Ganpat University

Faculty of Engineering & Technology

Computer Science & Engineering

Name:- ***Dwij Vatsal Desai***

Sem:- ***2***

Sub: - ***ESFP-II***

Enrollment No.:- ***23162121027***

Prac:- ***8***

Date:- ***16/4/2024***

Practical 8

Definition:

Complete the code for the object already assigned to you during practical 7 to satisfy following specifications.
This is the extended part of practical number-7, which is to be performed as per the following instructions :
1. Append all string data to first string of your object and then display only string data of all objects together (one record in one line).
2. Find the length of the string prepared in point 1, for all records and display on screen with appropriate message.
3. Display last number of characters from given string field like customer name, product name or brand, name, etc as per user choice.
4. Find first and last character of all the string fields found in point 1.
5. Extract substring from given string as per your string field available in the given record.
6. Design a search method, which asks a data to be searched based on name and other parameters of the object. Display record data if match found. Display appropriate message if match not found.
7. Display all record information/data record in ascending and descending orders based on their name field.
[Note: You must implement object-oriented concept for the practical]

Code:-

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

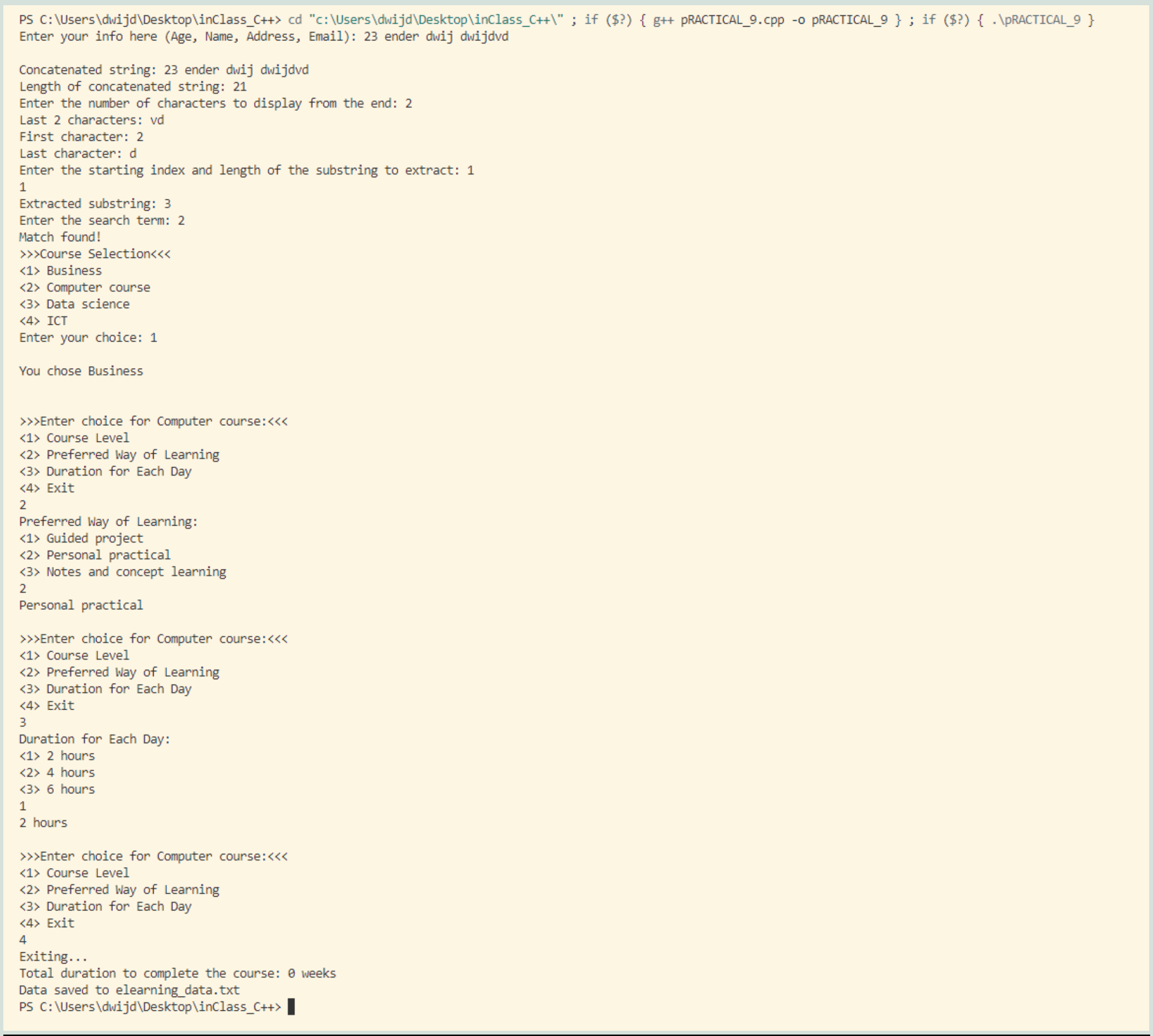
```
#include <string>

using namespace std;

void userinfo() {
    string age, name, address, email;

    cout << "Enter your info here (Age, Name, Address, Email): ";
    cin >> age >> name >> address >> email;
    cout << endl;

    // Append all strings into one
    string tinfo = age + " " + name + " " + address + " " + email;
    cout << "Concatenated string: " << tinfo << endl;

    // Display length of the concatenated string
    cout << "Length of concatenated string: " << tinfo.length() << endl;

    // Display last characters based on user choice
    int num_chars;
    cout << "Enter the number of characters to display from the end: ";
    cin >> num_chars;
    cout << "Last " << num_chars << " characters: " <<
tinfo.substr(tinfo.length() - num_chars) << endl;

    // Find and display first and last characters
    cout << "First character: " << tinfo.front() << endl;
    cout << "Last character: " << tinfo.back() << endl;

    // Extract substring based on user input
    int start_index, length;
    cout << "Enter the starting index and length of the substring to extract:
";
    cin >> start_index >> length;
    cout << "Extracted substring: " << tinfo.substr(start_index, length) <<
endl;

    // Search method
    string search_term;
    cout << "Enter the search term: ";
    cin >> search_term;
    if (tinfo.find(search_term) != string::npos) {
        cout << "Match found!" << endl;
    } else {
        cout << "No match found." << endl;
    }
}
```

```cpp
class elearning {
private:
    int expression;
    string selectedCourse;
    string courseLevel;
    string preferredWayOfLearning;
    string durationForEachDay;

public:
    elearning() {
        expression = 0;
    }

    elearning(int exp) {
        expression = exp;
        switch (expression) {
        case 1:
            cout << endl << "You chose Business" << endl << endl;
            selectedCourse = "Business";
            selectComputerCourse();
            break;
        case 2:
            cout << endl << "You chose Computer course" << endl << endl;
            selectedCourse = "Computer course";
            selectComputerCourse();
            break;
        case 3:
            cout << endl << "You chose Data science" << endl << endl;
            selectedCourse = "Data science";
