Tutorial Link https://codequotient.com/tutorials/Pointer to objects/5b390577c6a1d0259e728f6f

TUTORIAL

Pointer to objects

Chapter

1. Pointer to objects

Pointer is an important aspect of C++ which provides a technique to handle data. It is a derived data type which stores the memory address of another variable. It is declared as

```
data-type * Pointer variable;
```

Here data-type is the valid data type in C++ like int, char etc and pointer variable is the name of the pointer. The asterisk (*) symbol distinguishes a pointer variable from other variables. Similarly a pointer can point to an object created by the class. The object pointer is declared as

```
< Class name> * Object pointer;
```

Object pointers are used in creating the objects at run time and can be used to access the public members of the class. The public member functions of the class are accessed by the syntax

```
Object pointer -> member function ( );
```

Let us take an example.

```cpp
#include<iostream>

using namespace std;

class Employee
{
  int Emp_id;
  float salary;
  public:
  void getdata( int e, float s)
  {
    Emp_id = e;
    salary = s;
  }
  void showdata ( )
  {
    cout<<"Employee id : "<<Emp_id;
    cout<<"\n Salary : Rs."<<salary;
  }
};

int main()
{
  Employee E;
  Employee *Etr = &E;
  Etr->getdata(234, 20000);
  Etr->showdata();
  return 0;
}
```

In this program, Etr is the object pointer which refers to the object E of the class Employee. The statements using arrow operator and object pointer

```
Etr -> getdata (234, 20000);
Etr -> showdata ( );
```

are equivalent to the normal access using dot operator and the object i.e.

```
E . getdata (234, 20000);
E . showdata ( );
```