

Класс Тодорхойлж Объект Үүсгэх Лабораторийн Ажил (Лаборатори №3)

1. ОРШИЛ/УДИРТГАЛ

Амьдралд байгаа объектыг компьютерийн програмд загварчлахад классыг тодорхойлж, объект үүсгэн шинж чанарыг болон ажилчдыг бүртгэх, цаг бүртгэх, цалин тооцох зэрэг үйлдлийг объектод хийнэ.

2. ЗОРИЛГО

Ажилчин нэртэй объектын классыг өгөгдсөн шаардлагын дагуу зарлаж, гишүүн функцуудийг тодорхойлно. Үүний тулд дараах зорилтуудыг тавьж ажилласан:

1. Классын гишүүн өгөгдлүүдийн төрлийг тодорхойлох,
2. Классын гишүүн функцүүдийг тодорхойлох,
3. Объект үүсгэж, боловсруулалт хийх.

3. ОНОЛЫН СУДАЛГАА

3.1 Класс ба объект

Бодит объектыг програмын орчинд хийсвэрлэхдээ Класс /Class/ гэсэн ойлголтыг ашигладаг. Класс гэдэг нь объектын ерөнхий загварчилгаа бөгөөд хэрэглэгчийн тодорхойлсон өгөдлийн төрөл гэж ойлгож болно. Класс нь гишүүн өгөгдөл болон гишүүн функцуудээс тогтоно.

Объект бол уг классын тодорхойлсны дагуу бий болсон нэгэн төрлийн хувьсагч юм. Объект нь өөрийн шинж чанар, зан төлөвтэй байдаг. Энд шинж чанар нь өгөгдөл, зан төлөв нь функц юм.

3.2 Гишүүн өгөгдөл

Тухайн классын агуулж буй хувьсагчууд бөгөөд тухайн хувьсагчийн утгад гишүүн функцуудээр дамжуулж хандах боломжтой.

C++ хэлд гурван ялгаатай хандалтын тодорхойлогч байдаг.

Public: классын гаднаас хандаж утгыг өөрчлөх боломжтой хувьсагчдыг заана.

Private: классын гаднаас хандах боломжгүй бөгөөд зөвхөн гишүүн функцуудээр дамжуулж хандаж болох хувьсагчдыг заана.

Protected: private-тай төстэй боловч удамшсан классаас хандах боломжтой хувьсагчдыг заана.

Классын гишүүн өгөгдөлд болон функцэд хандахдаа (.) операторыг ашиглана:

Объектийн_нэр . гишүүн_өгөгдөл

Объектийн_нэр . гишүүн_функц() байдлаар хандана.

3.3 Гишүүн функц

Гишүүн функц нь классын дотор тодорхойлогдсон функц юм. Гишүүн функцийг бусад функцээс ялгарах гол ялгаа өөрийн классын хувьсагчуудад ямар ч тохиолдолд хандах бүрэн боломжтой байдал юм.

Гишүүн функцийг классын гадна тодорхойлохдоо:

```
буцаах_утгын_төрөл классын_нэр :: функцийг нэр ( авах_параметрууд) {  
    .....  
}
```

байдлаар бичнэ.

4. ХЭРЭГЖҮҮЛЭЛТ

Даалгаварт өгсөн загварын дагуу классын гишүүн өгөгдөл, гишүүн функцийг тодорхойлсон. Бүх өгөгдөл болон функц гаднаас хандаж болохуйц public хандалттай байсан.

Уг ажилчин класс нь ажилтны дугаар, нэр, албан тушаал, ажилласан цаг гэх өгөгдлүүдийг агуулна. Мөн гарааны утга оноох, гараас утга авах, ажилчдын мэдээллийг дэлгэцлэх, цалин бодох, ажилласан цагийг нэмэгдүүлэх гэх мэт функцүүдтэй юм.

Main функц дотор уг классын объектон хүснэгт үүсгээд шинэ ажилчдын мэдээллийг нэмж оруулж болохуйц байдлаар байгуулсан.

Main функц нь давталтаар зохион байгуулагдсан бөгөөд гарах нөхцлийг хангах хүртэл програм ажиллаж ажилчин нэмэх, мэдээллийг оруулах, мэдээллийг харах байдлаар хийсэн.