            selectComputerCourse();
            break;
        case 4:
            cout << endl << "You chose ICT" << endl << endl;
            selectedCourse = "ICT";
            selectComputerCourse();
            break;
        default:
            break;
        }
    }

    void selectCourse() {
        cout << ">>>Course Selection<<<" << endl;
        cout << "<1> Business" << endl;
        cout << "<2> Computer course" << endl;
        cout << "<3> Data science" << endl;
        cout << "<4> ICT" << endl;
    }
```

```cpp
void selectComputerCourse() {
    int totalDuration = 0;
    int innerExpression = 0;
    while (true) {
        cout << endl << ">>>Enter choice for Computer course:<<<" << endl;
        cout << "<1> Course Level" << endl;
        cout << "<2> Preferred Way of Learning" << endl;
        cout << "<3> Duration for Each Day" << endl;
        cout << "<4> Exit" << endl;
        cin >> innerExpression;

        switch (innerExpression) {
        case 1: {
            int level = 0;
            cout << "Course Level:" << endl;
            cout << "<1> 1-2 weeks (basic)" << endl;
            cout << "<2> 2-5 weeks (intermediate)" << endl;
            cout << "<3> 6-7 weeks (advance)" << endl;
            cin >> level;
            switch (level) {
            case 1:
                cout << "1-2 weeks (basic)" << endl;
                courseLevel = "1-2 weeks (basic)";
                totalDuration += 2;
                break;
            case 2:
                cout << "2-5 weeks (intermediate)" << endl;
                courseLevel = "2-5 weeks (intermediate)";
                totalDuration += 5;
                break;
            case 3:
                cout << "6-7 weeks (advance)" << endl;
                courseLevel = "6-7 weeks (advance)";
                totalDuration += 7;
                break;
            default:
                cout << "Invalid choice" << endl;
                break;
            }
            break;
        }
        case 2: {
            int preferredWay = 0;
            cout << "Preferred Way of Learning:" << endl;
            cout << "<1> Guided project" << endl;
            cout << "<2> Personal practical" << endl;
            cout << "<3> Notes and concept learning" << endl;
            cin >> preferredWay;
            switch (preferredWay) {
            case 1:
                cout << "Guided project" << endl;
                preferredWayOfLearning = "Guided project";
                break;
            case 2:
                cout << "Personal practical" << endl;
                preferredWayOfLearning = "Personal practical";
                break;
            case 3:
                cout << "Notes and concept learning" << endl;
                preferredWayOfLearning = "Notes and concept learning";
                break;
            default:
                cout << "Invalid choice" << endl;
                break;
            }
            break;
        }
        case 3: {
            int duration = 0;
            cout << "Duration for Each Day:" << endl;
            cout << "<1> 2 hours" << endl;
            cout << "<2> 4 hours" << endl;
            cout << "<3> 6 hours" << endl;
            cin >> duration;
            switch (duration) {
            case 1:
                cout << "2 hours" << endl;
                durationForEachDay = "2 hours";
                break;
            case 2:
                cout << "4 hours" << endl;
                durationForEachDay = "4 hours";
                break;
            case 3:
                cout << "6 hours" << endl;
                durationForEachDay = "6 hours";
                break;
            default:
                cout << "Invalid choice" << endl;
                break;
            }
            break;
        }
        case 4:
            cout << "Exiting..." << endl;
```

