Name: Vishwas M	SRN: PES2UG20CS390	Section: F
	Date:29-09-2021	Week Number:4

1 Write a program to show the different uses of scope resolution operator.

```
#include  <iostream>
using namespace std;
int a = 20;
int main()
{
    int a = 10;
    cout << "Value of local  a: " << a << endl;
    cout << "Value of global a: " << ::a << endl;
    return 0;
}
```

Output Screenshot:

```
PS D:\vishwas\sem3\C++\week4> cd "d:\vishwas\sem3\C++\week4\" ; if ($?) { g++ week4ques1.cpp -o week4ques1 } ; if ($?) { .\week4ques1
Value of local  a: 10
Value of global a: 20
PS D:\vishwas\sem3\C++\week4>
```

2 Write a program to show the working of a constant member function.

Program:

```
#include<iostream>
using namespace std;
class Demo
{
   int val;
   public:
   Demo(int x = 0)
   {
      val = x;
   }
```

```
    int getValue() const {
        return val;
    }
};
int main()
{
    const Demo d(28);
    Demo d1(8);
    cout << "The value using object 1 : " << d.getValue();
    cout << "\nThe value using object 2 : " << d1.getValue();
    return 0;
}
```

Output screenshot:

```
PS D:\vishwas\sem3\C++\week4> cd "d:\vishwas\sem3\C++\week4\" ; if ($?) { g++ week4ques2.cpp -o week4ques2 } ; if ($?) { .\week4ques2
The value using object 1 : 28
The value using object 2 : 8
PS D:\vishwas\sem3\C++\week4>
```

3 Write a program to write the working of a static and non static variable.

```
#include<iostream>
using namespace std;
int funcnostatic(int)
{
    int sum = 0;
    sum = sum + 10;
    return sum;
}
int funcstatic(int)
{
    static int sum = 0;
    sum = sum + 10;
    return sum;
}
int main(void)
{
    int r = 5, s;
    cout<<"Without static keyword\n";
    s = funcnostatic(r);
    cout<<"1st time function call, s = "<<s<<endl;
    s = funcnostatic(r);
    cout<<"2nd time function call, s = "<<s<<endl;
    s = funcnostatic(r);
    cout<<"3rd time function call, s = "<<s<<endl;
    cout<<"With static keyword\n";
    s = funcstatic(r);
    cout<<"1st time function call, s = "<<s<<endl;
    s = funcstatic(r);
    cout<<"2nd time function call, s = "<<s<<endl;
    s = funcstatic(r);
    cout<<"3rd time function call, s = "<<s<<endl;
    return 0;
}
```

Screenshot:

```
PS D:\vishwas\sem3\C++\week4> cd "d:\vishwas\sem3\C++\week4\" ; if ($?) { g++ tempCodeRunnerFile.cpp -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Without static keyword
1st time function call, s = 10
2nd time function call, s = 10
3rd time function call, s = 10
With static keyword
1st time function call, s = 10
2nd time function call, s = 20
3rd time function call, s = 30
PS D:\vishwas\sem3\C++\week4>
```

4 Write a program to show the working of different working specifiers.

Program:

```
#include<iostream>
using namespace std;
class baseclass
{
    private:
        int s;
    protected:
        int t;
    public:
        int u;
    baseclass()
    {
        s = 11;
        t = 12;
        u = 13;
    }
};
class deriveclass: private baseclass
{
    public:
        void show ()
        {
            cout << "s is not accessible";
            cout << "\nt is " << t;
            cout << "\nu is " << u;
        }
};
int main()
{
    deriveclass l;
    l.show();
    return 0;
}
```

SCREENSHOT:

```
PS D:\vishwas\sem3\C++\week4> cd "d:\vishwas\sem3\C++\week4\" ; if ($?) { g++ week4ques4.cpp -o week4ques4 } ; if ($?) { .\week4ques4 }
s is not accessible
t is 12
u is 13
PS D:\vishwas\sem3\C++\week4>
```

5 Write a program to show the working of an object as a pointer.

Program:

```
#include<iostream>
#include<conio.h>
using namespace std;

class student
{
    char name[100];
    int age,roll;
    float percent;
    public:
        void getdata()
        {
            cout<<"Enter data"<<endl;
            cout<<"Name:";
            cin>>name;
            cout<<"Age:";
            cin>>age;
            cout<<"Roll:";
            cin>>roll;
            cout<<"Percent:";
            cin>>percent;
            cout<<endl;
        }
        student &max(student &s1,student &s2)
        {
            if(percent>s1.percent && percent>s2.percent)
                return *this;
            else if(s1.percent>percent && s1.percent>s2.percent)
                return s1;
            else if(s2.percent>percent && s2.percent>s1.percent)
                return s2;
        }
        void display()
        {
            cout<<"Name:"<<name<<endl;
            cout<<"Age:"<<age<<endl;
            cout<<"Roll:"<<roll<<endl;
```

```
            cout<<"Percent:"<<percent;
        }
};
int main()
{
    student s,s1,s2,s3;
    s1.getdata();
    s2.getdata();
    s3.getdata();
    s=s3.max(s1,s2);
    cout<<"Student with highest percentage"<<endl;
    s.display();
    getch();
    return 0;
}
```

Output screenshot:

```
PS D:\vishwas\sem3\C++\week4> cd "d:\vishwas\sem3\C++\week4\" ; if ($?) { g++ week4ques5.cpp -o week4ques5 } ; if ($?) { .\week4ques5 }
Enter data
Name:vishwas
Age:20
Roll:390
Percent:97

Enter data
Name:vishnu
Age:398
Roll:222
Percent:96

Enter data
Name:raghav
Age:18
Roll:375
Percent:95

Student with highest percentage
Name:vishwas
Age:20
Roll:390
Percent:97
```