

C++

C++ is an object oriented programming language. It was developed by Bjarne Stroustrup at AT &T's bell laboratory in USA in the early 1980's. Stroustrup was very strong supporter of C & Simula67. He wanted to combine the best of both languages and create a more powerful language that could support object oriented programming features and still retain the power and efficiency of C and the result was C++. Therefore, C++ is an extension of C with a major addition of the class construct feature of Simula67. Since the class was a major addition to the original C language, Stroustrup essentially called the new language C with classes however later in 1983 the name was changed to C++. The idea of C++ comes from the C increment operator ++.

The most important facilities that C++ adds into C are classes, inheritance, function overloading, operator overloading etc. These features enable creating of abstract datatypes and support polymorphism. Thereby making C++ is an object oriented language.

Advantages of object oriented programming

Object oriented programming provides many advantages to the programmer and the user. This technology solve many problems related to the software development, provides improved qualities and low cost software. Object oriented programming has the following advantages.

- *object oriented program can be easily upgraded.
- *using inheritance redundancy can be made on program code.
- *Redundant program code can be eliminated and the use of previously defined classes may be continued.
- *the technology of data hiding facilities, the programmer to design and develop secure program that do not disturb code in other parts of program.

* The encapsulation feature provided by object oriented programming language allows programmer to define the class with many functions and characteristics, only few functions are exposed to the user.

* all the object oriented languages can create extended and reusable part of program.

* object oriented programming enhances the thought process of programmer leading to rapid development of new software in short time intervals.

Syntax for class declaration:-

```
Class <classname.
```

```
{
```

```
visibility mode:
```

```
data members;
```

```
.....
```

```
Member function();
```

```
Visibility mode:
```

```
data members;
```

```
.....
```

```
Member function();
```

```
};
```

Q. write a program to enter a record of student and print the record using class and object.

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
#include<stdio.h>
```

```
Class student
```

```
{
```

Private:

```
Char nane[20];
```

```
int roll;
```

```
float m1,m2,m3,t,per;
```

public:

```
void input()
```

```
{
```

```
clrscr();
```

```
cout<<"\n enter roll";
```

```
cin>>roll;
```

```
cout<<"\n enter name of student";
```

```
gets(name);
```

```
cout<<"\n enter marks of maths";
```

```
cin>>m1;
```

```
cout<<"\n enter marks of sci";
```

```
cin>>m2;
```

```
cout<<"\n enter marks of eng";
```

```
cin>>m3;
```

```
t=m1+m2+m3;
```

```
per=t/3;
```

```
}
```

```
void output()
```

```
{
```

```
cout<<"\n name ="<<name;
```

```
cout<<"\n roll ="<<roll;
```

```
cout<<"\n total marks ="<<t;
```

```
cout<<"\n percentage ="<<per;
```

```

if(per<33)
{
cout<<"\n failed";
}
if(per>=33&&per<45)
{
cout<<"\n third";
}
if(per>=45&&per<60)
{
Cout<< "\n second";
}
If(per>=60)
{
Cout<< "\n first";
}
}
};

Void main()
{
Student s;
s.input();
s.output();
getch();
}

```

Q.write a program to enter two record of student and print the record using class and object.

```

#include<stdio.h>
#include<conio.h>
#include<stdio.h>
Class student
{
Private:
Char nane[20];
int roll;
float m1,m2,m3,t,per;
public:
void input()
{
clrscr();
cout<<"\n enter roll";
cin>>roll;
cout<<"\n enter name of student";
gets(name);
cout<<"\n enter marks of maths";
cin>>m1;
cout<<"\n enter marks of sci";
cin>>m2;
cout<<"\n enter marks of eng";
cin>>m3;
t=m1+m2+m3;
per=t/3;
}
void output()

```

```

{
cout<< "\n name ="<<name;
cout<< "\n roll="<<roll;
cout<< "\n total marks="<<t;
cout<< "\n percentage="<<per;
if(per<33)
{
cout<<"\n failed";
}
if(per>=33&&per<45)
{
cout<<"\n third";
}
if(per>=45&&per<60)
{
Cout<< "\n second";
}
If(per>=60)
{
Cout<< "\n first";
}
}
};
Void main()
{
Student s,s1;
s.input();

```

