

Объект ба классын хэрэглээ болон ажилчин классыг тодорхойлох (Лаборатори №3)

Ү.Мэндсайхан

МУИС, ХШУИС, Компьютерын ухаан, mendsaikhan@gurvanshiidel.mn

1. ОРШИЛ

Програмчлалын C++ хэл дээр Ажилчин класс тодорхойлох. Түүнд гарааны утга оноох, гараас утга авах, мэдээлэл дэлгэцлэх, цалин бодох, захирлын цалин бодох, ажилласан цаг нэмэгдүүлэх зэрэг функцүүдийг тодорхойлж, ажиллуулаад, асуултуудад хариулна.

2. ЗОРИЛГО

2.1 Класс гэж юу болох, онцлог, үүргийн талаар бичих

2.2 Класс болон C++ объектын ялгааг тайлбарлах

2.3 Гишүүн функц, гишүүн өгөгдлийн хамаарал, өгөгдлийн битүүмжлэлийн талаар бичих

2.4 Классын гишүүн өгөгдөл болон гишүүн функцэд хэрхэн хандахыг судлах

2.5 Ажилчин гэсэн класс тодорхойлох

3. ОНОЛЫН СУДАЛГАА

3.1 Класс гэж юу болох, онцлог, үүргийн талаар бич.

Класс гэдэг нь объектыг загварчилдаг үүсмэл өгөгдлийн төрөл юм. Онцлог нь амьдрал дээр байгаа ямарваа нэгэн зүйлийг хийсвэрээр тодорхойлж ашигладаг. Мөн класс нь програмчлалыг хялбарчлах үүрэгтэй. Класс нь объектод зориулсан загвар бөгөөд объект үүсгэхэд загвар болгон ашигладаг. Класс нь гишүүн өгөгдөл болон гишүүн функцээс бүрддэг. Класс нь бодит объектын үйл хөдлөлийг функцээр, шинж чанарыг өгөгдлөөр тодорхойлдог.

3.2 Класс болон C++ объектын ялгаа?

Объект нь өөрийн гэсэн шинж чанар болон үйл хөдлөлтэй зүйл юм. Бидний амьдралын эргэн тойронд байгаа зүйлс бүгд объект юм. Объектыг програмчлалаар үүсгэхийн тулд класс ашигладаг. Классаас үүссэн хувьсагчийг объект байгуулах гэнэ. Объект нь үйл хөдлөл, шинж чанартай зүйлийг хэлдэг. Мөн объект нь класс дотор байгуулагдана. Класс нь объектыг агуулж буй нэг төрлийн үүсмэл өгөгдлийн төрөл юм. Класст тодорхойлогдсон өгөгдөлд хандаж функцийг ажиллуулахын тулд объектыг үүсгэдэг.

3.3 Гишүүн функц, гишүүн өгөгдөл хоёр ямар хамааралтай байдаг вэ? Өгөгдлийн битүүмжлэл гэж юу вэ?

Гишүүн функц нь объектын үйл хөдлөл, гишүүн өгөгдөл нь объектын шинж чанар юм. Гишүүн функц нь классынхаа өгөгдөл дээр боловсруулалт хийн программын бусад функцуудтэй мэдээлэл солилцдог. Гишүүн өгөгдөл зарлагдахдаа хүрээгээрээ тухайн классаа хамардаг учир зарлагдсан гишүүн функц дотор гишүүн өгөгдлүүдийг ашиглаж болдог. Объектын гишүүн өгөгдөл болон гишүүн функцүүд нэг хүрээнд байж бусад объекттой хамааралгүй байх ойлголтыг өгөгдлийн битүүмжлэлт гэнэ. Объектын өгөгдөлд тухайн классын биш өөр функцээс хандах боломжгүй. Объект дотор гишүүн хувьсагч зарлахдаа өгөгдлийн битүүмжлэлийг зарлаж өгдөг. Public, Protected, Private гэсэн гурван P битүүмжлэлийн аль нэгийг сонгож өгдөг. Энэ нь тухайн хувьсагчид хаанаас хандах боломжтой, боломжгүйг зааж өгнө.

3.4 Классын гишүүн өгөгдөл болон гишүүн функцэд хэрхэн хандах вэ?