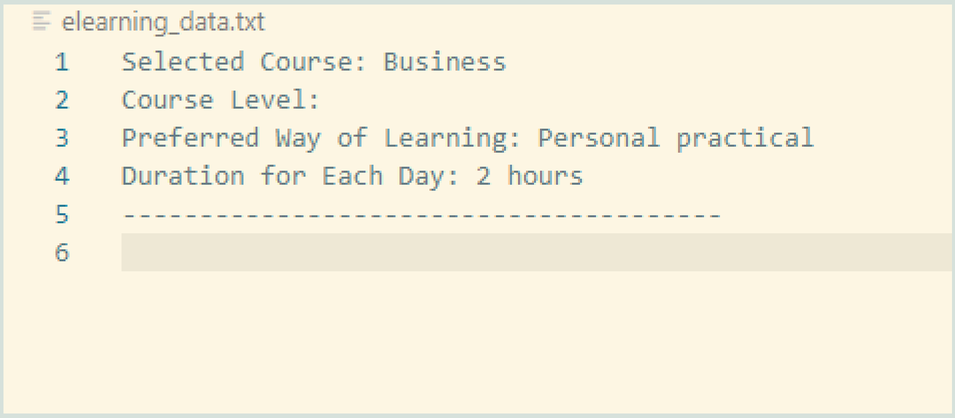
```cpp
                cout << "Total duration to complete the course: " <<
totalDuration << " weeks" << endl;
                storeDataToFile();
                return;
            default:
                cout << "Invalid choice" << endl;
                break;
            }
        }
    }

    void storeDataToFile() {
        ofstream file("elearning_data.txt", ios_base::app);
        if (file.is_open()) {
            file << "Selected Course: " << selectedCourse << endl;
            file << "Course Level: " << courseLevel << endl;
            file << "Preferred Way of Learning: " << preferredWayOfLearning <<
endl;
            file << "Duration for Each Day: " << durationForEachDay << endl;
            file << "---------------------------------------" << endl;
            file.close();
            cout << "Data saved to elearning_data.txt" << endl;
        } else {
            cout << "Unable to open file" << endl;
        }
    }
};

int main() {
    userinfo();

    elearning obj1;
    obj1.selectCourse();
    int userChoice;
    cout << "Enter your choice: ";
    cin >> userChoice;
    cin.ignore();

    elearning obj2(userChoice);

    return 0;
}
```