```
s1.input();
s.output();
s1.output();
getch();
}
```

PRIVATE MEMBER:-

The private member of the class is available only to the member function to their class. Private members are not accessible to function outside the class. All data members and member functions at the beginning of the class default to private. Therefore the data at the beginning of the class can not be accessed outside the class and this property is called data hiding.

Public member:-

It can be accessed from outside the class. Generally, functions are public member of the class. We should keep in mind that the private variable is not available to the calling program so we can only use the variable by calling function defined within the public part of the class.

Syntax:-

```
<return datatype> <classname>::function name(argument)
{
Statements;
.....
}
```

Q. Write a program to enter a record of employee using class and object.

Bs	ta	da	hra	tax
<5000	10%	15%	11%	0
-<10000	15%	18%	12%	5%
>=10000	20%	25%	30%	10%

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
void class
{
Private:
int en;
float bs,ta,da,hra,tax,gross,pay;
char  name[20];
public:
void input();
void output();
}
};
Void employee::input()
{
Cout<<"\n enter employee no";
cout>>en;
cout<<"\n enter employee name:";
gets(name);
cout<<"\n enter basicsalary";
cin>>bs;
if(bs<5000)
{
ta=(bs*10)/100;
da=(bs*15)/100;
hra=(bs*11)/100;

```



```

tax=0;
}
if(bs>=5000&&bs<10000)
{
ta=(bs*15)/100;
da=(bs*18)/100;
hra=(bs*12)/100;
tax=(bs*5)/100;
}
if(bs>=10000)
{
ta=(bs*20)/100;
da=(bs*25)/100;
hra=(bs*30)/100;
tax=(bs*10)/100;
}
gross=bs+ta+da+hra;
pay=gross-tax;
}
void employee::output()
{
cout<<"\n employeenumber="<<en;
cout<<"\n basic salary"<<bs;
cout<<"\n ta is"<<ta "da="<<da "hra="<<hra;
cout<<"\n tax="<<tax "gross="<<gross "payment="pay;
}
void main()

```

```

{
employee   e;
e.input();
e.output();
getch();
}

```

Q.write a program to enter a record of electric consumer and print the record using class and object . calculation should be made using following criteria.

Unit	rate/unit	stax	vat	otax
<100	2.84	10%	12%	5%
-<200	3.10	12%	14%	7%
>=200	4.10	14%	16%	9%

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
class electric
{
private:
int cno,st,end,unit;
char name[20],address[100];
float bill,tbill,stax,vat,otax;
public:
void entry();
void display();
};
void electric::entry()
{

```

```
cout<<"\n enter consumer number";
cin>>cno;
cout<<"\n enter consumer name";
gets(name);
cout<<"\n enter consumer address";
gets(address);
cout<<"\n enter starting meter reading";
cin>>st;
cout<<"\n enter last meter reading";
cin>>end;
unit=end-st;
if(unit<100)
{
bill=unit* 2.84;
stax=(bill*10)/100;
vat=(bill*12)/100;
otax=(bill*5)/100;
}
if(unit>=100&&unit<200)
{
bill= unit*3.10;
stax=(bill*12)/100;
vat=(bill*14)/100;
otax=(bill*7)/100;
}
if(unit>=200)
{
```

```
bill=unit*4.1;
stax=(bill*14)/100;
vat=(bill*16)/100;
otax=(bill*9)/100;
}
tbill=bill+stax+vat+otax;
}
void main()
{
clrscr();
electric e;
e.entry();
e.display();
getch();
}
void electric::display()
{
cout<<"\n consumer name="<<name;
cout<<"\n consumer number="<<cno;
cout<<"\n consumer address="<<address;
cout<<"\n starting meter reading="<<st;
cout<<"\n last meter reading="<<end;
cout<<"\n total unit consumed="<<unit;
cout<<"\n total bill="<<tbill;
}
```