

// Write a program in C++ to Create a student database of SE IT class (at least 10 records)
and perform Selection Sort.
// Using C++

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

typedef struct student
{
    int roll_num;
    string name;
    float marks;
} stud;

void create(stud s[20], int n);
void display(stud s[20], int n);
void selectionSort(stud s[20], int n);

int main()
{
    stud s[20];
    int ch, n;

    do
    {
        cout << "\n 1) Create Student Database";
        cout << "\n 2) Display Student Records";
        cout << "\n 3) Selection Sort by Marks";
        cout << "\n 4) Exit";
        cout << "\n Enter Your Choice: ";
        cin >> ch;

        switch (ch)
        {
            case 1:
                cout << "\n Enter the Number of Records (at least 10): ";
                cin >> n;
                if (n < 10)
                {
                    cout << "\n Please enter at least 10 records.\n";
                    break;
                }
                create(s, n);
                break;

            case 2:
```

```

        display(s, n);
        break;

    case 3:
        selectionSort(s, n);
        display(s, n);
        break;

    case 4:
        return 0;

    default:
        cout << "\n Invalid choice" << endl;
    }
} while (ch != 4);
}

void create(stud s[20], int n)
{
    for (int i = 0; i < n; i++)
    {
        cout << "\n Enter the roll number: ";
        cin >> s[i].roll_num;
        cin.ignore();
        cout << " Enter the Name: ";
        getline(cin, s[i].name);
        cout << " Enter the marks: ";
        cin >> s[i].marks;
    }
}

void display(stud s[20], int n)
{
    cout << "\n\tRoll No\t\tName\t\tMarks";
    for (int i = 0; i < n; i++)
    {
        cout << "\n\t" << s[i].roll_num << "\t\t" << s[i].name << "\t\t" << s[i].marks;
    }
    cout << endl;
}

void selectionSort(stud s[20], int n)
{
    stud temp;
    for (int i = 0; i < n - 1; i++)
    {
        int minIdx = i;
        for (int j = i + 1; j < n; j++)

```

```
{
    if (s[j].marks < s[minIdx].marks)
        minIdx = j;
}
temp = s[i];
s[i] = s[minIdx];
s[minIdx] = temp;
}
cout << "\nRecords sorted by Marks (Selection Sort).\n";
}
```