Output:-



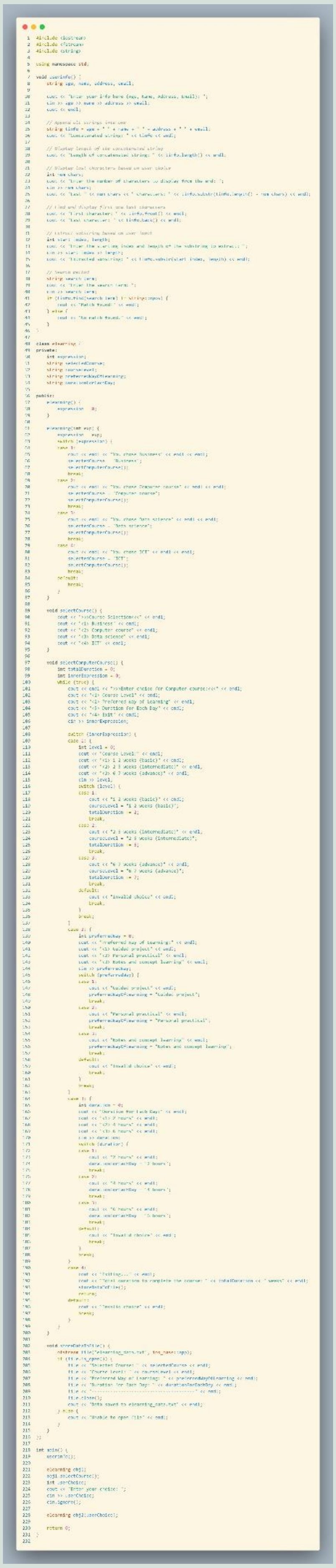
```
PS C:\Users\dwijd\Desktop\inClass_C++> cd "c:\Users\dwijd\Desktop\inClass_C++\" ; if ($?) { g++ pRACTICAL_9.cpp -o pRACTICAL_9 } ; if ($?) { .\pRACTICAL_9 }
Enter your info here (Age, Name, Address, Email): 23 ender dwij dwijdvd

Concatenated string: 23 ender dwij dwijdvd
Length of concatenated string: 21
Enter the number of characters to display from the end: 2
Last 2 characters: vd
First character: 2
Last character: d
Enter the starting index and length of the substring to extract: 1
1
Extracted substring: 3
Enter the search term: 2
Match found!
>>>Course Selection<<<
<1> Business
<2> Computer course
<3> Data science
<4> ICT
Enter your choice: 1

You chose Business


>>>Enter choice for Computer course:<<<
<1> Course Level
<2> Preferred Way of Learning
<3> Duration for Each Day
<4> Exit
2
Preferred Way of Learning:
<1> Guided project
<2> Personal practical
<3> Notes and concept learning
2
Personal practical

>>>Enter choice for Computer course:<<<
<1> Course Level
<2> Preferred Way of Learning
<3> Duration for Each Day
<4> Exit
3
Duration for Each Day:
<1> 2 hours
<2> 4 hours
<3> 6 hours
1
2 hours

>>>Enter choice for Computer course:<<<
<1> Course Level
<2> Preferred Way of Learning
<3> Duration for Each Day
<4> Exit
4
Exiting...
Total duration to complete the course: 0 weeks
Data saved to elearning_data.txt
PS C:\Users\dwijd\Desktop\inClass_C++>
```

__Recorded data in text file:-__



elearning_data.txt

```
Selected Course: Business
Course Level:
Preferred Way of Learning: Personal practical
Duration for Each Day: 2 hours
----------------------------------------

```



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

using namespace std;

void userInfo() {
    string age, name, address, email;

    cout << "Enter your info here (Age, Name, Address, Email): ";
    cin >> age >> name >> address >> email;
    cout << endl;

    // Append all strings into one
    string info = age + " " + name + " " + address + " " + email;
    cout << "Concatenated string: " << info << endl;

    // Display length of the concatenated string
    cout << "Length of concatenated string: " << info.length() << endl;

    // Display last characters based on user choice
    int num_chars;
    cout << "Enter the number of characters to display from the end: ";
    cin >> num_chars;
    cout << "Last " << num_chars << " characters: " << info.substr(info.length() - num_chars) << endl;

    // Find and display first and last characters
    cout << "First character: " << info.front() << endl;
    cout << "Last character: " << info.back() << endl;

    // Extract substring based on user input
    int start_index, length;
    cout << "Enter the starting index and length of the substring to extract: ";
    cin >> start_index >> length;
    cout << "Extracted substring: " << info.substr(start_index, length) << endl;

    // Search method
    string search_term;
    cout << "Enter the search term: ";
    cin >> search_term;
    if (info.find(search_term) != string::npos) {
        cout << "Match found!" << endl;
    } else {
        cout << "No match found." << endl;
    }
}

class elearning {
private:
    int expression;
    string selectedCourse;
    string courseLevel;
    string preferredWayOfLearning;
    string durationForEachDay;

public:
    elearning() {
        expression = 0;
    }

    elearning(int exp) {
        expression = exp;
        switch (expression) {
        case 1:
            cout << endl << "You chose Business" << endl << endl;
            selectedCourse = "Business";
            selectComputerCourse();
            break;
        case 2:
            cout << endl << "You chose Computer course" << endl << endl;
            selectedCourse = "Computer course";
            selectComputerCourse();
            break;
        case 3:
            cout << endl << "You chose Data science" << endl << endl;
            selectedCourse = "Data science";
            selectComputerCourse();
            break;
        case 4:
            cout << endl << "You chose ICT" << endl << endl;
            selectedCourse = "ICT";
            selectComputerCourse();
            break;
        default:
            break;
        }
    }

