไฟล์ message.txt
3. จากนั้นเปิดไฟล์ขึ้นมาอ่าน แล้วแสดงผลข้อความทั้งหมดที่ถูกบันทึก
4. หากไฟล์ยังไม่มี ให้แจ้งเตือนว่า "ไฟล์ยังไม่มีข้อมูล"

📌 ตัวอย่างผลลัพธ์ที่ต้องการ

Enter your message: Hello, this is a test message!

[บันทึกข้อมูลเรียบร้อย!]

[อ่านข้อมูลจากไฟล์ message.txt]

Hello, this is a test message!

## Code C++

```
#include <iostream>
#include <fstream>
#include <string>

using namespace std;

int main() {
    string message;
    cout << "Enter your message: ";
    getline(cin, message);

    ofstream outFile("message.txt");
    outFile << message << endl;
    outFile.close();

    ifstream inFile("message.txt");
    if (inFile) {
        string line;
        cout << "Contents of message.txt:" << endl;
        while (getline(inFile, line)) {
            cout << line << endl;
        }
        inFile.close();
    } else {
```

```
        cout << "The file has no data." << endl;
    }

    return 0;
}
```

ผลการทดลอง

message.txt U X

message.cpp U

message.py U

lab\_12 > message.txt

1 Hello kirato

2

TERMINAL

GITLENS

PROBLEMS

OUTPUT

PORTS

ROBOT OUTPUT

C

● > g++ message.cpp -o message && ./message

Enter your message: Hello kirato

Contents of message.txt:

Hello kirato

~/Desktop/MUT/Y\_1 T\_2/MIIA0106/lab\_12 | main !2 ?2

○ >

## Code Python

```
message = input("Enter your message: ")

with open("message.txt", "w") as outFile:
    outFile.write(message + "\n")

try:
    with open("message.txt", "r") as inFile:
        print("Contents of message.txt:")
        for line in inFile:
            print(line, end="")
except FileNotFoundError:
    print("The file has no data.")
```

ผลการทดลอง

message.txt U × message.py U

lab\_12 > message.txt

```
1 Hello python and CPP too !  
2
```

TERMINAL GITLENS PROBLEMS OUTPUT PORTS ROBOT OUTPUT

```
• > g++ message.cpp -o message && ./message  
Enter your message: Hello kirato  
Contents of message.txt:  
Hello kirato  
• > python3 message.py  
Enter your message: Hello python and CPP too !  
Contents of message.txt:  
Hello python and CPP too !
```

~/Desktop/MUT/Y\_1 T\_2/MIIA0106/lab\_12 | main !2 ?2

○ >



◆ แบบฝึกหัด 2: จำนวนและบันทึกผลลัพธ์ลงไฟล์

ดูตัวอย่าง code ที่ LAB12\_07

โจทย์

1. ให้โปรแกรมรับตัวเลข 2 ค่า (a และ b) จากผู้ใช้
2. จำนวนผลลัพธ์ของ  $a + b$ ,  $a - b$ ,  $a * b$ ,  $a / b$
3. บันทึกผลลัพธ์ลงไฟล์ calculation.txt
4. เปิดไฟล์ calculation.txt แล้วแสดงผลข้อมูลที่บันทึกไว้

ตัวอย่างผลลัพธ์ที่ต้องการ

Enter first number: 10

Enter second number: 5

[บันทึกผลลัพธ์เรียบร้อยแล้ว!]

[อ่านข้อมูลจากไฟล์ calculation.txt]

$10 + 5 = 15$

$10 - 5 = 5$

$10 * 5 = 50$

$10 / 5 = 2$



## Code

```
#include <iostream>
#include <fstream>
using namespace std;

int main() {
    double a, b;
    cout << "Enter first number: ";
    cin >> a;
    cout << "Enter second number: ";
    cin >> b;

    double sum = a + b;
    double diff = a - b;
    double prod = a * b;
    double quot = a / b;


    ofstream outfile("calculation.txt");
    outfile << a << " + " << b << " = " << sum << endl;
    outfile << a << " - " << b << " = " << diff << endl;
    outfile << a << " * " << b << " = " << prod << endl;
    outfile << a << " / " << b << " = " << quot << endl;
    outfile.close();

    cout << "[บันทึกผลลัพธ์เรียบร้อยแล้ว]" << endl;

    ifstream infile("calculation.txt");
    string line;
    cout << "\n[อ่านข้อมูลจากไฟล์calculation.txt]" << endl;
    while (getline(infile, line)) {
        cout << line << endl;
    }
    infile.close();

    return 0;
}
```

