



Sir Syed CASE
Institute of Technology

OOP Lab #5

SUBMITTED BY:

UBAID AHMAD

ROLL NO:

2410-0011

SUBMITTED TO:

Mam Laiba Tanveer

DATE:

21/10/2025

Example 1 Static variable

Code:

```
#include <iostream>
using namespace std;
void demo(){
    static int count=0; //when the function is called
                        //for the second time
                        //it is not initialized with zero again because
                        // it is static and retains it's value
    cout<<count<<" ";
    count++;
}

void normal_demo(){
    int count=0;
    cout<<count<<" ";
    count++;
}

int main(){
    cout<<"static display:"<<endl;
    for(int i=0;i<5;i++){
        demo();
    }
    cout<<"\nnormal display:"<<endl;
    for(int i=0;i<5;i++){
        normal_demo();
    }
}
```

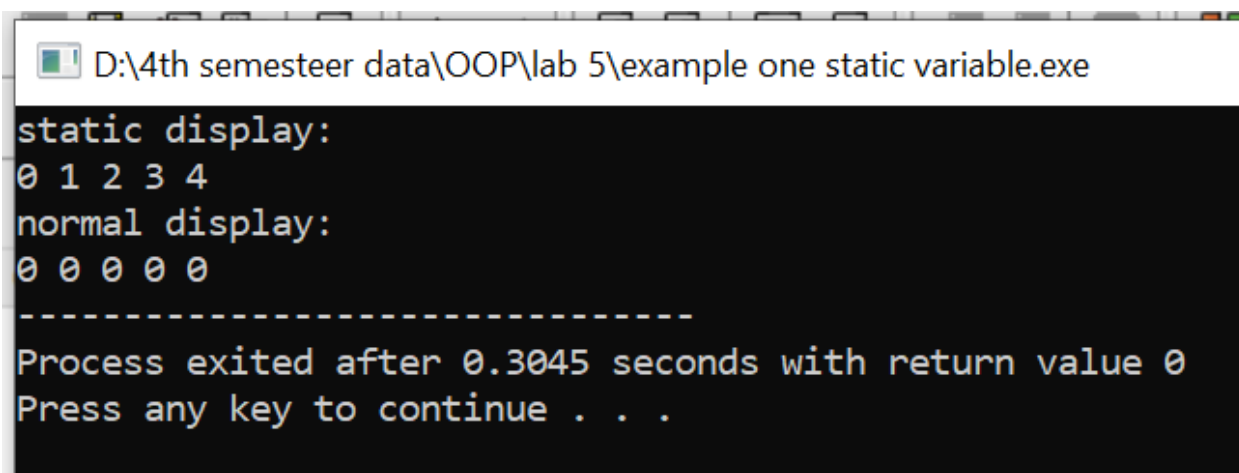
```

    }

    return 0;
}

```

Output:



```

D:\4th semesteeer data\OOP\lab 5\example one static variable.exe
static display:
0 1 2 3 4
normal display:
0 0 0 0 0
-----
Process exited after 0.3045 seconds with return value 0
Press any key to continue . . .

```

Example 2 Static Function

Code:

