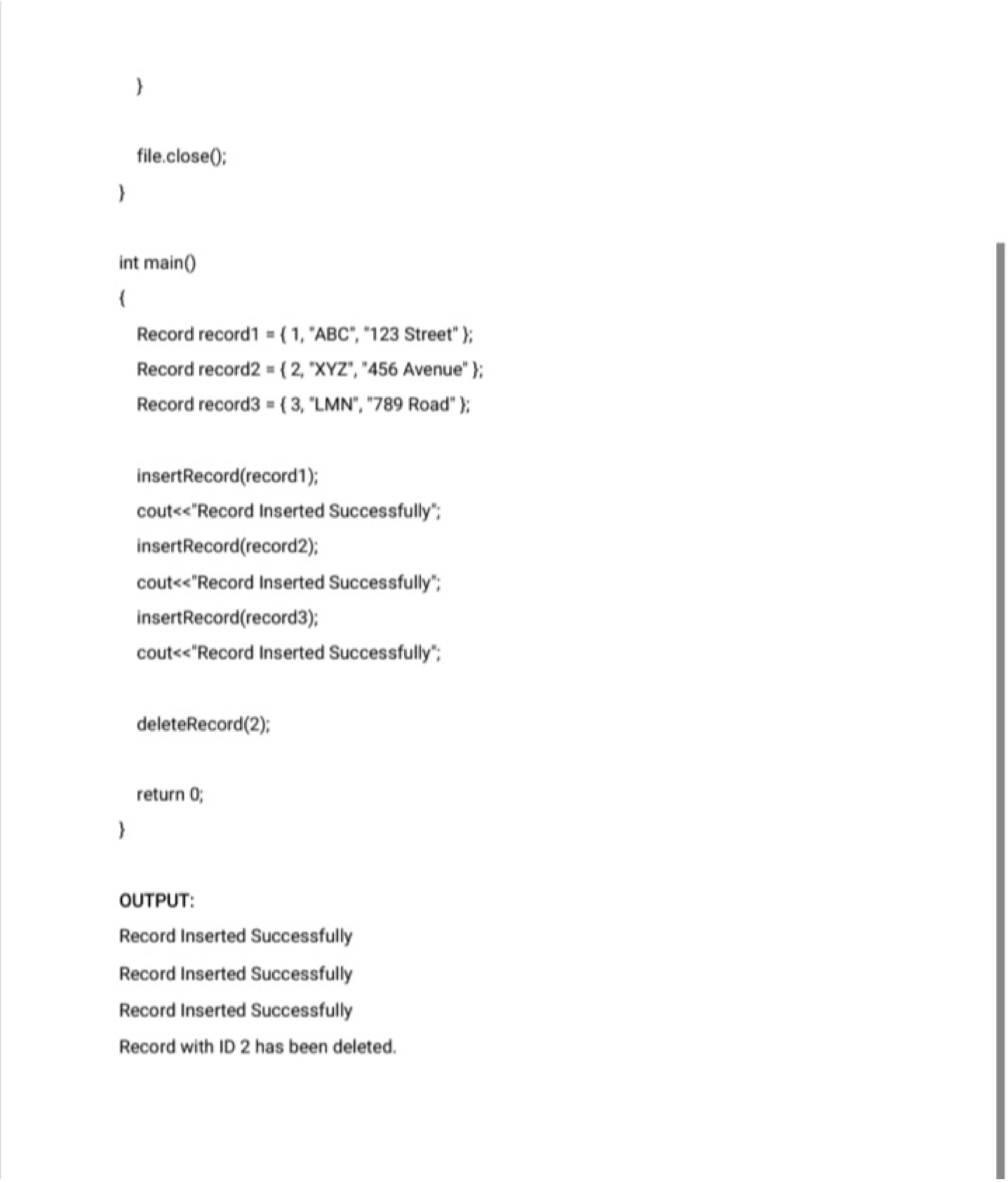
CODE:

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

using namespace std;

struct Record {
   int id;
   string name;
   string address;
};

void insertRecord(const Record& record)
{
   ofstream file("data.txt", ios::binary | ios::app);
   if (!file) {
      cout << "Error opening file for writing!" << endl;
      return;
   }

   file.write(reinterpret_cast<const char*>(&record), sizeof(Record));

   file.close();
}
```

```
void deleteRecord(int id)
{
    fstream file("data.txt", ios::binary | ios::in | ios::out);
    if (!file) {
        cout << "Error opening file for reading and writing!" << endl;
        return;
    }

    Record record;
    bool found = false;
    while (!file.eof()) {
        streampos position = file.tellg();
        file.read(reinterpret_cast<char*>(&record), sizeof(Record));
        if (file.eof())
            break;

        if (record.id == id) {
            found = true;
            break;
        }
    }

    if (found) {
        file.seekp(-static_cast<int>(sizeof(Record)), ios::cur);
        Record emptyRecord;
        file.write(reinterpret_cast<const char*>(&emptyRecord), sizeof(Record));
        cout << "Record with ID " << id << " has been deleted." << endl;
    } else {
        cout << "Record with ID " << id << " not found." << endl;
```



```
    }

    file.close();
}

int main()
{
    Record record1 = { 1, "ABC", "123 Street" };
    Record record2 = { 2, "XYZ", "456 Avenue" };
    Record record3 = { 3, "LMN", "789 Road" };

    insertRecord(record1);
    cout<<"Record Inserted Successfully";
    insertRecord(record2);
    cout<<"Record Inserted Successfully";
    insertRecord(record3);
    cout<<"Record Inserted Successfully";

    deleteRecord(2);

    return 0;
}
```

OUTPUT:

```
Record Inserted Successfully
Record Inserted Successfully
Record Inserted Successfully
Record with ID 2 has been deleted.
```