

क्लास और ऑब्जेक्ट

पाठ्यपुस्तक के प्रश्न

वस्तुनिष्ठ प्रश्न

प्रश्न 1. एक यूजर डिफाइन्ड टाईप जो डेटा और फंक्शन को एक साथ बाँधे रखता है, उसे कहा जाता है

- (अ) ऑब्जेक्ट
- (ब) क्लास
- (स) ऐरे
- (द) पॉइन्टर

उत्तर: (ब) क्लास

प्रश्न 2. स्वतः ही क्लास के मेम्बर होते हैं

- (अ) पब्लिक
- (ब) प्राइवेट
- (स) प्रोटेक्टेड
- (द) इनमें से कोई नहीं

उत्तर: (ब) प्राइवेट

प्रश्न 3. इनमें से कौन-सा एक एक्सेस मोडिफायर है?

- (अ) public
- (ब) private
- (स) (अ) और (ब) दोनों
- (द) इनमें से कोई नहीं

उत्तर: (स) (अ) और (ब) दोनों

प्रश्न 4. इनमें से कौनसा स्टैटिक डेटा मेम्बर के संदर्भ में सत्य है?

- (अ) इसकी प्रारम्भिक वैल्यू शून्य होती है जब इसके क्लास का पहला ऑब्जेक्ट बनाया जाता है।
- (ब) केवल एक ही प्रतिलिपि इस डेटा मेम्बर की बनती है।
- (स) इसे क्लास वेरिएबल भी कहा जाता है।
- (द) उपरोक्त सभी

उत्तर: (द) उपरोक्त सभी

प्रश्न 5. इनमें से कौनसा स्टैटिक मेम्बर फंक्शन के संदर्भ में सत्य है?

- (अ) स्टैटिक कीवर्ड के साथ घोषित किया जाता है।
- (ब) केवल क्लास के दूसरे स्टैटिक डेटा मेम्बर और मेम्बर फंक्शन को एक्सेस कर सकते हैं।
- (स) इनको क्लास के नाम की सहायता से कॉल किया जाता है।
- (द) उपरोक्त सभी

उत्तर: (द) उपरोक्त सभी

प्रश्न 6. इनमें से कौनसा फ्रेंड फंक्शन के संदर्भ में सत्य है?

- (अ) इसे सामान्य फंक्शन की तरह कॉल किया जाता है।
- (ब) इसे क्लास के अन्दर कहीं भी घोषित किया जा सकता है।
- (स) सामान्यतः इसके आरग्यूमेन्ट ऑब्जेक्ट होते हैं।
- (द) उपरोक्त सभी

उत्तर: (द) उपरोक्त सभी

अतिलघु उत्तरीय प्रश्न

प्रश्न 1. क्लास किसे कहते हैं?

उत्तर- क्लास (class) एक यूजर डिफाइन्ड टाईप है जो डेटा और फंक्शन को एक साथ बाँधे रखता है। क्लास की घोषणा में इसके डेटा और मेम्बर फंक्शन की घोषणा होती है।

प्रश्न 2. ऑब्जेक्ट किसे कहते हैं?

उत्तर- Class type के variables को objects कहा जाता है। ऑब्जेक्ट के द्वारा हम क्लास के वेरिएबल्स और फंक्शन को access कर सकते हैं।

प्रश्न 3. फ्रेंड क्लास किसे कहते हैं?

उत्तर- अगर एक क्लास के सभी मेम्बर फंक्शन दूसरी क्लास में फ्रेंड घोषित कर दिये जाते हैं तब उस क्लास को फ्रेंड क्लास कहा जाता है।

लघु उत्तरीय प्रश्न

प्रश्न 1. प्राइवेट और पब्लिक एक्सेस मोडिफायर में क्या अन्तर है?