ผลการทดลอง

 calculation.txt U ✕

lab\_12 >  calculation.txt

```
1 1024 + 8 = 1032
2 1024 - 8 = 1016
3 1024 * 8 = 8192
4 1024 / 8 = 128
5
```

TERMINAL

GITLENS

PROBLEMS

OUTPUT

PORTS

ROBOT OUTPUT

● > g++ calculation.cpp -o cal && ./cal

Enter first number: 1024

Enter second number: 8

[บันทึกผลลัพธ์เรียบร้อยแล้ว]

[อ่านข้อมูลจากไฟล์ calculation.txt]

1024 + 8 = 1032

1024 - 8 = 1016

1024 \* 8 = 8192

1024 / 8 = 128

~/Desktop/MUT/Y\_1 T\_2/MIIA0106/lab\_12 | main !2 ?2

○ >

## Code Python

```
def main():
    a = float(input("Enter first number: "))
    b = float(input("Enter second number: "))

    sum_result = a + b
    diff_result = a - b
    prod_result = a * b
    quot_result = a / b

    with open("calculation.txt", "w") as file:
        file.write(f"{a} + {b} = {sum_result}\n")
        file.write(f"{a} - {b} = {diff_result}\n")
        file.write(f"{a} * {b} = {prod_result}\n")
        file.write(f"{a} / {b} = {quot_result}\n")

    print("[บันทึกผลลัพธ์เรียบร้อยแล้ว]")

    print("\n[อ่านข้อมูลจากไฟล์ calculation.txt]")
    with open("calculation.txt", "r") as file:
        for line in file:
            print(line.strip())

if __name__ == "__main__":
    main()
```

ผลการทดลอง

calculation.txt U ✕

lab\_12 > calculation.txt

```
1 512.0 + 8.0 = 520.0
2 512.0 - 8.0 = 504.0
3 512.0 * 8.0 = 4096.0
4 512.0 / 8.0 = 64.0
5
```

TERMINAL

GITLENS

PROBLEMS

OUTPUT

PORTS

ROBOT OUTPUT

COMMENTS

Enter first number: 1024

Enter second number: 8

[บันทึกผลลัพธ์เรียบร้อยแล้ว!]

[อ่านข้อมูลจากไฟล์ calculation.txt]

1024 + 8 = 1032

1024 - 8 = 1016

1024 \* 8 = 8192

1024 / 8 = 128

● > python3 calculation.py

Enter first number: 512

Enter second number: 8

[บันทึกผลลัพธ์เรียบร้อยแล้ว!]

[อ่านข้อมูลจากไฟล์ calculation.txt]

512.0 + 8.0 = 520.0

512.0 - 8.0 = 504.0

512.0 \* 8.0 = 4096.0

512.0 / 8.0 = 64.0

~/Desktop/MUT/Y\_1 T\_2/MIIA0106/lab\_12 | main !2 ?2

◆ แบบฝึกหัด 3: อ่านไฟล์ CSV และแสดงผลข้อมูล

ดูตัวอย่าง code ที่ LAB12\_08

📌 โจทย์

1. ให้มีไฟล์ data.csv ที่เก็บข้อมูลพนักงานในรูปแบบ CSV เช่น:
2. ID,Name,Salary
3. 101,John,50000
4. 102,Alice,60000
5. 103,Bob,55000
6. ให้เขียนโปรแกรม อ่านข้อมูลจากไฟล์ data.csv
7. แสดงข้อมูลทั้งหมดออกมาทางหน้าจอในรูปแบบตาราง

📌 ตัวอย่างผลลัพธ์ที่ต้องการ

[แสดงข้อมูลจากไฟล์ data.csv]

```
-----  
| ID | Name | Salary |  
-----  
| 101 | John | 50000 |  
| 102 | Alice | 60000 |  
| 103 | Bob | 55000 |  
-----
```

