

Lab Task 6:

Name: Saif Majid Khan

SAP-ID: 57114

DSA

Github:

<https://github.com/saif01234567/Lab-Tasks-DS.git>

```
#include <iostream>

using namespace std;
```

```
class Node {
public:
    string name;
    int sapID;
    Node* next;

    Node(string name, int sapID) {
        this->name = name;
        this->sapID = sapID;
        this->next = NULL;
    }
};
```

```
class LinkedList {
    Node* head;
public:
    LinkedList() { head = NULL; }
```

```
// Insert new student at the end
```

```
void insertStudent(string name, int sapID) {
```

```

Node* newNode = new Node(name, sapID);

if (!head) {
    head = newNode;
    return;
}

Node* temp = head;
while (temp->next != NULL)
    temp = temp->next;
temp->next = newNode;
}

```

```

// Delete student at given position
void deleteStudent(int pos) {
    if (!head) {
        cout << "List is empty" << endl;
        return;
    }

    Node* temp = head;
    if (pos == 1) {
        head = head->next;
        delete temp;
        return;
    }

    Node* prev = NULL;
    while (pos-- > 1 && temp) {
        prev = temp;

```

```
        temp = temp->next;
    }
    if (temp) {
        prev->next = temp->next;
        delete temp;
    }
}
```

```
// Print all student records
```

```
void printList() {
    Node* temp = head;
    while (temp != NULL) {
        cout << "Name: " << temp->name << ", SAP ID: " << temp->sapID << endl;
        temp = temp->next;
    }
}
};
```

```
int main() {
    LinkedList list;
    string name;
    int sapID;

    // Input at least 5 students
    for (int i = 0; i < 5; ++i) {
        cout << "Enter name of student " << i + 1 << ": ";
```

```

//cin >> name; // only reads one words (Saif), not (Saif Khan..)
if (i == 0) {
    cin.ignore(); // Clear any leftover newline only before the first getline
}

getline(cin, name); // for more than one word in name..

cout << "Enter SAP ID of student " << i + 1 << ": ";

cin >> sapID;

cin.ignore(); // Clear buffer after sapID for the next name

list.insertStudent(name, sapID);
}

// Delete 2nd and 5th students

list.deleteStudent(2);

list.deleteStudent(4); // After deleting 2nd, 5th is now 4th

// Print remaining list

cout << "\nRemaining students:\n";

list.printList();

return 0;
}

```

main.cpp



Share

Run

Output

Clear

```
66 };\n67\n68 int main() {\n69     LinkedList list;\n70     string name;\n71     int sapID;\n72\n73     // Input at least 5 students\n74     for (int i = 0; i < 5; ++i) {\n75         cout << "Enter name of student " << i + 1 << ": "; \n76         //cin >> name; // only reads one words (Saif), not\n            (Saif Khan..)\n77         if (i == 0) {\n78             cin.ignore(); // Clear any leftover newline only before\n                the first getline\n79         }\n80         getline(cin, name); // for more than one word in name..\n81         cout << "Enter SAP ID of student " << i + 1 << ": "; \n82         cin >> sapID;\n83         cin.ignore(); // Clear buffer after sapID for the next\n            name\n84
```

```
/tmp/wTugEksLpc.o\nEnter name of student 1: Saif Khan\nEnter SAP ID of student 1: 57114\nEnter name of student 2: Haider Ali\nEnter SAP ID of student 2: 53109\nEnter name of student 3: Ahmed Ali\nEnter SAP ID of student 3: 55346\nEnter name of student 4: Zia\nEnter SAP ID of student 4: 54322\nEnter name of student 5: Imran\nEnter SAP ID of student 5: 64566\n\nRemaining students:\nName: aif Khan, SAP ID: 57114\nName: Ahmed Ali, SAP ID: 55346\nName: Zia, SAP ID: 54322\n\n=== Code Execution Successful ===
```