

## Chapter 10 RECORDS (Structs )

In Chapter 6 we learned that an array is set up to use multiple elements of the same data type. What if we want to have several different related types within one data structure then an array will not do. This is where we need to use a record or more generally called *struct* in C++. We can use a *class* as well which is covered in the next chapter. With object oriented programming it does not make sense to use struct any more. However, to understand existing code, one must learn about struct. To set up variables of a struct data type, one must declare a data type and then declare variables of that data type. Here is an example of declaring a struct data type. Each component of the struct is called a data members or fields. In the following example we will declare a struct data type of friendStructType.

```
struct friendStructType {  
    char name[30];  
    char address[20];  
    char city[20];  
    char state[2];  
    int zip;  
    char tele[12];  
};
```

Once the struct data type is declared, variables of this type can be made just as any other variable declaration.

```
friendStructType friend1, friend2, friend3;
```

Values can be assigned to variable friend1 by either using the assignment statement or reading into the variable. For example,

```
friend1.name = "Jose Rodriguez";  
cin >> friend.name;
```

In the following sample code I have declared to variables (myfriend1 and myfriend2) of the friendStructType.

```
int main ()  
{  
    friendStructType myfriend1, myfriend2;  
    cout << "Enter information about first friend." <<endl;  
    cout << "Name-----> ";cin.getline(myfriend1.name,30,'\n');  
    cout << "Address-----> ";cin.getline(myfriend1.address,20,'\n');  
    cout << "City-----> ";cin >> myfriend1.city;  
    cout << "State-----> ";cin >> myfriend1.state;  
    cout << "Zip-----> ";cin >> myfriend1.zip;
```

```

    cout << "Telephone-----> ";cin >> myfriend1.tele;
    cin.get();
    cout << "\nEnter information about second friend.\n";

    cout << "Name-----> ";cin.getline(myfriend2.name,30,'\n');
    cout << "Address-----> ";cin.getline (myfriend2.address,20,'\n');
    cout << "City-----> ";cin >> myfriend2.city;
    cout << "State-----> ";cin >> myfriend2.state;
    cout << "Zip-----> ";cin >> myfriend2.zip;
    cout << "Telephone-----> ";cin >> myfriend2.tele;

    cout << "\nHere is the information you entered." <<endl;
    cout << myfriend1.name <<endl;
    cout << myfriend1.address <<endl;
    cout << myfriend1.city <<endl;
    cout << myfriend1.state <<endl;
    cout << myfriend1.zip <<endl;
    cout << myfriend1.tele <<endl <<endl;
    cout << myfriend2.name <<endl;
    cout << myfriend2.address <<endl;
    cout << myfriend2.city <<endl;
    cout << myfriend2.state <<endl;
    cout << myfriend2.zip <<endl;
    cout << myfriend2.tele <<endl;
    return(0);
}

```

The following example shows how to create an array of the struct. Specifically, a struct data type friendStructType is declared and using this, another type (using typedef) arrayRecType (typedef friendStructType arrayRecType[50]) is declared which is an array of 50 individual records. Finally using the arrayRecType a variable of friends is declared (arrayRecType friends).

```

/*****
Struct 2
Written by Dr. John Abraham
for 2380 students
*****/

#include <iostream>
#include <string>
struct friendStructType {

```

```

char name[30];
char address[25];
char city[20];
char state[3];
int zip;
char tele[13];
};

typedef friendStructType arrayRecType[50];

int main ()
{
    arrayRecType friends;
    int i=1, n;
    cout << "Enter information about friends. Enter 'quit' to stop entry. " <<endl;
    cout << "Name-----> ";cin.getline(friends[i].name,30,'\n');
    while(strcmp (friends[i].name, "quit") !=0)
    {
        cout << "Address-----> ";cin.getline(friends[i].address,20,'\n');
        cout << "City-----> ";cin.getline(friends[i].city,20,'\n');
        cout << "State-----> ";cin >> friends[i].state;
        cout << "Zip-----> ";cin >> friends[i].zip;
        cout << "Telephone-----> ";cin >> friends[i].tele;
        cin.get();
        i++;
        cout <<"\n-----Record Number " << i <<"-----\n";
        cout << "Name-----> ";cin.getline(friends[i].name,30,'\n');
    }
    cout <<"Enter the record number to display (or 0 to quit) "; cin >> n;
    while (n >0 && n <i)
    {
        cout << "Name-----> " << friends[n].name;
        cout << "\nAddress-----> " << friends[n].address;
        cout << "\nCity-----> " << friends[n].city;
        cout << "\nState-----> " << friends[n].state;
        cout << "\nZip-----> " << friends[n].zip;
        cout << "\nTelephone-----> " << friends[n].tele <<endl;
        cout <<"\n-----Record Number " << n <<"-----\n";
        cout <<"Enter another record number to display (or 0 to quit)"; cin >> n;
    }

    return 0;
}