उत्तर- प्राइवेट और पब्लिक कीवर्ड्स को एक्सेस मोडिफायर्स कहा जाता है। चूंकि ये क्लास के मेम्बर को एक्सेस करने की प्रणाली को नियंत्रित करते हैं। क्लास के पब्लिक मेम्बर को क्लास के बाहर से एक्सेस किया जा सकता है। सामान्यतः क्लास के मेम्बर फंक्शन को पब्लिक अनुभाग में रखा जाता है। क्लास के प्राइवेट मेम्बर क्लास के बाहर से एक्सेस नहीं किये जा सकते हैं। यहाँ तक कि उस क्लास के ऑब्जेक्ट के द्वारा भी नहीं किये जा सकते हैं। सामान्यतः क्लास के डेटा मेम्बर को प्राइवेट अनुभाग में रखा जाता है।

प्रश्न 2. स्टैटिक डेटा मेम्बर के गुण क्या हैं?

उत्तर- क्लास के डेटा मेम्बर को स्टैटिक के रूप में भी घोषित किया जा सकता है। स्टैटिक डेटा मेम्बर के गुण इस प्रकार हैं

- इसकी प्रारम्भिक वेल्यू शून्य होती है जब इसके क्लास का पहला ऑब्जेक्ट बनाया जाता है।
- केवल एक ही प्रतिलिपि इस डेटा मेम्बर की बनती है और इसे क्लास के सभी ऑब्जेक्ट साझा करते हैं।
- चूंकि यह सम्पूर्ण क्लास के साथ जुड़ा हुआ है इसे क्लास वेरिएबल भी कहा जाता है।

प्रश्न 3. स्टैटिक मेम्बर फंक्शन के क्या गुण होते हैं?

उत्तर- स्टैटिक मेम्बर फंक्शन
स्टैटिक कीवर्ड के साथ घोषित मेम्बर फंक्शन को स्टैटिक मेम्बर फंक्शन कहा जाता है। स्टैटिक मेम्बर फंक्शन के निम्नलिखित गुण हैं।

- यह केवल क्लास के दूसरे स्टैटिक डेटा मेम्बर और मेम्बर फंक्शन को एक्सेस कर सकते हैं।
- इनको क्लास के नाम की सहायता से कॉल किया जाता है।

निबंधात्मक प्रश्न

प्रश्न 1. फ्रेंड फंक्शन किसे कहते हैं? इसे गुण लिखो।

उत्तर- फ्रेंड फंक्शन-जैसा कि हम जानते हैं कि क्लास के प्राइवेट मेम्बर क्लास के बाहर से एक्सेस नहीं किये जा सकते हैं। एक फ्रेंड फंक्शन क्लास के प्राइवेट डेटा को उस क्लास के ऑब्जेक्ट के जरिए एक्सेस कर सकते हैं। जब एक फंक्शन दो क्लासों में एक समान हो, सामान्यतः हम उस फंक्शन को दोनों के लिए फ्रेंड बना लेते हैं। इस फंक्शन को friend कीवर्ड के साथ घोषित किया जाता है। एक फ्रेंड फंक्शन की निम्नलिखित विशेषताएँ होती हैं।

- इसे सामान्य फंक्शन की तरह कॉल किया जाता है।
- क्लास के ऑब्जेक्ट की सहायता से कॉल नहीं किया जा सकता है।
- यह क्लास के मेम्बर को केवल उस क्लास के ऑब्जेक्ट की सहायता से एक्सेस कर सकता है।
- इसे क्लास के अन्दर कहीं भी घोषित किया जा सकता है।

- सामान्यतः इसके आरग्यूमेन्ट ऑब्जेक्ट होते हैं।

प्रश्न 2. एक 'Complex' क्लास बनाइए जो एक Complex नम्बर को बताता है और दो कॉम्प्लेक्स नम्बरों को जोड़ने और घटाने के लिए मेम्बर फंक्शन परिभाषित करने का प्रोग्राम लिखो।