    void selectCourse() {
        cout << ">>>Course Selection<<<" << endl;
        cout << "<1> Business" << endl;
        cout << "<2> Computer course" << endl;
        cout << "<3> Data science" << endl;
        cout << "<4> ICT" << endl;
    }

    void selectComputerCourse() {
        int totalDuration = 0;
        int innerExpression = 0;
        while (true) {
            cout << endl << ">>>Enter choice for Computer course:<<<" << endl;
            cout << "<1> Course Level" << endl;
            cout << "<2> Preferred Way of Learning" << endl;
            cout << "<3> Duration for Each Day" << endl;
            cout << "<4> Exit" << endl;
            cin >> innerExpression;

            switch (innerExpression) {
            case 1: {
                int level = 0;
                cout << "Course Level:" << endl;
                cout << "<1> 1 2 weeks (basic)" << endl;
                cout << "<2> 2 5 weeks (intermediate)" << endl;
                cout << "<3> 6 7 weeks (advance)" << endl;
                cin >> level;
                switch (level) {
                case 1:
                    cout << "1 2 weeks (basic)" << endl;
                    courseLevel = "1 2 weeks (basic)";
                    totalDuration += 2;
                    break;
                case 2:
                    cout << "2 5 weeks (intermediate)" << endl;
                    courseLevel = "2 5 weeks (intermediate)";
                    totalDuration += 5;
                    break;
                case 3:
                    cout << "6 7 weeks (advance)" << endl;
                    courseLevel = "6 7 weeks (advance)";
                    totalDuration += 7;
                    break;
                default:
                    cout << "Invalid choice" << endl;
                    break;
                }
                break;
            }
            case 2: {
                int preferredWay = 0;
                cout << "Preferred Way of Learning:" << endl;
                cout << "<1> Guided project" << endl;
                cout << "<2> Personal practical" << endl;
                cout << "<3> Notes and concept learning" << endl;
                cin >> preferredWay;
                switch (preferredWay) {
                case 1:
                    cout << "Guided project" << endl;
                    preferredWayOfLearning = "Guided project";
                    break;
                case 2:
                    cout << "Personal practical" << endl;
                    preferredWayOfLearning = "Personal practical";
                    break;
                case 3:
                    cout << "Notes and concept learning" << endl;
                    preferredWayOfLearning = "Notes and concept learning";
                    break;
                default:
                    cout << "Invalid choice" << endl;
                    break;
                }
                break;
            }
            case 3: {
                int duration = 0;
                cout << "Duration for Each Day:" << endl;
                cout << "<1> 2 hours" << endl;
                cout << "<2> 4 hours" << endl;
                cout << "<3> 6 hours" << endl;
                cin >> duration;
                switch (duration) {
                case 1:
                    cout << "2 hours" << endl;
                    durationForEachDay = "2 hours";
                    break;
                case 2:
                    cout << "4 hours" << endl;
                    durationForEachDay = "4 hours";
                    break;
                case 3:
                    cout << "6 hours" << endl;
                    durationForEachDay = "6 hours";
                    break;
                default:
                    cout << "Invalid choice" << endl;
                    break;
                }
                break;
            }
            case 4:
                cout << "Exiting..." << endl;
                cout << "Total duration to complete the course: " << totalDuration << " weeks" << endl;
                storeDataToFile();
                return;
            default:
                cout << "Invalid choice" << endl;
                break;
            }
        }
    }

    void storeDataToFile() {
        ofstream file("elearning_data.txt", ios_base::app);
        if (file.is_open()) {
            file << "Selected Course: " << selectedCourse << endl;
            file << "Course Level: " << courseLevel << endl;
            file << "Preferred Way of Learning: " << preferredWayOfLearning << endl;
            file << "Duration for Each Day: " << durationForEachDay << endl;
            file << "-----------------------------------" << endl;
            file.close();
            cout << "Data saved to elearning_data.txt" << endl;
        } else {
            cout << "Unable to open file" << endl;
        }
    }
};

int main() {
    userInfo();

    elearning obj1;
    obj1.selectCourse();
    int userChoice;
    cout << "Enter your choice: ";
    cin >> userChoice;
    cin.ignore();

    elearning obj2(userChoice);

    return 0;
}
```

Photo:-