Member function, static, constructor

1. Define a class Complex to represent a complex number with instance variables a and b to store real and imaginary parts. Also define following member functions
   a. void setData(int,int)
   b. void showData()
   c. Complex add(Complex)

```
#include <iostream>
using namespace std;

class Complex
{
private:
    int a, b;

public:
    void setData(int x, int y)
    {
        a = x;
        b = y;
    }
    void showData()
    {
        cout << a << " + " << b << "i" << endl;
    }
    Complex add(Complex C)
    {
        Complex add;
        add.a = a + C.a;
        add.b = b + C.b;
        return add;
    }
};

int main()
{
    Complex C1, C2, C3;
    C1.setData(5, 7);
    C2.setData(12, 9);
    C3 = C1.add(C2);
    C1.showData();
    C2.showData();
    C3.showData();

    return 0;
}
=============================================================================
Output:
5 + 7i
12 + 9i
17 + 16i
```

2. Define a class Time to represent a time with instance variables h,m and s to store hour, minute and second. Also define following member functions
   a. void setTime(int,int,int)
   b. void showTime()
   c. void normalize()
   d. Time add(Time)

```
#include <iostream>
#include <stdlib.h>
```

```cpp
using namespace std;

class Time
{
private:
    int h, m, s;

public:
    void setTime(int x, int y, int z)
    {
        h = abs(x);
        m = abs(y);
        s = abs(z);
    }
    void showTime()
    {
        cout << h << " : " << m << " : " << s << endl;
    }
    void normalize()
    {
        if(s >= 60)
        {
            int tempS, tempM;
            tempS = s % 60;
            tempM = s / 60;
            s = tempS;
            m = m + tempM;
        }
        if(m >= 60)
        {
            int tempM, tempH;
            tempM = m % 60;
            tempH = m / 60;
            m = tempM;
            h = h + tempH;
        }
    }
    Time add(Time T)
    {
        Time add;
        add.h = h + T.h;
        add.m = m + T.m;
        add.s = s + T.s;
        return add;
    }
};

int main()
{
    Time T1, T2, T3;
    cout<<"Before normalization"<<endl;
    T1.setTime(-22, 258, -354);
    T2.setTime(25, -954, 550);
    T3 = T1.add(T2);
    T1.showTime();
    T2.showTime();
    T3.showTime();
    cout<<"After normalization"<<endl;
    T1.normalize();
    T2.normalize();
    T3.normalize();
    T1.showTime();
    T2.showTime();
    T3.showTime();
    return 0;
}
==============================================================================
==
Output:
Before normalization
22 : 258 : 354
25 : 954 : 550
47 : 1212 : 904
After normalization
26 : 23 : 54
41 : 3 : 10
67 : 27 : 4
```

3. Define a class Cube and calculate Volume of Cube and initialize it using constructor.

```
#include <iostream>
using namespace std;

class Cube
{
private:
    int a;

public:
    Cube(int x) { a = x; }
    int cubeVolume()
    {
        return(a * a * a);
    }
};

int main()
{
    Cube c(6);
    int v;
    v = c.cubeVolume();
    cout<<"Volume of cube is " << v << " unit " << endl;
    return 0;
}
==============================================================================
Output:
Volume of cube is 216 unit
```

4. Define a class Counter and Write a program to Show Counter using Constructor.

```
#include <iostream>
using namespace std;

class Counter
{
private:
    static int count;

public:
    Counter()
    {
        count++;
        cout << "Counter : " << count << endl;
    }
};
int Counter::count;

int main()
{
    Counter c1, c2, c3;
    return 0;
}
==============================================================================
Output:
Counter : 1
Counter : 2
Counter : 3
```

5. Define a class Date and write a program to Display Date and initialize date object using Constructors.

```
#include <iostream>
using namespace std;

class Date
{
private:
    int d, m, y;

public:
    Date(int a, int b, int c)
    {
        d = a;
        m = b;
        y = c;
    }
    void displayDate()
    {
        cout << d << "/" << m << "/" << y << endl;
    }
};

int main()
{
    Date d1(5, 11, 2021), d2(12, 9, 2022);
    d1.displayDate();
    d2.displayDate();
    return 0;
}
==============================================================================
Output:
5/11/2021
12/9/2022
```

6. Define a class student and write a program to enter student details using constructor and define member function to display all the details.

```
#include <iostream>
using namespace std;

class Student
{
private:
    char name[50], contact[15], bg[4];
    int age, roll;

public:
    Student()
    {
        cout << "Enter roll number, name, age, blood group and contact number
respectively:" << endl;
        cin >> roll;
        cin.ignore();
        cin.getline(name, 50);
        cin >> age;
        cin.ignore();
        cin.getline(bg, 4);
        cin.getline(contact, 15);
    }
    void displayDetail()
    {
        cout << endl;
        cout << "Student Details" << endl;
        cout << "Roll number          :  " << roll << endl;
        cout << "Name                 :  " << name << endl;
        cout << "Age                  :  " << age << endl;
        cout << "Blood group          :  " << bg << endl;
        cout << "Contact number       :  " << contact << endl;
    }
};

int main()
{
    Student s1, s2;
    s1.displayDetail();
    s2.displayDetail();
    return 0;
}
==============================================================================
Output:
Enter roll number, name, age, blood group and contact number respectively:
1
Shekh Akhtar
26
AB+
8770356569
Enter roll number, name, age, blood group and contact number respectively:
2
Gautam Sharma
27
O-
9770117166

Student Details
Roll number          :  1
Name                 :  Shekh Akhtar
Age                  :  26
Blood group          :  AB+
Contact number       :  8770356569

Student Details
Roll number          :  2
Name                 :  Gautam Sharma
Age                  :  27
Blood group          :  O-
Contact number       :  9770117166
```