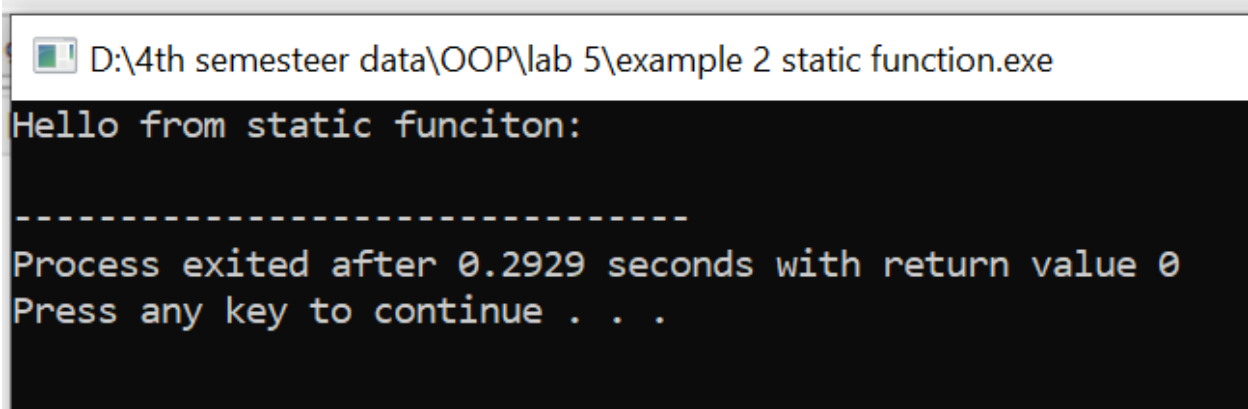
```

#include <iostream>
using namespace std;
static void greet(){
    cout<<"Hello from static funciton:"<<endl;
}
void callGreet(){
    greet();
}
int main(){
    callGreet();
    return 0;
}

```

```
}
```

Output:



```
D:\4th semester data\OOP\lab 5\example 2 static function.exe
Hello from static funciton:
-----
Process exited after 0.2929 seconds with return value 0
Press any key to continue . . .
```

Example 3 const keyword

Code:

```
#include <iostream>
using namespace std;
class myclass{
    public:
        int value;
        myclass(int v){
            value=v;
        }

        void display(){
            cout<<"value is: "<<value<<endl;
        }

        void modify(int v){
            value=v;
        }
}
```

```

        }
};

int main(){
    const myclass object(10);
    //object.display();
    //cannot call display function
    //until is declared const

    //trying to modify the value

    //    object.modifiy(20);
    //
    //    object.display();

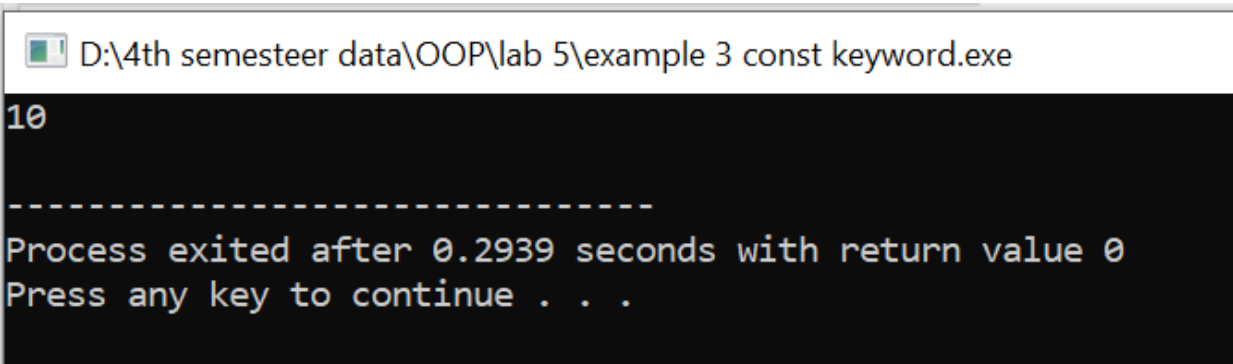
    cout<<object.value<<endl;

    return 0;

}

```

Output:



```

D:\4th semesteer data\OOP\lab 5\example 3 const keyword.exe
10
-----
Process exited after 0.2939 seconds with return value 0
Press any key to continue . . .