उत्तर-

```
#include<iostream.h>
using namespace std;
class complex
{
int re, im;
public;
void get()
{
cin>>re>>im;
}
void disp ()
{
cout<<re "+" << im << "1";
}
void sum (complex, complex);
void subt (complex, complex);
};
void complex:: Sum (complex C1, Complex C2)
{
re = C1.re + C2.re;
im = C1.im + C2.im;
}
void complex:: Subt (complex C1, Complex C2)
{
re = C1.re - C2.re;
im = C1.im - C2.im;
}
int main()
{
complex C1, C2, C3, C4;
cout<<"Enter 1st complex number :";
C1.get ();
cout<< "Enter 2nd complex number :";
```

```

C2.get ();
cout<< "The 1st complex number is";
C1. disp ();
cout<< "The 2nd complex number is";
C2. disp();
C3.sum (C1, C2);
cout << "\n The resultant complex number is";
C3.disp();
C4.subt (C1, C2);
cout<<"\n The resultant complex number is";
C4.disp();
}

```

प्रश्न 3. दो क्लासों के डेटा मेम्बर की अदला-बदली करने का फ्रेंड फंक्शन की सहायता से प्रोग्राम लिखो।

उत्तर:

```

#include<iostream.h>
#include<conio.h>
class b;
class a
{
int x;
public :
void assign (int t)
{
x=t;
}
void display ()
{
cout<< "value of x is :"<< x << endl;
}
friend void swap (a &, b&);
};
class b
{
int y;
public:
void assign(int w)

```

```

{
y=w;
}
void display()
{
cout<<"value of Y is:" <<y<< endl;
}
friend void swap (a&, b&);
};
void swap (a&c, b&d)
{
int temp;
temp = c.x;
c.x = d.y;
d.y = temp;
}
int main()
{
a n;
b m;
int a, b;
cout<<"enter x =";
cin>>a;
cout <<"enter Y=";
cin>>b;
n.assign (a);
m.assign (b);
Swap (n, m) ;
n.display ();
m.display ();
getch ();
}

```

आउटपुट

enter X=4

enter Y=8

Value of X is : 8

Value of Y is: 4

अन्य महत्वपूर्ण प्रश्न

अतिलघु उत्तरीय प्रश्न

प्रश्न 1. डेटा हाइडिंग किसे कहते हैं?

उत्तर- डेटा को क्लास के प्राइवेट अनुभाग में घोषित करना डेटा हाइडिंग कहलाता है और यह ऑब्जेक्ट ऑरिएन्टेड प्रोग्रामिंग भाषा का एक महत्वपूर्ण फीचर है।

प्रश्न 2. प्राइवेट अनुभाग में और पब्लिक अनुभाग में घोषित क्लास के मेम्बर में मुख्य अन्तर बताइए।

उत्तर- क्लास के मेम्बर जिसकी घोषणा प्राइवेट अनुभाग में है उनको उसी क्लास के मेम्बर ही एक्सेस कर सकते हैं। क्लास के मेम्बर जिनकी घोषणा पब्लिक अनुभाग में हुई है उनको क्लास के बाहर से एक्सेस कर सकते हैं।

प्रश्न 3. इनलाइन फंक्शन किसे कहते हैं?

उत्तर- क्लास के अन्दर परिभाषित मेम्बर फंक्शन इनलाइन फंक्शन माने जाते हैं।

प्रश्न 4. फ्रेंड फंक्शन क्लास के प्राइवेट डाटा को किस प्रकार एक्सेस कर सकते हैं?

उत्तर- एक फ्रेंड फंक्शन क्लास के प्राइवेट डाटा को उस क्लास के आब्जेक्ट के जरिए एक्सेस कर सकते हैं।

प्रश्न 5. क्या फ्रेंड फंक्शन ऑब्जेक्ट रिटर्न कर सकते हैं?

उत्तर- हाँ, फ्रेंड फंक्शन ऑब्जेक्ट रिटर्न कर सकते हैं।

प्रश्न 6. डिरेफरेंसिंग ऑपरेटर क्या कार्य होता है?