## Code

```
#include <iostream>
#include <fstream>
#include <sstream>
#include <vector>
#include <iomanip>
#include <stdexcept>

using namespace std;

struct Employee {
    int id;
    string name;
    int salary;
};

int main() {
    ifstream file("data.csv");
    if (!file) {
        cerr << "Unable to open file data.csv" << endl;
        return 1;
    }

    string line;
    vector<Employee> employees;

    // Skip the header line
    getline(file, line);

    // Read the data
    while (getline(file, line)) {
        stringstream ss(line);
        string id, name, salary;
        getline(ss, id, ',');
        getline(ss, name, ',');
        getline(ss, salary, ',');

        try {
            int emp_id = stoi(id);
            int emp_salary = stoi(salary);
            employees.push_back({emp_id, name, emp_salary});
        } catch (const invalid_argument& e) {
            cerr << "Invalid data: " << line << endl;
        } catch (const out_of_range& e) {
            cerr << "Out of range data: " << line << endl;
        }
    }
}
```

```

file.close();

// Display the data in table format
cout << "-----" << endl;
cout << "| ID   | Name   | Salary   |" << endl;
cout << "-----" << endl;
for (const auto& emp : employees) {
    cout << "| " << setw(4) << emp.id << " | " << setw(6) << emp.name << " | "
<< setw(10) << emp.salary << " |" << endl;
}
cout << "-----" << endl;

return 0;
}

```

ผลการทดลอง

data.csv U X

lab\_12 > data.csv > data

```
1 ID,Name,Salary
2 101,John,50000
3 102,Alice,60000
4 103,Bob,55000
5 104,Kiratipat,100000
6
```

TERMINAL

GITLENS

PROBLEMS

OUTPUT

PORTS

ROBOT OUTPUT

COMMENTS

● > g++ -std=c++11 display\_data.cpp -o csv && ./csv

```
-----
| ID | Name | Salary |
-----
| 101 | John | 50000 |
| 102 | Alice | 60000 |
| 103 | Bob | 55000 |
| 104 | Kiratipat | 100000 |
-----
```

~/Desktop/MUT/Y\_1 T\_2/MIIA0106/lab\_12 | main !2 ?2



## Code Python

```
import csv

employees = []

# Read the data from the CSV file
with open('data.csv', newline='') as csvfile:
    reader = csv.DictReader(csvfile)
    for row in reader:
        employees.append(row)

# Display the data in table format
print("-----")
print("| ID    | Name    | Salary    |")
print("-----")
for emp in employees:
    print(f"| {emp['ID']:4} | {emp['Name']:6} | {emp['Salary']:10} |")
print("-----")
```

ผลการทดลอง

data.csv U X

lab\_12 > data.csv > data

```
1 ID,Name,Salary
2 101,John,50000
3 102,Alice,60000
4 103,Bob,55000
5 104,Kiratipat,6711860001
6
```

TERMINAL

GITLENS

PROBLEMS

OUTPUT

PORTS

ROBOT OUTPUT

COMMENTS

• > python3 display\_data.py

```
-----
| ID   | Name   | Salary |
-----
| 101  | John   | 50000  |
| 102  | Alice  | 60000  |
| 103  | Bob    | 55000  |
| 104  | Kiratipat | 900000 |
-----
```

• > python3 display\_data.py

```
-----
| ID   | Name   | Salary |
-----
| 101  | John   | 50000  |
| 102  | Alice  | 60000  |
| 103  | Bob    | 55000  |
| 104  | Kiratipat | 6711860001 |
-----
```

#### ◆ แบบฝึกหัด 4: บันทึก Log การใช้งานลงไฟล์

##### 📌 โจทย์

1. ให้เขียนโปรแกรมที่บันทึก "กิจกรรมการใช้งาน" ลงไฟล์ log.txt
2. ทุกครั้งที่โปรแกรมทำงาน ให้เพิ่มข้อความ "User started the program at [timestamp]"
3. อ่านไฟล์ log.txt และแสดงผลประวัติการใช้งานทั้งหมด

##### 📌 ตัวอย่างผลลัพธ์ที่ต้องการ

[บันทึกการใช้งานลงไฟล์เรียบร้อยแล้ว]

[อ่านข้อมูลจาก log.txt]

