

Статик члннк

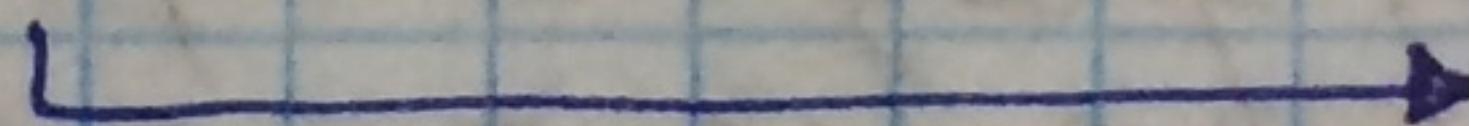
- Классы члннк овороге статик члннк тмнн бий
- Статик овороге бол члннк обекттэй холбогдуулж, классы овороги.
- Статик оворогийн буйх обект гүнгээ зарчмынхээгээ броихийн чуулзжээ.

Члннк оворог:

```
class class-name
{
```

```
private :
```

```
    data-members;
    static int number;
```



хувьсар

тирецүүр ыг

```
public :
```

```
    member-functions;
```

```
}
```

```
int class-name:: numbers = 0; → разрешение  
имени
```

Example:

```
class employee
{
```

```
private :
```

```
    static int number;
```

```
    char name [13];
```

```
    int basicpay;
```

```
    int allowance;
```

```
public :
```

```
employee();
~employee();
employee (char *n, int b, int a);
employee (employee &e);
void show number();
```

}

```
int employee::number = 0;
```

```
employee::employee()
```

{

```
strcpy (name, "Hello c++");
basicpay = allowance = 0;
number ++;
```

}

```
employee:: employee (char *n, int b, int a)
```

{

```
number++;
strcpy (name, kn);
basicpay = b;
allowance = a;
```

}

```
employee:: emp'syee (employee &e)
```

{

```
strcpy (name, e.name);
basicpay = e.basicpay;
allowance = e.allowance;
number ++;
```

}

```
employee:: ~ employee ()
```

{

```
cout << "\n Bye , bye , " << name << "
number--;
```

```
cout << number << " object left behind"
```

{

```
void main() {
    clrscr();
    employee e1("Bush", 100, 100);
    e1.show-number(); //A
    employee e2("Bill", 200, 200);
    cout << endl;
    e1.show-number(); //A
    e2.show-number();
    employee e3("Jack", 300, 300);
    cout << endl;
    e1.show-number();
    e2.show-number();
    e3.show-number();

    employee e = e1;
    cout << endl;
    e1.show-number(); //B
    e2.show-number();
    e3.show-number();
    e.show-number();

    e3 = e2;
    cout << endl;

    e1.show-number(); //C
    e2.show-number();
    e3.show-number();
    e.show-number();
    cout << endl;
    getch();
}
```

## Result:

Number : 1 }

Number : 2 }

Number : 2 }

A

Number : 3 }

— ↗ : 3

— ↗ : 3

— ↗ : 4

— ↗ : 4

— ↗ : 4

— ↗ : 4

B

Number : 4 }

— ↗ : 4

— ↗ : 4

— ↗ : 4

C.

Bye , bye Bush 3 object left behind

— ↗ Bill 2 — —

— ↗ Bill 1 — —

— ↗ Bush 0 — —

## СТАТИК ГҮШҮҮН ФУНКЦИЯЛЫГА

- Классын чиңүүн функция статик чиңүүнөө барык.
- Түхәйдін классын бүх объекттогү заралындашып да.
- Чүрөг зөвхөн статик орнадык болсодорууллаш
- Классын са статик бүй чиңүүгүн хангам тагажчы, this хуббасар нэг хангам тагажчы.

• static түркүүлүр үчүн өп-шеш  
төгөлдөр толвонуң көрсөткүү экспорталык.

• static түркүүлүр үчүн зобхон функциялар  
төгөлдөрүнүүн жотап өп-биз үзүүлэргээ  
экспорталык.

### EXAMPLE :

```
class employee
{
    private:
        static int number;
        char name[13];
        int basicpay;
        int allowance;
    public:
        employee();
        ~employee();
        employee (char *n, int b, int a);
        employee (employee &e);
        static void show_number();
};
```

### DEFINITION:

```
void employee :: show_number() // valid.
{
    - - - - -
    - - - - -
}
```

### СТАТИК ӨЧҮНКҮСҮҮР ӨГҮҮЛЭХ:

```
employee e1("Bush", 100, 100);
```

```
> e1. show_number();
```

```
employee e2("Bill", 200, 200);
```

```
> e1. show_number();
```

```
> e2. show_number();
```

```

employee e1("Bush", 100, 100);
> employee :: show_number();
employee e2("Bill", 200, 200);
> employee :: show_number();

```

## Объекттын ҳағылышы

Объекттөн қалған күйсемдердин ганашуулар шицүүлүк орунчы рүү ҳаңгак.

