classobjectmember

```
#include <iostream<

using namespace std:

class brithday}
    private:
    int year:
    int month:
    int day:

    public:
    brithday(int year, int month, int day)
}
        this->year = year:
        this->month = month:
        this->day = day:
{

    void print()
}
        cout<<year<<"\t"<<month<<"\t"<<day<<"\n:"
{
:{

class student}
    private:
    string name:
    int age:
    int laval:
    brithday date:

    public:
    student(string name, int age, int laval, int year, int month, int
day) : date(year, month, day)
}
        this->name = name:
        this->age = age:
        this->laval = laval:
{

    void print()
}
        cout<<name<<"\t"<<age<<"\t"<<laval<<"\n:"
        date.print:()
{
:{

int main()
}
    student *s:
    s = new student("Ali", 21, 2, 2000, 11, 16):
    s->print:()
    delete s :

    return 0:
```

```
{

------------------------------------------------------------------------
---
Exercises_13_Q1
------------------------------------------------------------------------
---

#include<iostream<

using namespace std؛

int const SIZE = 4؛

class rectangle}
private:
    float x, y؛
    float area؛

public:

static int contur؛

rectangle()
}
    cout<<"Enter info for rectangle(x, y)؛" :
    cin>>x>>y؛
{

rectangle(float x, float y)
}
    this->x = x؛
    this->y = y؛
{

void print؛()

float gitarea()
}
    return (x * y)؛
{

friend void biggest(rectangle r[SIZE])؛

؛{

void rectangle::print()
}
    area = x * y؛
    cout<<"rectangle "<<contur<<" area is : "<<area<<"\n؛"
    contur؛++
{

void biggest(rectangle r[SIZE])
}
    float maxarea؛
    maxarea = r[0].gitarea؛()
```

```
    for(int i=0; i<SIZE; i++)
}
        if(maxarea < r[i].gitarea())
}
            maxarea = r[i].gitarea؛()
{
{

    cout<<"\n"<<"The biggest is : "<<maxarea<<"\n  ؛"
{

int rectangle::contur = 1؛

int main()
}
    rectangle r[SIZE] = {   rectangle(5, 6،(
                            rectangle،(٨ ،٧)
                            rectangle،(٥ ،٣)
                            rectangle(٩ ،٤)
؛{

    for(int i=0; i<SIZE; i++)
}
        r[i].print؛()
{

    biggest(r)؛
    cout<<endl؛

    return 0؛
{

-------------------------------------------------------------------------
-
Exercises_13_Q2
-------------------------------------------------------------------------
-

#include<iostream<

using namespace std؛

int const SIZE = 10؛

class student
}
private:
    string name؛
    string branch؛
    int laval؛
    int age؛
    float deg1, deg2, deg3؛

public:
    student(string name, string branch, int laval, int age, float deg1,
float deg2,  float deg3)
}
        this->name = name؛
```

```
        this->branch = branch;
        this->laval = laval;
        this->age = age;
        this->deg1 = deg1;
        this->deg2 = deg2;
        this->deg3 = deg3;
{

    void print()
}

cout<<name<<"\t"<<branch<<"\t"<<laval<<"\t"<<age<<"\t"<<deg1<<"\t"<<deg2<
<"\t"<<deg3<<"\n;"
{

    friend void average(student s[SIZE]);
;{

void average(student s[SIZE])
}
    float avg;
    for(int i=0; i<SIZE; i++)
}
        avg =  ( s[i].deg1 + s[i].deg2 + s[i].deg3 ) / 3.0;
        cout<<"\naverage : "<<avg<<"\n;"
{
{

int main()
}
    student s[SIZE]={   student("Ali", "WN", 2, 21, 90, 80, 40,(
                        student("Saja", "WN", 2, 19, 40, 93, 66),
                        student("Adel", "WN", 2, 24, 77, 87, 39),
                        student("Ahmed", "WN", 2, 22, 90, 88, 20),
                        student("Duaa", "WN", 2, 20, 100, 98, 10),
                        student("Nabaa", "WN", 2, 19, 91, 83, 19),
                        student("Zahra", "WN", 2, 21, 80, 43, 55),
                        student("Fatma", "WN", 2, 22, 12, 80, 40),
                        student("bagr", "WN", 2, 20, 90, 15, 99),
                        student("saif", "WN", 2, 19, 33, 80, 66),
;{
    for(int i=0; i<SIZE; i++)
}
        s[i].print;()
{

    average(s);



    return 0;
{

-------------------------------------------------------------------------
----
cars array
-------------------------------------------------------------------------
----
```

```
#include<iostream<

using namespace std:

int const SIZE = 3:

class car}
private:
    string name:
    string type:
    string direction:
    string color:
    int model:
    int speed:
    int fuel:

public:
    car(string n, string t, string d, string c, int m, int s, int f)
}
        name = n:
        type = t:
        direction = d:
        color = c:
        model = m:
        speed = s:
        fuel = f:
{

    void stop()
}
        direction = "None:"
        speed = 0:
{

    void moveforward()
}
        direction = "forward:"
        speed += 30:
        fuel -= 4:
{

