Bahria University,

Karachi Campus



LAB EXPERIMENT NO.

**06**

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| 1 | **Write a program that pass your information to client side by using all above mentioned client and server function.**   * + **Your information message must contain following details.**     - **“student\_information**     - **Name = arooba**     - **Section = A/b**     - **Department = software “** |
|  |  |
|  |  |

Submitted On:

**29/12/2022**

**Task # 01: Write a program that pass your information to client side by using all above mentioned client and server function.**

* + **Your information message must contain following details.**
    - **“student\_information**
    - **Name = arooba**
    - **Section = A/b**
    - **Department = software “**

**Solution:**

**Server Code:**

#define UNICODE

#include<Windows.h>

#include<windowsx.h>

#include<iostream>

using namespace std;

int main() {

HANDLE fFileMap;

BOOL bResult;

PCHAR lpBuffer = NULL;

char Buffer[1024] = ".........WELCOME..........";

char c[1024] = "\t\"Student Information\"\nName: Muaz\nSection: B\nDepartment: Software";

size\_t szBuffer = sizeof(c);

//step 1

fFileMap = CreateFileMapping(INVALID\_HANDLE\_VALUE, NULL, PAGE\_READWRITE, 0, 256, L"Local\\MyFile");

if (fFileMap == NULL) {

cout << "Creating File Mapping Error No: " << GetLastError() << endl;

}

cout << "Creating File Mapped Successful " << endl;

//step 2 map view file

lpBuffer = (PCHAR)MapViewOfFile(fFileMap, FILE\_MAP\_ALL\_ACCESS, 0, 0, 256);

if (lpBuffer == NULL) {

cout << "Map View File Error No: " << GetLastError() << endl;

}

cout << "Map View File Success" << endl;

//step 3 copy map view file

CopyMemory(lpBuffer, c, szBuffer);

//step 4 Unmapped

bResult = UnmapViewOfFile(lpBuffer);

if (bResult == false) {

cout << "Unmapped View File Error No" << GetLastError() << endl;

}

cout << "Unmapped Success" << endl;

system("PAUSE");

return 0;

}

**Client Code:**

#define UNICODE

#include<Windows.h>

#include<windowsx.h>

#include<iostream>

using namespace std;

int main() {

cout << "File\_mapping Client Side" << endl;

HANDLE fFileMap;

BOOL bResult;

PCHAR lpBuffer = NULL;

//step 1

fFileMap = OpenFileMapping(FILE\_MAP\_ALL\_ACCESS, FALSE, L"Local\\MyFile");

if (fFileMap == NULL) {

cout << "Creating file Mapping Error No: " << GetLastError() << endl;

}

else {

cout << "Creating file Mapped Successful " << endl;

}

//step 2

lpBuffer = (PCHAR)MapViewOfFile(fFileMap, FILE\_MAP\_ALL\_ACCESS, 0, 0, 256);

if (lpBuffer == NULL) {

cout << "Map View File Error No: " << GetLastError() << endl;

}

else {

cout << "Map View File Success" << endl;

}

//step 3

cout << "Data Reading From Server\n" << lpBuffer << endl;

//step 4

bResult = UnmapViewOfFile(lpBuffer);

if (bResult == false) {

cout << "Unmapped View File Error No" << GetLastError() << endl;

}

else {

cout << "\n\nUnmapped Success" << endl;

}

CloseHandle(fFileMap);

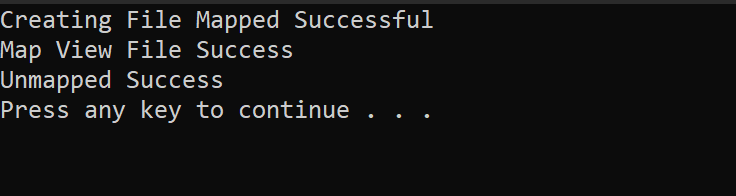
system("PAUSE");

return 0;

}

**Output:**

**Server:**

****

**Client:**

**Text

Description automatically generated**