EXPERIMENT NO. 2

PROGRAM CODE:

```
#include<iostream>
using namespace std;
struct node
{
string name;
node *B[5];
};
class book
{
int c,s,sub;
public:
node *temp = new node;
void Getnewnode()
{
for(int i = 0 ; i<5 ; i++)
{
temp->B[i] = new node;
temp->B[i]->name = "empty";
for(int j=0; j<5; j++)
{
temp->B[i]->B[j] = new node;
temp->B[i]->B[j]->name = "empty";
for(int k = 0 ; k<5; k ++)
{
temp->B[i]->B[j]->B[k] = new node;
temp->B[i]->B[j]->B[k]->name = "empty";
}
}
}
}
void add_title()
{
cout<<": ENTER THE TITLE OF THE BOOK = ";
cin>>temp->name;
Getnewnode();
}
void add_chapter()
{
string cname;
int cnum;
cout<<": ENTER NUMBER OF CHAPTERS IN THE BOOK = ";
cin>>cnum;
c = cnum;
for(int i = 0 ; i < cnum ; i++)
{
cout<<endl<<"Chapter "<<i+1<<" = ";
cin>>cname;
```

```
temp->B[i]->name = cname;
}
}
void add_section()
{
string sname;
int snum;
int chnum;
cout<<": CHAPTER NUMBER WHERE YOU NEED TO ADD SECTIONS = ";
cin>>chnum;
cout<<": ENTER NUMBER OF SECTIONS = ";
cin>>snum;
s = snum;
for(int i = 0 ; i < snum ; i++)
{
cout<<endl<<"Section "<<chnum<<"."<<i+1<<" = ";
cin>>sname;
temp->B[chnum-1]->B[i]->name = sname;
}
}
void add_sub_section()
{
string subname;
int subnum;
int snum;
int chnum;
cout<<": CHAPTER NUMBER WHERE YOU NEED TO ADD SUB-SECTIONS = ";
cin>>chnum;
cout<<": SECTION NUMBER WHERE YOU NEED TO ADD SUB-SECTIONS = ";
cin>>snum;
cout<<": ENTER NUMBER OF SUB-SECTIONS = ";
cin>>subnum;
sub = subnum;
for(int i = 0 ; i < subnum ; i++)
{
cout<<endl<<"Sub-Section "<<chnum<<"."<<snum<<"."<<i+1<<" = ";
cin>>subname;
temp->B[chnum-1]->B[snum-1]->B[i]->name = subname;
}
}
void display() {
    cout << endl << "======= INDEX =======" << endl;
    cout << endl << ": TITLE : " << temp->name;
    cout << endl << ": CHAPTERS : " << endl;

    for (int i = 0; i < c; i++) {
        if (temp->B[i]->name != "empty") {
            cout << endl << ": " << i + 1 << ". " << temp->B[i]->name;
            for (int j = 0; j < s; j++) {
                if (temp->B[i]->B[j]->name != "empty") {
```

```
                    cout << endl << " : " << i + 1 << "." << j + 1 << " " <<
temp->B[i]->B[j]->name;
                    for (int k = 0; k < sub; k++) {
                        if (temp->B[i]->B[j]->B[k]->name != "empty") {
                            cout << endl << " : " << i + 1 << "." << j + 1 << "."
<< k + 1 << " " << temp->B[i]->B[j]->B[k]->name;
                        }
                    }
                }
            }
        }
    }
    cout << endl;
}
};
int main()
{
book s;
s.add_title();
s.add_chapter();
s.add_section();
s.add_sub_section();
s.display();
return 0;
}
```