XML FILE GENERATOR

GROUP MEMEBERS :

1] RAJVAIBHAV RAHANE.(17U283)

2] JUGAL PATIL (17U183)

3] SAKSHI KHODE (17U116)

4] SHRAVAN SHETTY(17U201)

5] SUMEDH KULKARNI(17U150)

AIM :

Student database of marks in spreadsheet.

DATA STRUCTURES :

1] List

2] Map

PROGRAM :

```
/*
*@Rajvaibhav Rahane
*/
#include<iostream>
#include<map>
#include<string.h>
#include<list>
#include<fstream>
using namespace std;
class Exam{
```

```cpp
    string name;

    int totalMarks;

    public:

    Exam(){}

    void setName(string name){

        this->name=name;

    }

    string getName(){

        return name;

    }

    void setTotalMarks(int totalMarks){

        this->totalMarks=totalMarks;

    }

    int getTotalMarks(){

        return totalMarks;

    }
};
class ExamDetails{

    private:

    list<Exam> examList;

    public:

        void addExam();

        list<Exam> getExamList();

        int getExamListSize();

};
```

```cpp
void ExamDetails::addExam(){

    string examName;

    int totalMarks;

    cout<<"Enter Exam name : \t";

    cin>>examName;

    cout<<"Enter Total Marks : \t";

    cin>>totalMarks;


    Exam *exam=new Exam();

    (*exam).setName(examName);

    (*exam).setTotalMarks(totalMarks);

    examList.push_back(*exam);

}
list<Exam> ExamDetails::getExamList(){

    return examList;

}
int ExamDetails::getExamListSize(){

    return examList.size();

}
class Student{

    string name;

    int prnNo;

    map<Exam*,int> examMarksMap;

    public:

    void setPrnNo(int prnNo){
```

```cpp
            this->prnNo=prnNo;

        }

        void setName(string name){

            this->name=name;

        }

        string getName(){

            return name;

        }

        int getPrnNo(){

            return prnNo;

        }

        map<Exam*,int> getExamMarksMap(){

            return examMarksMap;

        }

        void setStudent(ExamDetails);

        void printStudent();

};

void Student::setStudent(ExamDetails examDetails){

        cout<<"Enter Name : ";cin>>name;

        cout<<"Enter PRN No : ";cin>>prnNo;

        list<Exam> examList=examDetails.getExamList();

        cout<<"Enter Subject Marks\n";

        int marks;

        for(list<Exam>::iterator it=examList.begin();it!=examList.end();it++){

            cout<<(*it).getName()<<" : ";cin>>marks;Exam exam;
```

```cpp
        std::pair<Exam*,int> examMarks=std::make_pair(&(*it),marks);

        examMarksMap.insert(examMarks);

    }

}

void Student::printStudent(){

    cout<<name<<""<<prnNo<<endl;

    for(map<Exam*,int>::iterator
it=examMarksMap.begin();it!=examMarksMap.end();it++){

        cout << it->first->getName() <<" =>"<< it->second << '\n';

    }

}

void createFile(ExamDetails examDetails){

    ofstream studentDB_oFile("examResults.xlsx");


    studentDB_oFile<<"Roll No\t"<<"Name\t";


    list<Exam> examList=examDetails.getExamList();

    for(list<Exam>::iterator it=examList.begin();it!=examList.end();it++){

        studentDB_oFile<<(*it).getName()<<"\t";

    }

    studentDB_oFile<<"Total\t"<<endl;

    studentDB_oFile.close();

}

void addStudent(ExamDetails examDetails){

    Student student;

    student.setStudent(examDetails);
```

```cpp
        student.printStudent();

        fstream studentDatabase_oFile("examResults.xlsx");

        cout<<studentDatabase_oFile.is_open()<<endl;

        if(!studentDatabase_oFile.is_open()){     //file does not exist

            createFile(examDetails);

        }else{               //file exists

            studentDatabase_oFile.close();

        }

        studentDatabase_oFile.open("examResults.xlsx",ios::app);

        if(studentDatabase_oFile.is_open()){

        cout<<"open\n";

        studentDatabase_oFile<<student.getPrnNo()<<"\t"<<student.getName()<<"
\t";

        map<Exam*,int> examMarksMap=student.getExamMarksMap();

        int sum=0;

        for(map<Exam*,int>::iterator
it=examMarksMap.begin();it!=examMarksMap.end();it++){

            sum+=it->second;

            studentDatabase_oFile << it->second << '\t';

        }

        studentDatabase_oFile<<sum<<"\t"<<endl;


        studentDatabase_oFile.close();

        }

    }

    void printMenu(){
```

```cpp
        cout<<"\nMenu\n";

        cout<<"\t1)Add Student\n";

        cout<<"\t2)Exit\nChoice : ";

    }
    int main(){

        ExamDetails examDetails;

        /*fstream iFile("examDetails1.txt",ios::in);

        if(iFile.is_open()){

            iFile.close();

            iFile.open("examDetails1.txt",ios::in);cout<<"File Opened\n";

            while(!iFile.eof()){

                iFile.read((char*)(&examDetails),sizeof(ExamDetails));

            }
            iFile.close();

        }else{*/

            int exams;

            cout<<"Enter Number of Exams : ";

            do{

                cin>>exams;

                if(exams<=0)

                    cout<<"Number of Exams should be greater than zero,
Reenter : ";

            }while(exams<=0);


            while(exams--)

                examDetails.addExam();
```

```cpp
        //ofstream oFile("examDetails1.txt",ios::out);

        //list<Exam> examList=examDetails.getExamList();

        //oFile.write((char*)(&examDetails),sizeof(ExamDetails));

        /*for(list<Exam>::iterator
it=examList.begin();it!=examList.end();it++){



        }*/

        //oFile.close();

    //}

    list<Exam> examList=examDetails.getExamList();

    for(list<Exam>::iterator it=examList.begin();it!=examList.end();it++){

        cout<<(*it).getName()<<""<<(*it).getTotalMarks()<<"\t";

    }cout<<endl;

    int choice,count;

    do{

        printMenu();cin>>choice;

        switch(choice){

            case 1:{

                addStudent(examDetails);break;

            }case 2:{

                break;

            }

        }

    }while(choice!=2);

    return 0;

}
```

OUTPUT :

SPREADSHEET :