**Name : Sundas Ambreen**

**Class: BSCS-2**

**Sap ID : 58822**

**Lab 13:**

***Task 1:***

#include <iostream>

#include <fstream> // Required for file operations

using namespace std;

int main() {

// Create an ofstream object to write to a file

ofstream file("filename.txt");

// Check if the file is open

if (file.is\_open()) {

cout << "This is a sample file.\n";

cout << "Line 1:Hello, World!\n";

cout << "Line 2:File handling in C++\n";

file.close();

} else {

// If the file could not be opened

cout<< "Error: Unable to create or write to the file!" <<endl;

}

return 0;

}

***Task 2:***

#include <iostream>

#include <fstream> // For file operations

using namespace std;

int main() {

// Create an ofstream object to write to a file

std::ofstream file("output.txt");

// Check if the file is open

if (file.is\_open()) {

// Write the content into the file

cout << "Output file\n";

cout << "Line 1:Writting to a file in C++\n";

cout << "Line 2:File handling is essential for data manipulation \n";

// Close the file

file.close();

cout << "File 'output.txt' has been created and content written successfully." << endl;

} else {

// If the file could not be opened

cout << "Error: Unable to create or write to the file!" << endl;

}

return 0;

}

***Task 3:***

#include <iostream>

#include <fstream>

#include <vector>

#include <string>

using namespace std;

void logActivity(const string& message) {

ofstream logFile("UserLog.txt",ios::app);

if (!logFile) {

cerr << "Error: Unable to open file for logging." << endl;

return;

}

logFile << message << endl;

logFile.close();

}

void readLog() {

ifstream logFile("UserLog.txt");

if (!logFile) {

cerr << "Error: Unable to open file for reading." << endl;

return;

}

string line;

cout << "\nUser Activities Log:\n";

while (getline(logFile, line)) {

cout << line << endl;

}

logFile.close();

}

int main() {

"User 'John' logged in.",

"User 'Jane' accessed the dashboard.",

"User 'Bob' performed a file operation.",

cout << "Reading logged activities...\n";

readLog();

return 0;

}