Ажилтны цалинг ажилласан цаг дээр үндэслэн бодож ажилчдын мэдээллийг дэлгэцлэхэд харагдана. Захиралын цалинг цагийн 10000, бусад ажилчдын цалинг цагийн 5000 гэж үзсэн.

5. ДҮГНЭЛТ

Классыг загварчилснаар өгөгдлийг илүү эмх цэгцтэй мөн хамгаалалт сайтай байдлаар зохион байгуулах боломжийг олгож байна. Объект болон классыг ашигласнаар програмчлалыг амьдралд илүү ойрхон болгож, ойлгомжгүй байдлыг бууруулж байна гэж бодож байна.

6. АШИГЛАСАН МАТЕРИАЛ

1. https://www.w3schools.com/cpp/cpp_classes.asp

7. XABCPAJIT

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

class Ajilchin {
public:
    int dugaar;
    char ner[20];
    char alban_tushaal[20];
    float ajillasan_tsag;
public:
    void garaanii_utga();
    void garaas_utga_avah();
    void medeelel_delgetsleh();
    float tsalin_bodoh();
    float zahiraliin_tsalin();
    bool tsag_nemeh(float);
};

void Ajilchin::garaanii_utga() {
    dugaar = 0;
    strcpy(ner, "");
    strcpy(alban_tushaal, "ajichin");
    ajillasan_tsag = 0;
}

void Ajilchin::garaas_utga_avah() {
    cout << "ajiltnii dugaariig oruul: ";
    cin >> dugaar;
    cout << "ajiltnii neriig oruul: ";
    gets(ner);
    gets(ner);
    cout << "ajiltnii alban tushaaliig oruul: ";
    gets(alban_tushaal);
}

void Ajilchin::medeelel_delgetsleh() {
    cout << dugaar << " dugaartai ";
    cout << ner << " ajiltan ni ";
    cout << alban_tushaal << " alban tushaaltai buguud ";
    cout << tsalin_bodoh() << " tsalintai baina.\n";
}

float Ajilchin::tsalin_bodoh() {
    if(strcmp(alban_tushaal, "zahirai")==0);
```

```

        return zahiraliin_tsalin();
    float tsagiin_huls = 5000;
    return ajillasan_tsag*tsagiin_huls;
}

float Ajilchin::zahiraliin_tsalin() {
    float tsagiin_huls = 10000;
    return ajillasan_tsag*tsagiin_huls;
}

bool Ajilchin::tsag_nemeh(float k) {
    if(k>=0 && k<=24) {
        ajillasan_tsag += k;
        return true;
    }
    return false;
}

int main()
{
    int a=0, n=0;
    cout << "1. Ajiltan nemeh\n2. Ajiltnii medeelel oruulah\n3. Ajilchdiin medeelliig haruulah\n4.
    Ajillasan tsag nemeh\n5. Garah\n";
    cin >> a;
    Ajilchin k[100];
    while(a != 5) {
        if(a==1) {
            k[n].garaanii_utga();
            n++;
        }
        if(a==2) {
            k[n].garaas_utga_avah();
        }
        if(a==3) {
            for(int i=0; i<=n; i++) {
                k[i].medeelel_delgetsleh();
            }
        }
        if(a==4) {
            cout << "Ajillasan tsagiig nemegduuleh gej bui ajiltnii dugaariig oruul: ";
            int kod;
            float tsag;
            cin >> kod;
            for(int i=0; i<=n; i++) {
                if(k[i].dugaar == kod) {
                    cout << "nemeh tsagaa oruul: ";
                    cin >> tsag;
                    if(k[i].tsag_nemeh(tsag) == true) {
                        cout << "Ajiltnii tsag amjilttai nemegdlee.\n";
                    }
                }
            }
        }
    }
}

```

```

        } else {
            cout << "Oruulsan tsag buruu baina.\n";
        }
    }
}
cout << "1. Ajiltan nemeh\n2. Ajiltnii medeel oruulah\n3. Ajilchdiin medeelliig haruulah\n4.
Ajillasan tsag nemeh\n5. Garah\n";
cin >> a;
}
return 0;
}

```