3.4.1 Гишүүн өгөгдөлд хандах

Гишүүн өгөгдөлд гишүүн функцээр дамжуулан ханддаг. Яагаад гэвэл классын өгөгдлүүд private шинжтэй байдаг учир классын бус функцийн хувьд хандаж болохгүй. Хэрэв классын өгөгдөл нь public түвшинд тодорхойлогдсон бол шууд объектоор нь дамжин хандах боломжтой.

3.4.2 Гишүүн функцэд хандах

Классын гишүүн функц рүү объектоор нь дамжуулан хадах учир гишүүн функцийн нэрээс шууд сонголтын “.” оператораар зааглан бичнэ.

4. ХЭРЭГЖҮҮЛЭЛТ БОЛОН ҮР ДҮН

4.1 Ажилчин класс тодорхойлох

Ажилчин класс тодорхойлсон нь:

```
class Worker {  
    // Гишүүн өгөгдөл зарлах  
private:  
    int id;           // Дугаар  
    char name[20];    // Нэр  
    char position[10]; // Албан тушаал  
    float workedHours; // Ажилласан цаг  
};
```

4.2 Гарааны утга оноох

Объект зарлагдах үед доорх функц ажиллан ажилчны дугаар, нэр, албан тушаал, ажилласан цагд гарааны утга автоматаар онооно.

```
Worker() {  
    id = 0;  
    strcpy(name, "");  
    strcpy(position, "Ажилчин");  
    workedHours = 0;  
};
```

4.3 Гараас утга авах

Ажилчны дугаар, нэр, албан тушаал, ажилласан цагийг гараас авна.

```
void get_data() {  
    cout << "Ажилчны дугаар оруулна уу?" << endl;  
    cin >> id;  
    cout << "Ажилчны нэр оруулна уу?" << endl;  
    cin >> name;  
    cout << "Ажилчны албан тушаал оруулна уу?" << endl;  
    cin >> position;  
    cout << "Ажилчны ажилласан цагийг оруулна уу?" << endl;  
    cin >> workedHours;  
};
```

4.4 Мэдээлэл дэлгэцлэх

Ажилчны дугаар, нэр, албан тушаал, ажилласан цагийг дэлгэцэд хэвлэнэ.

```
void put_data() {  
    cout << "Ажилчны дугаар: " << id << endl;  
    cout << "Ажилчны нэр: " << name << endl;  
    cout << "Ажилчны албан тушаал: " << position << endl;  
    cout << "Ажилчны ажилласан цаг: " << workedHours << endl;  
};
```

4.2 Цалин бодох

Цалин бодохдоо ажилласан цагийг 200 гаар үржүүлж бодсон. Хэрвээ албан тушаал нь “Захирал” бол захирлын цалин бодох функц ажиллаад цалинг буцаана.

```
float calc_salary() {  
    float salary = workedHours * 200;  
    if (position == "Zahiral") {  
        salary = calc_ceo_salary(salary);  
    }  
    return salary;  
};
```

4.2 Захирлын цалин бодох

Захирлын цалин бодохдоо үндсэн цалин дээр нэмэлтээр 5000-г нэмж буцаана.

```
float calc_ceo_salary(float n) {  
    return n += 5000;  
};
```

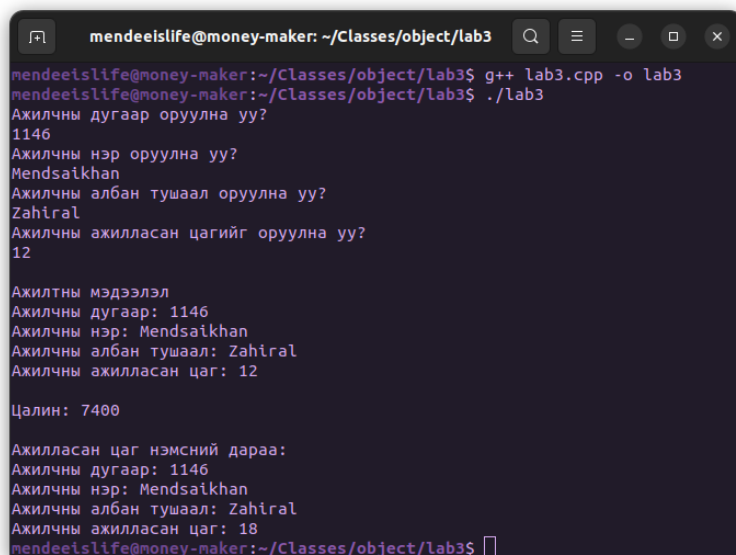
4.2 Ажилласан цаг нэмэх

Хэрвээ нэмэх цаг нь 0-24 хооронд байвал цагийг нэмээд true буюу 1 гэсэн утга буцаана.