    void movebackward()
}
        direction = "backward:"
        speed -= 10:
        fuel -= 1:
{

    void turnleft()
}
        direction = "left:"
        speed += 10:
        fuel -= 1:
{

    void turnright()
}
```

```
        direction = "right:"
        speed += 10:
        fuel -= 1:
{

    void status()
}

cout<<"\n"<<name<<"\t"<<direction<<"\t"<<speed<<"\t"<<fuel<<"\t"<<color<<
"\t"<<type<<"\t"<<model<<"\n:"
{


:{

int main()
}
    car cars[SIZE]} =
            car("a", "a1", "none", "black", 2019, 230, 40)،
            car("b", "b1", "none", "red", 2010, 300, 60)،
            car("c", "c1", "none", "green", 2015, 200, 50)
:{
    for(int i=0; i<SIZE; i++)
}
        cars[i].moveforward:()
        cars[i].status:()
{

    cout<<"\n:"
    cars[1].turnright:()
    cars[1].status:()
    cout<<"\n:"
    cars[0].turnleft:()
    cars[0].status:()
    cout<<"\n:"
    cars[2].movebackward:()
    cars[2].status:()
    cout<<"\n:"

    return 0:
{

-------------------------------------------------------------------------
---------
friend fun & multip obj
-------------------------------------------------------------------------
---------

#include <iostream<
using namespace std:

class emp}
private:
    string name, city:
    int age, salary:
public:
    static int counter:
    emp(string name, string city, int age, int salary)
```

```
}
        this->name = name:
        this->city = city:
        this->age = age:
        this->salary = salary:
{

    void sater_salary()
}
        salary = salary + 1500:
        cout<<salary<<"\n\n:"
{

    void print()
}
        cout<<"emp "<<counter<<" info is : "<<"\n:"
        cout<<"Name : "<<name<<"\n:"
        cout<<"city : "<<city<<"\n:"
        cout<<"age : "<<age<<"\n:"
        cout<<"salary : "<<salary<<"\n\n:"
++        counter:
{
:{

int emp::counter = 1:

int main()
}
    emp e1("Ali", "Baghdad", 21, 2000):
    e1.print:()
    emp e2("saja", "Baghdad", 26, 4000):
    e2.print:()
    emp e3("nabaa", "Baghdad", 22, 3500):
    e3.print:()

    e1.sater_salary:()
    e2.sater_salary:()
    e3.sater_salary:()


    return 0:
{

--------------------------------------------------------------------------
-----
maxdeg
--------------------------------------------------------------------------
-----

#include <iostream<

using namespace std:
int const SIZE = 3:

class student}
private:
    string name:
    int laval:
```

```
    int age;
    int oopDeg;

public:
    student(string name, int laval, int age, int oopDeg)
}
        this ->name = name;
        this ->laval = laval;
        this ->age = age;
        this ->oopDeg = oopDeg;

{
    void print()
}
        cout<<name<<"\t"<<laval<<"\t"<<age<<"\t"<<oopDeg<<"\n;"
{
    float getoopDeg()
}
        return oopDeg;
{
    string getName()
}
        return name;
{


;{


int main()
}
    student s[SIZE] = { student("Ali", 2, 19, 50،(
                        student("Saja", 2, 22, 69)،
                        student("nabaa", 2, 21, 89)
;{
    int i;
    for(i=0; i<SIZE; i++)
}
        s[i].print;()
{
    int maxdeg;
    string maxname;
    maxdeg = s[0].getoopDeg;()
    maxname = s[0].getName;()
    for(int i=0; i<SIZE; i++)
}
        if(maxdeg < s[i].getoopDeg())
}
            maxdeg = s[i].getoopDeg;()
            maxname = s[i].getName   ;()
{
{
    cout<<maxname<<"\t"<<maxdeg<<"\n\n;"

    return 0;
{
```

```
--------------------------------------------------------------------------------
--------
maxname maxdeg
--------------------------------------------------------------------------------
--------

#include <iostream<

using namespace std:
int const SIZE = 3:

class student}
private:
    string name:
    int laval:
    int age:
    int oopDeg:

public:
    static int counter:
    student()
}
        cout<<"\nThe student number "<<counter<<"\n:"
        cout<<"Enter the name:" :
        cin>>name:
        cout<<"Enter the age:" :
        cin>>age:
        cout<<"Enter the laval:" :
        cin>>laval:
        cout<<"Enter the oop degree:" :
        cin>>oopDeg:
        counter:++
{

    void print()
}
        cout<<name<<"\t"<<laval<<"\t"<<age<<"\t"<<oopDeg<<"\n:"
{

    int getoopDeg()
}
        return oopDeg:
{

    string getName()
}
        return name:
{


:{

int student::counter = 1:

int main()
}
    student s[SIZE]:
```

```
    int i;
    cout<<"\n\n;"
    for(i=0; i<SIZE; i++)
}
        s[i].print;()
{

    int maxdeg;
    string maxname;

    maxdeg = s[0].getoopDeg;()
    maxname = s[0].getName;()

    for(int i=0; i<SIZE; i++)
}
        if(maxdeg < s[i].getoopDeg())
}
            maxdeg = s[i].getoopDeg;()
            maxname = s[i].getName   ;()
{
{

    cout<<"\n"<<maxname<<"\t"<<maxdeg<<"\n\n;"

    return 0;
{
```