उत्तर- डिरेफरेंसिंग ऑपरेटर (.) की सहायता से मेम्बर फंक्शन को कॉल कर सकते हैं।

प्रश्न 7. क्या क्लास के डेटा मेम्बर को स्टैटिक घोषित किया जा सकता है?

उत्तर- हाँ, क्लास के डेटा मेम्बर को स्टैटिक के रूप में भी घोषित किया जा सकता है।

प्रश्न 8. क्या फ्रेंड फंक्शन ऑब्जेक्ट रिटर्न कर सकते हैं?

उत्तर- हाँ, फ्रेंड फंक्शन ऑब्जेक्ट रिटर्न कर सकते हैं।

लघु उत्तरीय प्रश्न

प्रश्न 1. क्लास की घोषणा का प्रारूप लिखिए।

अथवा

क्लास का सिन्टेक्स लिखिए।

उत्तर-

```
Class class_name
{
private:
variable declaration ;
function declaration ;
public:
variable declaration ;
function declaration ;
};
```

प्रश्न 2. क्लास के मेम्बर को एक्सेस किस प्रकार किया जाता है? उदाहरण भी दीजिए।

उत्तर- हम केवल क्लास के पब्लिक मेम्बरस को उस क्लास के ऑब्जेक्ट के द्वारा एक्सेस कर सकते हैं। पब्लिक मेम्बर फंक्शन को एक्सेस करने का प्रारूप इस प्रकार है
object_name. function_name (arguments list);

उदाहरण के लिए
p.input (10, 20);

ऑब्जेक्ट p के द्वारा input () फंक्शन को कॉल किया गया है। स्टेटमेन्ट p.x= 10; मान्य नहीं है क्योंकि x को प्राइवेट घोषित किया गया है और इसे केवल क्लास के मेम्बर फंक्शन ही सीधे एक्सेस कर सकते हैं न कि ऑब्जेक्ट।।

प्रश्न 3. एक्सेस मोडिफायरस पर संक्षिप्त टिप्पणी लिखिए।

उत्तर- एक्सेस मोडिफायरस-public और private कीवर्ड्स को एक्सेस मोडिफायरस कहा जाता है। ये क्लास के मेम्बर को एक्सेस करने की प्रणाली को नियंत्रित करते हैं। क्लास के पब्लिक मेम्बर को क्लास के बाहर से एक्सेस किया जा सकता है। सामान्यतः क्लास के मेम्बर फंक्शन को पब्लिक अनुभाग में रखा जाता है। क्लास के प्राईवेट मेम्बर क्लास के बाहर से एक्सेस नहीं किये जा सकते हैं। यहाँ तक कि उसे क्लास के ऑब्जेक्ट के द्वारा भी नहीं किये जा सकते हैं। सामान्यतः क्लास के डेटा मेम्बर को प्राईवेट अनुभाग में रखा जाता है।

प्रश्न 4. दो नम्बरों के योग का फ्रेंड फंक्शन की सहायता से प्रोग्राम लिखिए।

उत्तर-

```
#include<iostream>
using namespace std;
class test
{
int x,y;
public:
void getdata (int a, int b)
{
x=a;
y=b;
}
friend int sum (test t);
};
int sum(test t)
{
return (t.x+t.y);
}
int main()
{
test q;
q.getdata (10, 20) ;
cout<<"Sum="<<sum (q);
return 0;
}
प्रोग्राम का आउटपुट होगा
Sum=30
```

प्रश्न 5. हम क्लास के मेम्बर का एड्रेस पॉइन्टर को किस प्रकार असाइन कर सकते हैं? उदाहरण सहित बताइए।

उत्तर- हम एक क्लास के मेम्बर का एड्रेस पॉइन्टर को निम्न प्रकार से असाइन कर सकते हैं।

```
class X
{
int a;
public:
void show();
};
```

हम क्लास X के मेम्बर a के लिए पॉइन्टर परिभाषित इस प्रकार कर सकते हैं

```
intX::*p= &X::a;
```