- ▶ Тиңүүчүнчүлөштүүн `**` оператору  
хөрөнгөк
 

```
(*pptr).getdata();
(*pptr).showdata();
```
- ▶ Тиңүүчүнчүлөштүүн `**` оператору  
 $pptr \rightarrow getdata();$   
 $pptr \rightarrow showdata();$

```
void main()
{
```

```

clrscr();
employee emp;
employee *pptr;
pptr = &emp;
pptr → getdata();
heading();
pptr → showdata();
getch();
}
```

---

```

Enter the Employee Name : George { ppt
--> Basic Pay : 9000 { getda
--> Allowance : 2000

```

Employee Name	Basic	Allowance	Total
George	9000	2000	11000

```

employee *pptr = new employee;
employee *pptr = new employee ("George",
9000, 0);
pptr -> showdata();
delete pptr;

```

## NEW / DELETE OPERATOR BA KLAZZ

```

void main()
{
    clrscr();
    employee *pptr;
    pptr = new employee; → Нэрчүү  
объект
    pptr -> getdata();
    employee *pptr1 = new employee ("Bill",
                                    8000,
                                    5000);
    heading();
    pptr -> showdata();
    pptr -> Showdata();
    delete pptr;
    delete pptra;
    getch();
}

```

Enter the Employee Name : George  
 Basic Pay : 9000  
 Allowance : 2000 → pptr -> getdata()

Employee	Name	Basic	Allowance	Gross
	George	9000	2000	11000
	Bill	8000	5000	13000

→ pptr -> showdata();  
pptr1 -> showdata();

## "this" күбесен.

- Түшүүчүн өр-өнүк экспрессион  $\delta / \text{raa}$  бөлөктөөн xаруулж халж
- Уни this күбесен бол обзектөн xаруу күбесен жум.
- C++ командалар нь
  - Түшүүчүн өртүнүү this күбесен таңын өртүнүүт
  - Түшүүчүн өр-өнүк  $\text{this} - g$  жум
  - Дүйнегесин өртүнүү this күбесен жум.
- void main()
  - {
    - clr scr();
    - employee emp1, emp2;
    - emp1. showdata();
    - emp2. showdata();
    - getch();

The object emp 2's address is  
The object emp2's address is  
0x 8F80FFed ←  
0x 8F80F7d2 ←

this Dar gamep /this/g expres  
кеэгээ

```

void employee:: getdata()
{
    cout << "Enter the Employee Name: ";
    gets(name);
    cout << "Enter the Basic Pay: ";
    cin >> basicpay;
    cout << "Enter the Allowance: ";
    cin >> allowance;
}

```

Число 1 this -> / xangax xpsomx

```

void employee:: showdata()
{
    cout << setw(13) << this->name;
    cout << setw(8) << this->basicpay;
    cout << setw(12) << this->allowance;
}

```

this xpsomx  
xpsomx

ХАРДЫН ХҮВЬСАГЧИЙН  
КЛАССЫН КЭРӨГЛӨХ.

```

class employee
{
private:
    char* name;
    -----
    -----
};

```

- Одборын дэвшигийн яг new операторын сууригт

- Одборын ярийн яг delete

## операторы си' возвратов

- new оператором бывαι дай результат  
гетуп храних
  - delete оператором генерир го.  
тот храних
  - employee (char \*n, int b, int a); оружие  
name - оворгава сяяк си' хвбсах  
ханнин.
- employee :: employee (char \*n, int b,  
int a)
- {
- ```
name = new char [strlen(n) + 1];
strcpy (name, n);
basepay = b;
allowance = a;
```
- }
- Уничтожение оружия происходит name - идти  
занесение о/сан си' возвратов
- employee :: ~employee ()
- {
- ```
cout << "Bye, bye." << name << endl;
delete name;
```
- }
- EXAMPLE:

```
setName (char *n) {
```

```
delete name;
```

```
name = new char [strlen (n) + 1];
strcpy (name, n);
```

```
}
```

Классы и обьекты 3 из 20

- > employee e1("Bill", 9000, 8000); //напишем
- //или базируя нараас
- > char \* name = "Bill";  
employee (name, 9000, 8000);
- > char \* name[] = "Bill";  
employee (name, 9000, 8000); }  
~~...!~~

Харалт орлоголтой  
хийбөгтэй чуулж  
зорчим!