User started the program at 2025-02-14 12:34:56

User started the program at 2025-02-14 12:45:20

## Code

```
#include <iostream>
#include <fstream>
#include <ctime>
using namespace std;

void logActivity() {
    ofstream logFile("log.txt", ios_base::app);
    if (logFile.is_open()) {
        time_t now = time(nullptr);
        logFile << "User started the program at " << ctime(&now);
        logFile.close();
        cout << "[บันทึกการใช้งานลงไฟล์เรียบร้อย]\n";
    } else {
        cerr << "Unable to open log file.\n";
    }
}

void readLog() {
    ifstream logFile("log.txt");
    if (logFile.is_open()) {
        cout << "\n[อ่านข้อมูลจาก log.txt]\n";
        string line;
        while (getline(logFile, line)) {
            cout << line << "\n";
        }
        logFile.close();
    } else {
        cerr << "Unable to open log file.\n";
    }
}

int main() {
    logActivity();
    readLog();
    return 0;
}
```

 log.txt U Xlab\_12 >  log.txt

```
1 User started the program at Sun Feb 16 08:31:30 2025
2 User started the program at Sun Feb 16 08:32:06 2025
3 User started the program at Sun Feb 16 08:32:07 2025
4 User started the program at Sun Feb 16 08:32:07 2025
5 User started the program at Sun Feb 16 08:32:09 2025
6 User started the program at Sun Feb 16 09:19:07 2025
7
```

TERMINAL

GITLENS

PROBLEMS

OUTPUT

PORTS

ROBOT OUTPUT

COMMENTS

```
● > g++ activity_logger.cpp -o act 88 ./act
[บันทึกการใช้งานลงไฟล์เรียบร้อยแล้ว]
```

```
[อ่านข้อมูลจาก log.txt]
```

```
User started the program at Sun Feb 16 08:31:30 2025
User started the program at Sun Feb 16 08:32:06 2025
User started the program at Sun Feb 16 08:32:07 2025
User started the program at Sun Feb 16 08:32:07 2025
User started the program at Sun Feb 16 08:32:09 2025
User started the program at Sun Feb 16 09:19:07 2025
```

```
~/Desktop/MUT/Y_1 T_2/MIIA0106/lab_12 | main !2 ?2
```

```
○ > █
```

## Code Python

```
import datetime

def log_activity():
    with open("log.txt", "a") as log_file:
        now = datetime.datetime.now()
        log_file.write(f"User started the program at {now.strftime('%Y-%m-%d
%H:%M:%S')}}\n")
        print("[บันทึกการใช้งานลงไฟล์เรียบร้อยแล้ว]")

def read_log():
    try:
        with open("log.txt", "r") as log_file:
            print("\n[อ่านข้อมูลจาก log.txt]")
            for line in log_file:
                print(line, end="")
    except FileNotFoundError:
        print("Unable to open log file.")

if __name__ == "__main__":
    log_activity()
    read_log()
```

ผลการทดลอง

```
log.txt U X
lab_12 > log.txt
1 User started the program at Sun Feb 16 08:31:30 2025
2 User started the program at Sun Feb 16 08:32:06 2025
3 User started the program at Sun Feb 16 08:32:07 2025
4 User started the program at Sun Feb 16 08:32:07 2025
5 User started the program at Sun Feb 16 08:32:09 2025
6 User started the program at Sun Feb 16 09:19:07 2025
7 User started the program at 2025-02-16 09:19:36
8 User started the program at 2025-02-16 09:19:40
9

TERMINAL  GITLENS  PROBLEMS  OUTPUT  PORTS  ROBOT OUTPUT  COMMENTS
User started the program at Sun Feb 16 08:31:30 2025
User started the program at Sun Feb 16 08:32:06 2025
User started the program at Sun Feb 16 08:32:07 2025
User started the program at Sun Feb 16 08:32:07 2025
User started the program at Sun Feb 16 08:32:09 2025
User started the program at Sun Feb 16 09:19:07 2025
User started the program at 2025-02-16 09:19:36
● > python3 activity_logger.py
[บันทึกการใช้งานลงไฟล์เรียบร้อยแล้ว]

[อ่านข้อมูลจาก log.txt]
User started the program at Sun Feb 16 08:31:30 2025
User started the program at Sun Feb 16 08:32:06 2025
User started the program at Sun Feb 16 08:32:07 2025
User started the program at Sun Feb 16 08:32:07 2025
User started the program at Sun Feb 16 08:32:09 2025
User started the program at Sun Feb 16 09:19:07 2025
User started the program at 2025-02-16 09:19:36
User started the program at 2025-02-16 09:19:40
~/Desktop/MUT/Y_1 T_2/MIIA0106/lab_12 | main !2 ?2
○ >
```