X : * का मतलब क्लास X के मेम्बर के लिए पॉइन्टर।

&X : * का मतलब क्लास X का मेम्बर a का एड्रेस।

स्टेटमेंट int *p = &a; कार्य नहीं करेगा। पॉइन्टर p का उपयोग डेटा मेम्बर a को मेम्बर फंक्शन और फ्रेंड फंक्शन के अन्दर एक्सेस करने के लिए कर सकते हैं।

उदाहरण के लिए

```
void show()
{
X x; //object created
cout<<x.*p; //display value of a
cout<<x.a; //same as above
}
```

हम क्लास के मेम्बर फंक्शन के लिए पॉइन्टर सेट कर सकते हैं। मेम्बर फंक्शन को डिरेफरेंसिंग ऑपरेटर (*) की सहायता से कॉल कर सकते हैं।

उदाहरण के लिए

```
Xx;//object created
void(X::*pf) ()= &X:: show;
(x.*pf)();//invoke show
यहाँ pf मेम्बर फंक्शन show () के लिए पॉइन्टर है।
```

निबंधात्मक प्रश्न

प्रश्न 1. क्लास के मेम्बर फंक्शन को कितने प्रकार से परिभाषित किया जा सकता है? उदाहरण सहित बताइए।

उत्तर- मेम्बर फंक्शन को परिभाषित करना-क्लास के मेम्बर फंक्शन को क्लास के अन्दर और क्लास से बाहर परिभाषित किया जा सकता है।

क्लास के अन्दर

मेम्बर फंक्शन की घोषणा को क्लास के अन्दर उसकी वास्तविक परिभाषा से विस्थापित किया जाता है। क्लास के अन्दर परिभाषित मेम्बर फंक्शन इनलाईन फंक्शन माने जाते हैं। उदाहरण के लिए ।

```
class point
{
int x,y;
public:
void input (int a, int b)
{
x=a;
y=b;
}
void output (void)
{
cout<<"x="<<<<"\n";
cout<<"y="<<y;
}
}
```

क्लास के बाहर

मेम्बर फंक्शन जिनकी घोषणा क्लास के अन्दर की गई हो उनको क्लास से बाहर अलग से परिभाषित करना होता है। मेम्बर फंक्शन को परिभाषित करने के लिए प्रारूप

```
return_type class_name::function_name (arguments)
{
function body
}
```

यहाँ class_name दर्शाता है कि फंक्शन इस क्लास से सम्बन्धित है।
उदाहरण के लिए

```
class point
{
int x,y;
public:
void input (int a, int b);
void output (void);
```

```

};
void point::input(int a, int b)
{
x = a;
y = b;
}
void point::output (void)
{
cout<< "x ="<< x << "\n";
cout<< "y ="<< y;
}

```

प्रश्न 2. क्या ऐरे भी क्लास के डेटा मेम्बर के रूप में हो सकते हैं? उदाहरण सहित बताइए।

अथवा

क्लास के भीतर ऐरे का प्रोग्राम लिखिए।

उत्तर- ऐरे भी क्लास के डेटा मेम्बर के रूप में हो सकते हैं।
 उदाहरण के लिए
 प्रोग्राम-क्लास के भीतर ऐरे।

```

#include<iostream>
using namespace std;
class data
{
int a = [5];
public:
void getdata (void);
void showdata (void);
};
void data::getdata (void)
{
cout<< "Enter the elements of array \n";
for(int i = 0; i <5; i++)
{
cin >> a [i];
}
}

```

```

void data::showdata (void)
{
cout<<"Array elements are \n";
for(int i = 0; i <5; i++)
cout<<a[i]<<"\t";
}
int main()
{
data d;
d.getdata();
d.showdata();
return 0;
}

