

```
#include <iostream>
#include <algorithm>
#include <vector>
```

```
using namespace std;
```

```
struct Task {
    string description; // about task
    bool completed;
};
```

```
// Function for adding the task into the to-do list
void addTask(vector<Task> &todoList, const string
&description) {
    Task newTask = {description, false};
    todoList.push_back(newTask);
    cout << "Task added: " << description << endl;
}
```

```
// Function for viewing all tasks in the to-do list
void viewTasks(const vector<Task> &todoList) {
    if (todoList.empty()) {
        cout << "No tasks is available in the to-do list." << endl;
    } else {
        cout << "To-Do List:" << endl;
        for (size_t i = 0; i < todoList.size(); ++i) {
            cout << i + 1 << ". ";
            if (todoList[i].completed) {
                cout << "[X] ";
            } else {

```

```

        cout << "[ ] ";
    }
    cout << todoList[i].description << endl;
}
}
}

```

```

// Function for deleting a task from the to-do list
void deleteTask(vector<Task> &todoList, size_t index) {
    if (index < todoList.size()) {
        cout << "Task deleted: " << todoList[index].description <<
endl;
        todoList.erase(todoList.begin() + index);
    } else {
        cout << "Invalid task index." << endl;
    }
}
}

```

```

int main() {
    vector<Task> todoList;

    cout << "To-Do List Manager" << endl;

    while (true) {
        cout << "Options:" << endl << "1. Add Task" << endl << "2.
View Tasks" << endl << "3. Delete Task" << endl << "4. Exit" <<
endl;

```

```
int option;
cout << "choose desired option: ";
cin >> option;

switch (option) {
    case 1:
    {
        cin.ignore();
        string taskDescription;
        cout << "Enter task description: ";
        getline(cin, taskDescription);
        addTask(todoList, taskDescription);
    }
    break;
    case 2:
        viewTasks(todoList);
        break;
    case 3:
    {
        size_t taskIndex;
        cout << "Enter the index of the task to delete: ";
        cin >> taskIndex;
        deleteTask(todoList, taskIndex - 1);
    }
    break;
    case 4:
        cout << "Exiting the program." << endl;
        return 0;
    default:
        cout << "you have choosen wrong option" << endl;
```

```
}
```

```
}
```

```
return 0;
```

```
}
```