```

It is important to hide certain details when creating abstract data types. As long as a programmer knows how to use a declared data type, the details can be hidden. For example, we have been using the simple data types such as integer since the first chapter. Details of implementation of integer were left up to the developer of the language. We can also hide the details of implementation of our struct data type by creating a header file or include file. In order to create a headerfile start Microsoft Visual Studio and choose File from the dropdown menu and choose new file. From the installed templates choose C++ and click on headerfile and give it a name. Save the headerfile as given below in the directory you wish to keep it.

```

*****
Header file for Struct
Program 3
Written by Dr. John Abraham
for 2380 students
*****/
#ifndef PERSON_H //these directives prevents the preprocessor
#define PERSON_H //from including a file more than once. IF DEFINED END ELSE
DEFINE.

#include <iostream>
#include <string>
struct friendStructType {
    char name[30];
    char address[30];
    char city[20];
    char state[3];
    int zip;
    char tele[13];
};

typedef friendStructType arrayRecType[50];

//Two function PROTOYPES are included. getInfo and showInfo.
int getInfo (arrayRecType friends);
void showInfo (int, arrayRecType friends);

#endif

```

We can now write the program to use this headerfile as given below. Make sure to include appropriate directory path to the include file.

```

/*****

```

Struct 3 using includefile  
Written by Dr. John Abraham  
for 2380 students

\*\*\*\*\*/

```
#include <iostream>
#include <string>
#include "person.h"

using namespace std;
int main ()
{
    int i;
    arrayRecType friends;

    i = getInfo(friends);
    cout <<i <<"records read.\n";
    showInfo(i, friends);
    return 0;
}
int getInfo (arrayRecType friends)
{
    int i=1;
    cout << "Enter information about friends. Enter 'quit' to stop entry. " <<endl;
    cout << "Name-----> ";cin.getline(friends[i].name,30,'\n');
    while(strcmp (friends[i].name, "quit") !=0)
    {
        cout << "Address-----> ";cin.getline(friends[i].address,30,'\n');
        cout << "City-----> ";cin.getline(friends[i].city,20,'\n');
        cout << "State-----> ";cin >> friends[i].state;
        cout << "Zip-----> ";cin >> friends[i].zip;
        cout << "Telephone-----> ";cin >> friends[i].tele;
        cin.get();
        i++;
        cout <<"\n-----Record Number " << i <<"-----\n";
        cout << "Name-----> ";cin.getline(friends[i].name,30,'\n');
    }
}
return i;
}

void showInfo(int i, arrayRecType friends)
{
    int n;
    cout <<"Enter the record number to display (or 0 to quit) "; cin >> n;
    while (n >0 && n <i)
    {
```



```

int main ()
{
    int choice=0, last=0;
    arrayRecType friends;

    menu();
    cin >>choice;
    while ((choice >0) && (choice <5))
    {
        switch (choice)
        {
            case 1 : last = readInfo(friends, last);break;
            case 2 : last = getInfo(friends, last);break;
            case 3 : showInfo(friends,last);break;
            case 4 : saveInfo(friends, last);
        } //switch
        menu();
        cin >> choice;
    } //while
    return 0;
}

void menu ()
{
    cout <<"\n***** MENU *****\n";
    cout <<"Choose one of the following (or zero quit):\n";
    cout <<"1. Read from a file\n";
    cout <<"2. Enter Information\n";
    cout <<"3. Display Information\n";
    cout <<"4. Save to a file\n\n";
    cout <<"Your choice (1..4)--> ";
}

int getInfo(arrayRecType friends, int last)
{
    int i =last+1;
    cin.get();
    cout << "Enter information about friends. Enter 'quit' to stop entry. " <<endl;
    cout << "Name-----> ";cin.getline(friends[i].name,40,'\n');
    while(strcmp (friends [i].name, "quit") !=0)
    {
        cout << "Address-----> ";cin.getline(friends[i].address,40,'\n');
        cout << "City-----> ";cin.getline(friends[i].city,30,'\n');
        cout << "State-----> ";cin >> friends[i].state;
        cout << "Zip-----> ";cin >> friends[i].zip;
        cout << "Telephone-----> ";cin >> friends[i].tele;
        cin.get();
    }
}

```





```

        friendfile >>friends[i].zip ;
        friendfile >> friends[i].tele ;
        i++;
        friendfile.ignore(5,'\n');
    }
    friendfile.close();
    cout <<"\nTotal Records: "<<i-2; //reads blank space and end of file
return i-2;
}

```

```

void saveInfo (arrayRecType friends, int last)
{ int i;
  ofstream friendfile;
  friendfile.open("a:friends1.dta");
  assert(! friendfile.fail());
  for (i=1; i <= last; i++)
  {
    friendfile << friends[i].name <<"\n";
    friendfile << friends[i].address <<"\n";
    friendfile << friends[i].city << "\n";
    friendfile << friends[i].state << " ";
    friendfile << friends[i].zip << " ";
    friendfile << friends[i].tele <<"\n";
  }

  friendfile.close();
}

```