```

Example 4 Mutable keyword

Code:

```
#include <iostream>

using namespace std;

class myclass{
    public:
        int variable1;
        mutable int variable2;
        myclass(int v1,int v2){
            variable1=v1;
            variable2=v2;
        }

        void setVariable2(int z)const{
            variable2=z;
        }

        void display()const{
            cout<<"variable1 is: "<<variable1<<" variable2 is:
"<<variable2<<endl;
        }

};

int main(){
    const myclass object(10,20);
    cout<<"before changing variable 2 value"<<endl;
    object.display(); //can display beause display funciton is const
```

```
object.setVariable2(30);
```

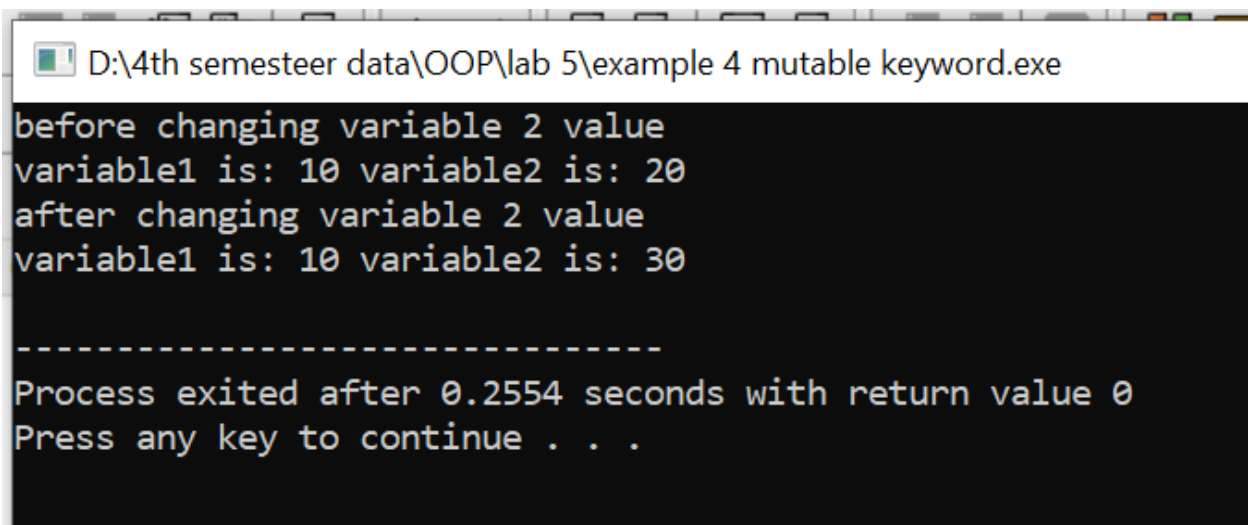
```
cout<<"after changing variable 2 value "<<endl;
```

```
object.display(); //it can work because variable 2 is mutable
```

```
return 0;
```

```
}
```

OUTPUT:



```
D:\4th semester data\OOP\lab 5\example 4 mutable keyword.exe
before changing variable 2 value
variable1 is: 10 variable2 is: 20
after changing variable 2 value
variable1 is: 10 variable2 is: 30
-----
Process exited after 0.2554 seconds with return value 0
Press any key to continue . . .
```

