\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ASSIGNMENT 26 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

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>

#include <string.h>

using namespace std;

class complex

{

private:

   int a, b;

public:

   int setdata(int x, int y)

   {

      a = x;

      b = y;

   }

   void showdata()

   {

      cout << "real part=" << a << " imaginary part=" << b << endl;

   }

   complex add(complex c)

   {

      cout << "After addition\n";

      complex temp;

      temp.a = a + c.a;

      temp.b = b + c.b;

      return temp;

   }

};

int main()

{

   complex c1, c2, c3;

   c1.setdata(3, 4);

   c2.setdata(5, 6);

   c1.showdata();

   c2.showdata();

   c3 = c1.add(c2);

   c3.showdata();

}

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 <string.h>

using namespace std;

class complex

{

private:

   int h, m, s;

public:

   int settime(int x, int y, int z)

   {

      h = x;

      m = y;

      s = z;

   }

   void showtime()

   {

      cout << "Hour=" << h << " minutes=" << m << " Sec=" << s << endl;

   }

   complex add(complex c4)

   {

      cout << "After addition\n";

      complex temp;

      temp.h = h + c4.h;

      temp.m = m + c4.m;

      temp.s = s + c4.s;

      if (temp.s >= 60)

      {

         temp.m = temp.m + 1;

         temp.s = temp.s - 60;

      }

      if (temp.m >= 60)

      {

         temp.h = temp.h + 1;

         temp.m = temp.m - 60;

      }

      return temp;

   }

};

int main()

{

   complex t1, t2, t3;

   int hour, minutes, second, hour2, minutes2, second2;

   cout << "Enter hour minutes and seconds of both time:";

   cin >> hour >> minutes >> second;

   cin >> hour2 >> minutes2 >> second2;

   t1.settime(hour, minutes, second);

   t2.settime(hour2, minutes2, second2);

   t1.showtime();

   t2.showtime();

   t3 = t1.add(t2);

   t3.showtime();

}

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

#include <iostream>

#include <string.h>

using namespace std;

class cube

{

   private:

     int side;

   public:

     cube()

     {

      side=0;

     }

     void volume(int s)

     {

         cout<<"Volume of cube is "<<s\*s\*s;

     }

};

int main()

{

   int a;

   cube c;

   cout<<"Enter side of cube:";

   cin>>a;

   c.volume(a);

}

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

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

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>

#include <string.h>

using namespace std;

class student

{

   private:

    int roll,standard;

   public:

    student(int r, int s)

    {

      roll=r;

      standard=s;

    }

    void showdata()

    {

      cout<<"Roll number="<<roll<<" Standard="<<standard<<endl;

    }

};

int main()

{

   student s1(3,4),s2(5,6);

   s1.showdata();

   s2.showdata();

}

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

#include <iostream>

#include <string.h>

using namespace std;

class box

{  private:

    int length,weidth,hight;

   public:

    box(int l, int b,int h)

    {

      length=l;

      weidth=b;

      hight=h;

    }

    void volume()

    {

      cout<<"Volume of the box="<<length\*weidth\*hight<<endl;

    }

};

int main()

{

   box b1(4,8,8),b2(5,7,7);

   b1.volume();

   b2.volume();

}

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. Initialise all details using constructor.

#include <iostream>

#include <string.h>

using namespace std;

class bank

{

private:

   float principal, rate, year;

public:

   bank()

   {

      principal = 0;

      rate = 0;

      year = 0;

   }

   setdata(float p, float r, float y)

   {

      principal = p;

      rate = r;

      year = y;

   }

   void simpleinterest()

   {

      cout << "principal interest = " << principal \* rate \* year / 100 << endl;

   }

};

int main()

{

   bank b1,b2;

   b1.setdata(3000,4,1);

   b2.setdata(100000,3,3);

   b1.simpleinterest();

   b2.simpleinterest();

}

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 :

Upto 100 unit RS. 1.20 per unit

From 100 to 200 unit RS. 2 per unit

Above 200 units RS. 3 per unit.

#include <iostream>

#include <string.h>

using namespace std;

class bill

{

private:

   float unit;

public:

   get(float u)

   {

      unit = u;

   }

   void calculatebill()

   {

      float ebill;

      if (unit <= 100)

         ebill = unit \* 1.2;

      if (unit > 100 && unit <= 200)

         ebill = unit \* 2;

      if (unit > 200)

         ebill = unit \* 3;

      cout << "Total bill amount=" << ebill;

   }

};

int main()

{

   bill b;

   float un;

   cout << "Enter unit of bill=";

   cin >> un;

   b.get(un);

   b.calculatebill();

}

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>

#include <string.h>

using namespace std;

class staticcount

{

   private:

    static int var;

   public:

   void increment()

   {

      var++;

   }

   void show()

   {

      cout<<"Variable is "<<var<<endl;

   }

};

int staticcount:: var=10;

int main()

{

   staticcount s;

   s.increment();

   s.show();

    s.increment();

   s.show();

    s.increment();

   s.show();

}