LECTURE-8

Array In OOP

We can use classes to create objects, and arrays can be used to store multiple objects of the same class.

```
#include<iostream>
using namespace std;
class employee{
private:
char name[20]; int age,sal;
public:
       void getdata();
       void putdata();
};
void employee :: getdata (){
       cout<<"enter name :";
       cin>>name;
       cout<<"enter age :";
       cin>>age;
       cout<<"enter salary:";
       cin>>sal;
}
void employee :: putdata ( ){
       cout<<"----------------"<<endl;
       cout<<name <<endl;
       cout<<age<<endl;
       cout<<sal<<endl;
}
main(){
       employee emp[2];
       for( int i=0;i<2;i++)
       {
       emp[i].getdata();
       }
       cout<<endl;
       for(int j=0;j<2;j++)
       {
       emp[j].putdata();
       }
}
```

Array of objective:

```
#include<iostream>
using namespace std;
class emp
{
        char name[20]; int age,sal;
public:
        void   getdata( ); void     putdata( );
};

void   emp :: getdata( )
{
        cout<<"enter empname:";    cin>>name;
        cout<<"enter age:"<<endl;  cin>>age;
        cout<<"enter salun :";     cin>>sal;
}
void   emp :: putdata ()
{
        cout<<"emp name:"<<name<<endl;
        cout<<"emp age:"<<age<<endl;
        cout<<"emp salary:"<<sal;
}

main()
{
        emp foreman[5];     emp engineer[5];
        for(int i=0;i<5;i ++)
        {
                cout<<" for foreman:";   foreman[i].getdata();
        }
        cout<<endl;
        for(int i=0;i<5;i++)
        {
                foreman[i].putdata();
        }
        for(int i=0;i<5;i ++)
        {
                cout<<" for engineer:";   engineer[i].getdata();
        }
        for(int i=0;i<5;i++)
        {
                engineer[i].putdata();
        }
}
```

Hom Work: Replace and sort using class.

The following is the C++ program for storing data of one Employee:

```
#include<iostream>
using namespace std;
class Employee
{
  int id;
  char name[30];
  public:
  void getdata();//Declaration of function
  void putdata();//Declaration of function
};
void Employee::getdata(){//Defining of function
  cout<<"Enter Id : ";
  cin>>id;
  cout<<"Enter Name : ";
  cin>>name;
}
void Employee::putdata(){//Defining of function
  cout<<id<<" ";
  cout<<name<<" ";
  cout<<endl;
}
main(){
  Employee emp; //One member
  emp.getdata();//Accessing the function
  emp.putdata();//Accessing the function
}
```

This program can take the data of only one Employee. What if there is a requirement to add data of more than one Employee. Here comes the answer Array of Objects. An array of objects can be used if there is a need to store data of more than one employee. Below is the C++ program to implement the above approach.

```
#include<iostream>
using namespace std;
class Employee
{
        int id;
        char name[30];
        public:
                void getdata();
                void putdata();
};
void Employee::getdata()
{
        cout << "Enter Id : ";
        cin >> id;
        cout << "Enter Name : ";
        cin >> name;
}
void Employee::putdata()
{
        cout << id << " ";
        cout << name << " ";
        cout << endl;
}
int main()
{
        Employee emp[30];
        int n, i;
        cout << "Enter Number of Employees - ";
        cin >> n;
        for(i = 0; i < n; i++)
        emp[i].getdata();
        cout << "Employee Data - " << endl;
        for(i = 0; i < n; i++)
        emp[i].putdata();
}
```

Advantages of Array of Objects:

- The array of objects represents storing multiple objects in a single name.
- In an array of objects, the data can be accessed randomly by using the index number.
- Reduce the time and memory by storing the data in a single variable.

```
#include<iostream>
using namespace std;
class item
{
       char name[30];
       int price;
       public:
              void getitem();
              void printitem();
};
void item::getitem()
{
       cout << "Item Name = ";
       cin >> name;
       cout << "Price = ";
       cin >> price;
}
void item ::printitem()
{
       cout << "Name : " << name << "\n";
       cout << "Price : " << price << "\n";
}
const int size = 3;
main()
{
       item t[size];
       for(int i = 0; i < size; i++)
       {
              cout << "Item : " << (i + 1) << "\n";
              t[i].getitem();
       }
       for(int i = 0; i < size; i++)
       {
              cout << "Item Details : " << (i + 1) << "\n";
              t[i].printitem();
       }
}
```