Task 1 BANKACCOUNT

CODE:

```
#include <iostream>
using namespace std;
class BankAccount{
```

```
int accountNo;
double balance;

public:
    static int accountCount;
    BankAccount(int number,double amount){

        accountNo=number;
        balance=amount;
        accountCount++;
    }

    void deposit(double amount){
        this->balance+=amount;
    }

    void withdraw(double amount){
        this->balance-=amount;
    }

    double getBalance()const{
        return this->balance;
    }

    static int getAccountCount(){
        return accountCount;
    }

};
```



```
int BankAccount::accountCount=0;

int main(){
    BankAccount acc(1234,1250.00);

    cout<<"Initially"<<endl;
    double balance=acc.getBalance();
    cout<<"Balance is: "<<balance<<endl;

    acc.deposit(500.00);

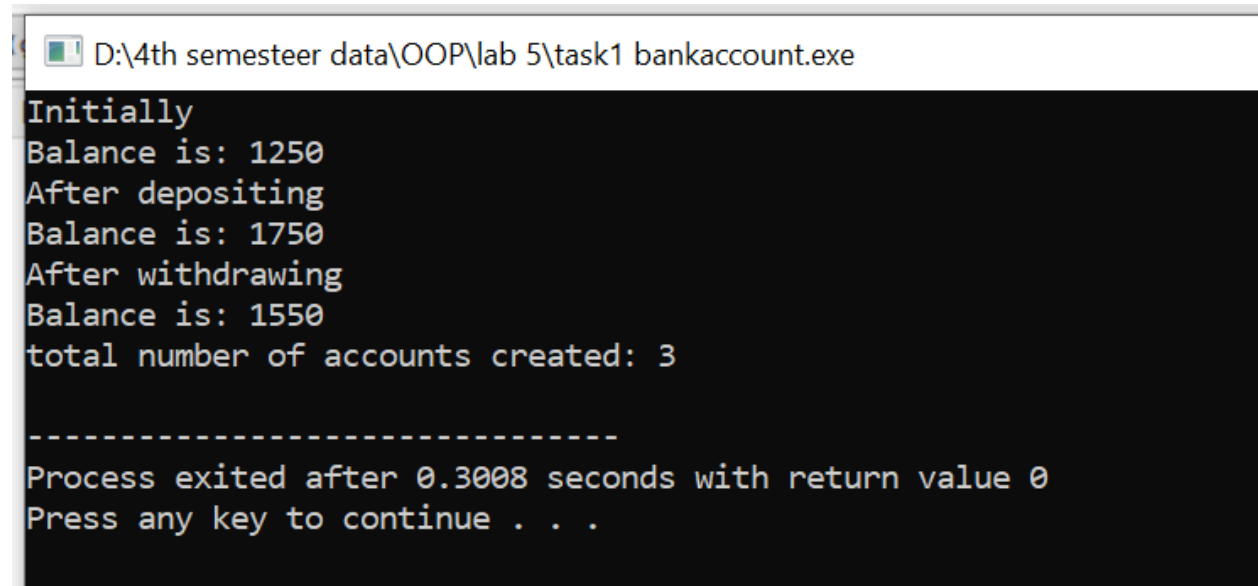
    cout<<"After depositing"<<endl;
    balance=acc.getBalance();
    cout<<"Balance is: "<<balance<<endl;

    acc.withdraw(200.00);
    cout<<"After withdrawing"<<endl;
    balance=acc.getBalance();
    cout<<"Balance is: "<<balance<<endl;

    //creating more accounts
    BankAccount acc2(5678,3000.0);
    BankAccount acc3(5678,4000.0);
    int count=acc.getAccountCount();
    cout<<"total number of accounts created: "<<count<<endl;
    return 0;
```

```
}
```

OUTPUT:



```
D:\4th semester data\OOP\lab 5\task1 bankaccount.exe
Initially
Balance is: 1250
After depositing
Balance is: 1750
After withdrawing
Balance is: 1550
total number of accounts created: 3

-----
Process exited after 0.3008 seconds with return value 0
Press any key to continue . . .
```

TASK 2 Hotel Order:

Code:

```
#include <iostream>
#include <string>
using namespace std;
const int minutesPassed=9;
class HotelOrder {
    string name;
    mutable string placedorder;
    int tableno;
    mutable int bill;

public:

    HotelOrder(string n, string order, int t, int b) {
```

```
    name = n;
    placedorder = order;
    tableno = t;
    bill = b;
}

void showOrder() const {
    cout << "Customer Name: " << name
        << "\nTable No: " << tableno
        << "\nPlaced Order: " << placedorder
        << "\nBill: " << bill << endl;
}

void changeOrder(string newOrder, int newBill) const {
    if (minutesPassed <= 10) { // allowed to change order
        placedorder = newOrder;
        bill = newBill;
        cout << "Order changed successfully within " << minutesPassed << "
minutes.\n";
    } else {
        cout << "Sorry! Order change time expired (" << minutesPassed << " minutes
passed).\n";
    }
}

};

int main() {
    const HotelOrder customer("Ubaid", "polao", 12, 700);
```