◆ แบบฝึกหัด 5: อ่านค่า **Config** จากไฟล์

📌 โจทย์

1. ให้มีไฟล์ config.txt ที่มีค่าตั้งค่า เช่น:
2. speed=2.5
3. max\_distance=150
4. mode=auto
5. ให้เขียนโปรแกรม อ่านค่า **config** จากไฟล์
6. แสดงค่าที่อ่านได้ออกมาทางหน้าจอ

📌 ตัวอย่างผลลัพธ์ที่ต้องการ

[Robot Configuration]

Speed: 2.5 m/s

Max Distance: 150 cm

Mode: Auto



## Code

```
#include <iostream>
#include <fstream>
#include <string>
#include <map>

using namespace std;

map<string, string> read_config(const string& file_path) {
    map<string, string> config;
    ifstream file(file_path);
    string line;
    while (getline(file, line)) {
        size_t pos = line.find('=');
        if (pos != string::npos) {
            string key = line.substr(0, pos);
            string value = line.substr(pos + 1);
            config[key] = value;
        }
    }
    return config;
}

void display_config(const map<string, string>& config) {
    cout << "[Robot Configuration]" << endl;
    cout << "Speed: " << config.at("speed") << " m/s" << endl;
    cout << "Max Distance: " << config.at("max_distance") << " cm" << endl;
    cout << "Mode: " << config.at("mode") << endl;
}

int main() {
    map<string, string> config = read_config("config.txt");
    display_config(config);
    return 0;
}
```

ผลการทดลอง

```
config.txt U x
lab_12 > config.txt
1 speed=5.43
2 max_distance=1,500
3 mode=auto
4

TERMINAL  GITLENS  PROBLEMS  OUTPUT  PORTS  ROBOT OUTPUT  COMMENTS
● > g++ read_config.cpp -o config && ./config
[Robot Configuration]
Speed: 5.43 m/s
Max Distance: 1,500 cm
Mode: auto
~/Desktop/MUT/Y_1 T_2/MIIA0106/lab_12 | main !2 ?2
○ > 
```

## Code Python

```
def read_config(file_path):
    config = {}
    with open(file_path, 'r') as file:
        for line in file:
            key, value = line.strip().split('=')
            config[key] = value
    return config

def display_config(config):
    print("[Robot Configuration]")
    print(f"Speed: {config['speed']} m/s")
    print(f"Max Distance: {config['max_distance']} cm")
    print(f"Mode: {config['mode'].capitalize()}")

if __name__ == "__main__":
    config = read_config('config.txt')
    display_config(config)
```

ผลการทดลอง

 config.txt U ✕

lab\_12 >  config.txt

```
1 speed=100
2 max_distance=10,000
3 mode=auto
4
```

TERMINAL

GITLENS

PROBLEMS

OUTPUT

PORTS

ROBOT OUTPUT

COMMENTS

```
● > python3 read_config.py
[Robot Configuration]
Speed: 100 m/s
Max Distance: 10,000 cm
Mode: Auto
~/Desktop/MUT/Y_1 T_2/MIIA0106/lab_12 | main !2 ?2
○ >
```

## ◆ แบบฝึกหัด 6: แก้ไขข้อมูลในไฟล์

ดูตัวอย่าง code ที่ LAB12\_09

📌 โจทย์

1. ให้มีไฟล์ data.txt ที่มีข้อมูลเดิม เช่น:
2. Apple
3. Banana
4. Cherry
5. ให้เขียนโปรแกรมที่สามารถ แก้ไขข้อมูลในไฟล์ ได้
6. ให้ผู้ใช้ป้อน "คำที่ต้องการแก้ไข" และ "คำใหม่ที่จะแทนที่"
7. บันทึกข้อมูลใหม่ลงไฟล์

📌 ตัวอย่างผลลัพธ์ที่ต้องการ

[ไฟล์เดิม: data.txt]

Apple

Banana

Cherry

Enter word to replace: Banana

Enter new word: Mango

[แก้ไขข้อมูลในไฟล์เรียบร้อยแล้ว!]