7. Define a class Box and write a program to enter length, breadth and height and initialize objects using constructor also define member functions to calculate volume of the box.

```
#include <iostream>
using namespace std;

class Box
{
private:
    int l, b, h;

public:
    Box()
    {
        cout << "Enter length, breadth and height : ";
        cin >> l >> b >> h;
    }
    int boxVolume()
    {
        return (l * b * h);
    }
};

int main()
{
    Box b1;
    int volume;
    volume = b1.boxVolume();
    cout << "Volume of the box is " << volume << " cubic unit" << endl;
    return 0;
}
==============================================================================
Output:
Enter length, breadth and height : 5 7 2
Volume of the box is 70 cubic unit
```

8. Define a class Bank and define member functions to read principal , rate of interest and year. Another member functions to calculate simple interest and display it. Initialize all details using the constructor.

```
#include <iostream>
using namespace std;

class Bank
{
private:
    float principal, roi, year, si;

public:
    Bank()
    {
        principal = 0.0;
        roi = 0.0;
        year = 0.0;
        si = 0.0;
    }
    Bank(int x, int y, int z)
    {
        principal = x;
        roi = y;
        year = z;
    }
    void simpleIntrest()
    {
        si = (principal * roi * year) / 100;
    }
    void display()
    {
        cout << "Simple Intrest : " << si << endl;
    }
};

int main()
{
    Bank b1(1000, 5, 3), b2(1000, 5, 10);
    b1.simpleIntrest();
    b1.display();
    b2.simpleIntrest();
    b2.display();
    return 0;
}
==============================================================================
Output:
Simple Intrest : 150
Simple Intrest : 500
```

9. Define a class Bill and define its member function get() to take detail of customer , calculateBill() function to calculate electricity bill using below tariff :
   a. Upto 100 unit RS. 1.20 per unit
   b. From 100 to 200 unit RS. 2 per unit
   c. Above 200 units RS. 3 per unit.

```
#include <iostream>
using namespace std;
// Defining Bill class
class Bill
{
private:
    char bp[11], name[30], contact[15];
    float total = 0.0;
    int unit = 0.0;

public:
    void calculateBill();
    void get();
    void showDetail()
    {
        cout << "Business partner number    :     " << bp << endl;
        cout << "Customer name              :     " << name << endl;
        cout << "Bill unit                  :     " << unit << endl;
        cout << "Contact number             :     " << contact << endl;
        cout << "Bill Amount                :     " << total << endl;
        cout << endl;
    }
};
// Defining get funtion to take customer detail
void Bill ::get()
{
    cout << "Enter business partner number : ";
    //cin.ignore();
    cin.getline(bp, 11);
    cout << "Enter customer name : ";
    //cin.ignore();
    cin.getline(name, 30);
    cout << "Enter bill unit : ";
    cin >> unit;
    cout << "Enter contact number : ";
    cin.ignore();
    cin.getline(contact, 15);
    cout << endl;
}
// Defining calculateBill function to calculate bill
void Bill::calculateBill()
{
    switch (unit)
    {
    case 0:
        cout << "Please enter valid input";
        break;
    case 1 ... 100:
        total = unit * 1.20;
        break;
    case 101 ... 200:
        total = unit * 2.0;
        break;
    default:
        total = unit * 3.0;
    }
}

// Defining main functon
int main()
{
    Bill c1, c2;
    float amount;
    c1.get();
    c2.get();
    c1.calculateBill();
    c2.calculateBill();
    c1.showDetail();
    c2.showDetail();
    return 0;
}
=============================================================================
Output:
Enter business partner number : 1005111036
Enter customer name : Shekh Akhtar
Enter bill unit : 105
Enter contact number : 8770356569

Enter business partner number : 100511038
Enter customer name : Gautam Sharma
Enter bill unit : 210
Enter contact number : 9770117166

Business partner number     :    1005111036
Customer name               :    Shekh Akhtar
Bill unit                   :    105
Contact number              :    8770356569
Bill Amount                 :    210

Business partner number     :    100511038
Customer name               :    Gautam Sharma
Bill unit                   :    210
Contact number              :    9770117166
Bill Amount                 :    630
```

10. Define a class StaticCount and create a static variable. Increment this variable in a function and call this 3 times and display the result.

```
#include <iostream>
using namespace std;

class StaticCount
{
private:
    static int count;

public:
    static void counter();
};
int StaticCount::count;
void StaticCount::counter()
{
    count++;
    cout << "Function call " << count << endl;
}
```

```
int main()
{
    StaticCount::counter();
    StaticCount::counter();
    StaticCount::counter();
    return 0;
}
==============================================================================
Output:
Function call 1
Function call 2
Function call 3
```