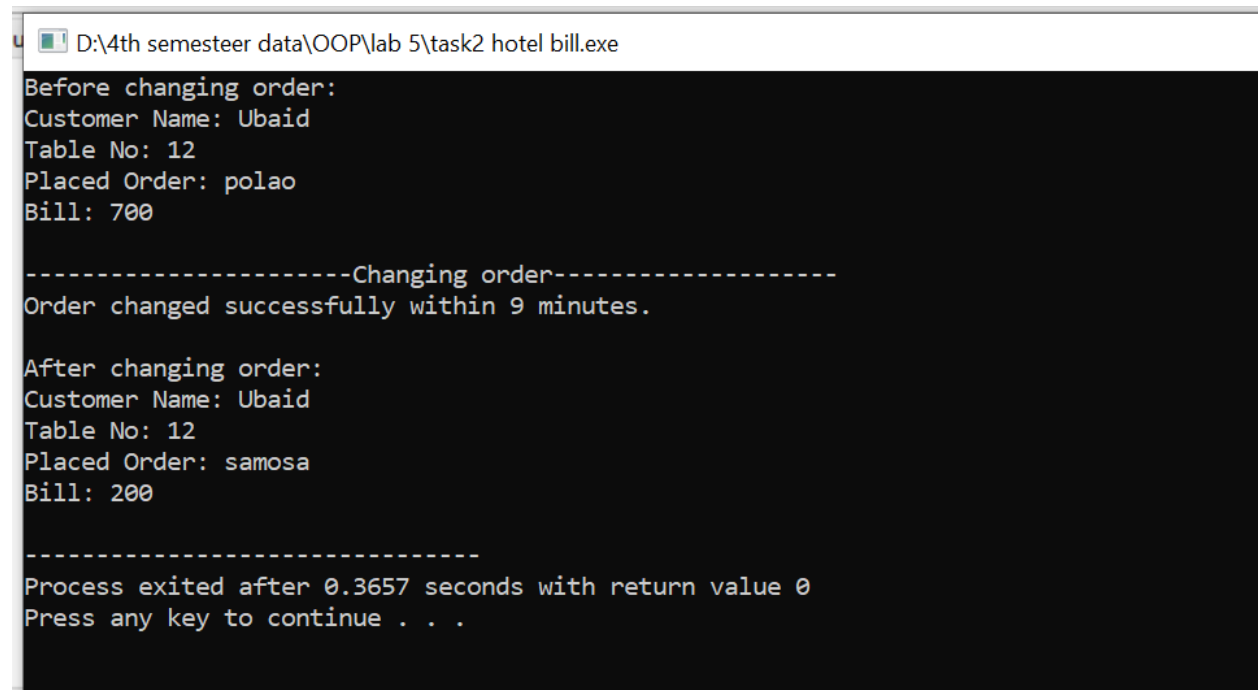
```
cout << "Before changing order:\n";
customer.showOrder();

cout << "\n-----Changing order-----\n";
customer.changeOrder("samosa", 200);

cout << "\nAfter changing order:\n";
customer.showOrder();

return 0;
}
```

Output:



```
D:\4th semesteer data\OOP\lab 5\task2 hotel bill.exe
Before changing order:
Customer Name: Ubaid
Table No: 12
Placed Order: polao
Bill: 700

-----Changing order-----
Order changed successfully within 9 minutes.

After changing order:
Customer Name: Ubaid
Table No: 12
Placed Order: samosa
Bill: 200

-----
Process exited after 0.3657 seconds with return value 0
Press any key to continue . . .
```

END,,,