Эсрэгээрээ – цаг юм уу 24+ цаг байвал цагийг нэмэхгүйгээр false буюу 0 гэсэн утга буцаана.

```
bool add_workedHours(float hour) {  
    if (hour > 0 && hour <= 24) {  
        workedHours += hour;  
        return true;  
    } else  
        return false;  
};
```

Үр дүн:



```
mendeislife@money-maker: ~/Classes/object/lab3  
mendeislife@money-maker:~/Classes/object/lab3$ g++ lab3.cpp -o lab3  
mendeislife@money-maker:~/Classes/object/lab3$ ./lab3  
Ажилчны дугаар оруулна уу?  
1146  
Ажилчны нэр оруулна уу?  
Mendsaikhan  
Ажилчны албан тушаал оруулна уу?  
Zahiral  
Ажилчны ажилласан цагийг оруулна уу?  
12  
  
Ажилтны мэдээлэл  
Ажилчны дугаар: 1146  
Ажилчны нэр: Mendsaikhan  
Ажилчны албан тушаал: Zahiral  
Ажилчны ажилласан цаг: 12  
  
Цалин: 7400  
  
Ажилласан цаг нэмсний дараа:  
Ажилчны дугаар: 1146  
Ажилчны нэр: Mendsaikhan  
Ажилчны албан тушаал: Zahiral  
Ажилчны ажилласан цаг: 18  
mendeislife@money-maker:~/Classes/object/lab3$
```

5. ДҮГНЭЛТ

Хэрэглэгч өөрийн тодорхойлсон классаа ашиглах нь олон хувьсагч зарлахаас сэргийлж бичсэн код нь илүү хялбар ойлгомжтой болдог бөгөөд объект загварчлаад түүнийгээ давтан ашиглах нь олон мөр код хэмнэнэ. Гишүүн өгөгдөл, функцийг зарлахдаа өгөгдлийн битүүмжлэл ашигласнаар санамсаргүй өөрчлөлтөөс сэргийлнэ.

6. АШИГЛАСАН МАТЕРИАЛ

1. Объект хандлагат технологийн C++ програмчлал, Ж.Пүрэв, 2008, Улаанбаатар

7. ХАВСРАЛТ

Github линк, [Lab 3](#)

```
#include <string.h>
#include <iostream>

using namespace std;

class Worker {
    // Гишүүн өгөгдөл зарлах
private:
    int id; // Дугаар
    char name[20]; // Нэр
    char position[10]; // Албан тушаал
    float workedHours; // Ажилласан цаг
    // Гишүүн функц зарлах
public:
    Worker() {
        id = 0;
        strcpy(name, "");
        strcpy(position, "Ажилчин");
        workedHours = 0;
    };

    void get_data() {
        cout << "Ажилчны дугаар оруулна уу?" << endl;
        cin >> id;
        cout << "Ажилчны нэр оруулна уу?" << endl;
        cin >> name;
        cout << "Ажилчны албан тушаал оруулна уу?" << endl;
        cin >> position;
        cout << "Ажилчны ажилласан цагийг оруулна уу?" << endl;
        cin >> workedHours;
    };

    void put_data() {
        cout << "Ажилчны дугаар: " << id << endl;
        cout << "Ажилчны нэр: " << name << endl;
        cout << "Ажилчны албан тушаал: " << position << endl;
        cout << "Ажилчны ажилласан цаг: " << workedHours << endl;
    };

    float calc_salary() {
        float salary = workedHours * 200;
        string str1 = "Zahiral";
        if (position == str1) {
            salary = calc_ceo_salary(salary);
        }
        return salary;
    };

    float calc_ceo_salary(float n) {
        float s = n + 5000;
        return s;
    };

    bool add_workedHours(float hour) {
        if (hour > 0 && hour <= 24) {
            workedHours += hour;
            return true;
        } else {
            return false;
        }
    };
};

int main() {
    Worker employee;
    employee.get_data();
    cout << endl << "Ажилтны мэдээлэл" << endl;
    employee.put_data();
    cout << endl << "Цалин: " << employee.calc_salary() << endl;
    employee.add_workedHours(6);
    cout << endl << "Ажилласан цаг нэмсний дараа:" << endl;
    employee.put_data();

    return 0;
}
```