[ไฟล์ใหม่: data.txt]

Apple

Mango

Cherry

## Code

```
#include <iostream>
#include <fstream>
#include <string>
#include <vector>

using namespace std;
```

```

int main() {
    ifstream file("data.txt");
    vector<string> lines;

    if (!file) {
        cerr << "Unable to open file data.txt. Creating a new file with default
values." << endl;
        vector<string> default_data = {"Hello", "World", "C++", "first initial"};
        ofstream outfile("data.txt");
        for (const auto &l : default_data) {
            outfile << l << endl;
        }
        outfile.close();
        lines = default_data;
    } else {
        string line;
        while (getline(file, line)) {
            lines.push_back(line);
        }
        file.close();
    }

    string old_word, new_word;
    cout << "Enter word to replace: ";
    cin >> old_word;
    cout << "Enter new word: ";
    cin >> new_word;

    for (auto &l : lines) {
        if (l == old_word) {
            l = new_word;
        }
    }

    ofstream outfile("data.txt");
    for (const auto &l : lines) {
        outfile << l << endl;
    }
    outfile.close();

    cout << "[แก้ไขข้อมูลในไฟล์เรียบร้อยแล้ว]" << endl;
    return 0;
}

```

ผลการทดลอง

replace\_word.cpp M × data.txt M ×

~/Desktop/MUT/Y\_1 T\_2/MIA0106/lab\_12/replace\_word.cpp • Modified

You, 3 seconds ago | 1 author (You)

```
1 Hello
2 World
3 C++
4 first initial
5
```

TERMINAL GITLENS PROBLEMS OUTPUT PORTS ROBOT OUTPUT COMMENTS

```
> g++ -std=c++11 replace_word.cpp -o re && ./re
Unable to open file data.txt. Creating a new file with default values.
Enter word to replace: 
```

C++ replace\_word.cpp M

data.txt M X

lab\_12 > data.txt

You, 1 second ago | 1 author (You)

```
1 Hello You, 3 seconds ago • Uncommitted changes
2 World
3 Python
4 first initial
5
```

TERMINAL

GITLENS

PROBLEMS

OUTPUT

PORTS

ROBOT OUTPUT

COMMENTS

```
● > g++ -std=c++11 replace_word.cpp -o re && ./re
Unable to open file data.txt. Creating a new file with default values.
Enter word to replace: C++
Enter new word: Python
[แก้ไขข้อมูลในไฟล์เรียบร้อยแล้ว!]
~/Desktop/MUT/Y_1 T_2/MIIA0106/lab_12 | main +29 !4
○ > 
```



## Code Python

```
def replace_word_in_file(file_path, old_word, new_word):
    with open(file_path, 'r') as file:
        lines = file.readlines()

    with open(file_path, 'w') as file:
        for line in lines:
            if line.strip() == old_word:
                file.write(new_word + '\n')
            else:
                file.write(line)

if __name__ == "__main__":
    file_path = 'data.txt'
    old_word = input("Enter word to replace: ")
    new_word = input("Enter new word: ")
    replace_word_in_file(file_path, old_word, new_word)
    print("[แก้ไขข้อมูลในไฟล์เรียบร้อยแล้ว]")
```

ผลการทดลอง

data.txt U × C++ replace\_word.cpp U TEMP.md 1, U replace\_word.py U

lab\_12 > data.txt

```
1 hello
2 Word
3 I'm python !
4 6711860001
5
6
```

TERMINAL GITLENS PROBLEMS 1 OUTPUT PORTS ROBOT OUTPUT COMMENTS

```
• > g++ -std=c++11 replace_word.cpp -o re && ./re
Enter word to replace: Orange
Enter new word: Word
[แก้ไขข้อมูลในไฟล์เรียบร้อยแล้ว!]
• > g++ -std=c++11 replace_word.cpp -o re && ./re
Enter word to replace: kirato
Enter new word: 6711860001
[แก้ไขข้อมูลในไฟล์เรียบร้อยแล้ว!]
• > python3 replace_word.py
Enter word to replace: Cherry
Enter new word: I'm python !
[แก้ไขข้อมูลในไฟล์เรียบร้อยแล้ว!]
~/Desktop/MUT/Y_1 T_2/MIIA0106/lab_12 | main !2 ?2
○ > 
```