```

प्रोग्राम का आउटपुट होगा

Enter the elements of array

6 5 9 8 1

Array elements are

6 5 9 8 1

प्रश्न 3. स्टैटिक मेम्बर फंक्शन का एक प्रोग्राम लिखिए।

उत्तर- प्रोग्राम-स्टैटिक मेम्बर फंक्शन

```

#include<iostream>
using namespace std;
class test
{
int x;
static int y;
public:
void set_xy(int a)
{
x = a;
y++;
}
void show_x (void)
{
cout<<"x ="<< x <<"\n";
}
}

```

```

static void show_y (void)
{
    cout<<"y ="<< y;
}
};
int test::y;
int main()
{
    test t1,t2;
    t1.set_xy(10);
    t2.set_xy (20);
    t1.show_x();
    t2.show_x();
    test::show_y(); // calling static function
    return 0;
}

```

प्रोग्राम का आउटपुट होगा

```

x = 10
x = 20
y = 2

```

प्रश्न 4. स्टेटिक डेटा मेम्बर के लिए एक प्रोग्राम लिखिए।

उत्तर- प्रोग्राम-स्टेटिक डेटा मेम्बर

```

#include<iostream>
using namespace std;
class data
{
    static int x;
    int y;
public:
    void getdata (int a)
    {
        y = a;
        x++;
    }
    void show-x(void)
    cout<<"x ="<<x <<"\n";
}

```

```

}
};
int data::x; //static member definition
int main()
{
data d1, d2; // x is initialized to zero
d1.show_x();
d2.show_x();
d1.getdata (10);
d2.getdata (20);
cout<<"After reading data"<<"\n";
d1, show_x ();
d2. show_x ();
return 0;
}

```

प्रोग्राम का आउटपुट होगा

x=0

x=0

After reading data

x=2

x=2

प्रश्न 5. फ्रेंड क्लास के लिए एक प्रोग्राम लिखिए।

अथवा

फ्रेंड क्लास का उपयोग करते हुए दो नम्बरों में से maximum नम्बर निकालने के लिए प्रोग्राम लिखिए।

उत्तर:

```

#include<iostream>
using namespace std;
class second; //forward declaration
class first
{
int x;
public:
void set_value(int a)

```

```

{
x=a;
}
friend void max (first, second);
};
class second
{
int y;
public:
void set value (int b)
{
y=b;
friend void max (first, second);
};
void max (first f, second s)
{
if (f.x>s.y)
cout<<"Maximum is"<<f.x;
else
cout<<"Maximum is"<<s.y;
}
int main()
{
first A;
second B;
A.set value (10);
B.set value (20);
max (A,B);//calling friend function
return 0
}
प्रोग्राम का आउटपुट होगा
Maximum is 20

```

प्रश्न 6. रिटर्निंग ऑब्जेक्ट के लिए एक प्रोग्राम लिखिए।

अथवा

दो `vectors` को जोड़ने के लिए रिटर्निंग ऑब्जेक्ट का उपयोग करते हुए प्रोग्राम लिखिए।

उत्तर:

```
#include<iostream>
using namespace std;
class vector
{
int V[3];
public:
void set_vector (void)
{
cout<<"Enter three numbers\n";
for (int i=0;i<3; i++)
cin>>V[i];
}
void display (void)
{
for (int i = 0; i<3; i++)
cout<<V[i]<<" ";
}
friend vector sum (vector, vector);
};
vector sum (vector P, vector q)
{
vector r;
for int j= 0; j<3; j++)
r.V[j]=p.V[j]+q.V[j];
return r;
}
int main()
{
vector v1, v2, v3;
v1.set_vector();
v2.set vector();
v3=sum (v1, v2);
cout<<"First vector is:";
v1.display();
cout<<"\n";
cout>>"Second vector is:";
v2.display();
cout<<"\n";
```

```
cout<<"Resultant vector is:";
v3.display();
return();
}
```

प्रोग्राम का आउटपुट होगा

Enter three numbers

3 -2 5

Enter three numbers

-8 6 7

First vector is: 3,-2, 5,

Second vector is: -8,6,7,

Resultant vector is:-5,4,12