Класс допор харалт орлогол хэрэгж  
чадац:

- тийхэнд орлогол нь обьектийн  
хагнах / new оператораар багасныг  
басин.

- Чүүний хамгийн харалт орлогол  
хагнанар гана.

- Объектийн хувьг онгос обьектоор  
үзүүлэх болох & боломж бий.
  - Employee e1 = e2; /граданыг  
үзүүлэх
  - e1 = e2. /үзүүлэх болох.
- Граданыг үзүүлэх ондох нь умса  
ондохоец эхийнтай боловсог обьект  
чуулж чадац нь удаа хийнгүйнэ.

- Их обьектыг онгос обьектоор  
граданыг үзүүлэх ондох тохиолдог  
басижүүлэх ғүүүдэгж -  
чадац.

Бүрчилгээний эхийн дараалуулалт хөсөн -  
ишиг хүргична.

- Удаах аргын хувьгийг 2 объект  
коопинг хүргчадаа чигдэг хийнгүй.
- Хоёр объектийн ишүүгүүг огогон  
бүрчилгээний харгалзухаа ишиг хүргична.

### EXAMPLE:

```
void main() {  
    employee e1("Bill", 9000, 800);  
    e1.showdata();  
    employee e2("George", 9000, 700);  
    e2.showdata();  
    employee e = e1; // Angela яганс.  
    e.showdata();  
}
```

### Илигээ:

- Хүрглэж болиж нутгийн нь объектийн  
хүргэхийн эмчилж эхийн дээр огог.

```
employee :: employee (const employee &e)  
{
```

~~delete name;~~

```
name = new char [strlen(e.name)+1];  
strcpy(name, e.name);  
basicpay = e.basicpay;  
allowance = e.allowance;  
}
```

- "Hello, e++" мөр бүхийд ойр зөвлөх
- new операторын ойр багасгыг  
түүнч e1 объектийн ишиг name огогийг  
ишиг нь багасгахийн ойн үзүүн хүүх'

Мы узрим фрагмент программы на языке C++

XY<sup>Y</sup>Λ AΓY      Гайдуляр / COPY /  
CONSTRUCTOR /  
Объекты      атрибуты  
Объекты      атрибуты  
class employee  
{

private :

```
char name [20];  
int basic_pay;  
int allowance;
```

public :

```
employee (char n [] = "George Bush",  
          int b = 9000, int a = 2000)
```

-----  
-----

}  
void change basic();

}

class Employee {

private :

-----  
-----

public :

```
Employee (char n [] = "George Bush",  
          int b = 9000, int a = 2000)
```

{

-----

}

Employee ( Employee & emp)

{  
cout < "Copy constructor invoke  
strcpy (name, emp.name);  
.basicpay = emp.basicpay;  
allowance = emp.allowance;

}

};

---

class Employee {

private:

---

public:

Employee ( char nc = "George  
Bush", int b = 900,  
int a = 2000 )

{

---

}

Employee ( Employee & )

};

~~with~~ Employee:: Employee ( Employee &emp)

{

---

## Хүрээрэв байгуулалт:

- employee emp(emp1),  
→ зин  $\hookrightarrow$  Employee (Employee3)

**ХАЯГАН ХУВЬСАГЧ БА  
БАЙГУУЛАРЧ.**

- Классу  $x$ нан хувьсарж ашигладыг бий.
- Самах сийг үр дүртэй яхин байгуулжсан түүдгүй ашигланад.

### EXAMPLE:

```
Employee :: Employee ( char *n, int b, int a)
```

```
{
    name = new char [strlen(n)+1];
    strcpy(name, n);
    basicpay = b;
    allowance = a;
}
```

```
Employee :: ~Employee () {
    cout << *~*name < endl;
    delete name;
}
```

Employee3 = e1; зин үүссээ  $e_1, e_3$   
 Яаралт объект болгоц. ~~зарим~~  
 үүссээ нь объектын төслийн төслийн.

## EXAMPLE

```
Employee :: Employee (const Employee &src)
{
    name = new char[strlen(src.name)];
    strcpy(name, src.name);
    basicpay = src.basicpay;
    allowance = src.allowance;
}

L new copy function for copy operation
Employee :: copy ( const Employee &src )
{
```

delete name; // removes old memory