

<객체지향프로그래밍_10주차과제_소스구현설명>

7조(202104140김경록,202104276이선우)

```
#include <iostream>

using namespace std;

class printer {
    string model, manufacturer;
    int printedCount, availableCount;
protected:
    printer(string model, string manufacturer, int availableCount) {
        this->model = model;
        this->manufacturer = manufacturer;
        this->availableCount = availableCount;
    }
    bool isprint(int pages) {
        if (availableCount >= pages)
            return true;
        else
            return false;
    }
    void print(int pages) {
        if (isprint(pages)) {
            printedCount += pages;
            availableCount -= pages;
            cout << "프린트 하였습니다." << endl;
        }
    }
};
```

```

        else {

            cout << "잉크가 부족하여 프린트할 수 없습니다." << endl;

        }

    }

    void remain(int pages) {

        cout << model << " , " << manufacturer << " , 남은 종이 " << availableCount <<
"장";

    }

};

```

```

class Inkjetprinter : public printer {

    int availableink;

public:

    Inkjetprinter(string model, string manufacturer, int availableCount, int
availableink) :printer(model, manufacturer, availableCount) {

        this->availableink = availableink;

    }

    bool isprinterInkjet(int pages) {

        if (availableink > pages)

            return true;

        else

            return false;

    }

    void printInkJet(int pages) {

        if (isprint(pages)) {

            if (isprinterInkjet(pages)) {

```

```

        print(pages);

        availableink -= pages;

    }

}

else cout << "토너가 부족하여 프린트를 할 수 없습니다." << endl;

}

void showinkprinter(int pages) {

    remain(pages);

    cout << " ,남은 잉크 " << availableink << endl;

}

};

class laserprinter : public printer {

    int availabletoner;

public:

    laserprinter(string model, string manufacturer, int availableCount, int
availabletoner) :printer(model, manufacturer, availableCount) {

        this->availabletoner = availabletoner;

    }

    bool isprinterlaser(int pages) {

        if (availabletoner > pages)

            return true;

        else

            return false;

    }

    void printlaser(int pages) {

```

```

        if (isprint(pages)) {
            if (isprinterlaser(pages)) {
                print(pages);
                availabletoner -= pages;
            }
        }
        else cout << "용지가 부족하여 프린트를 할 수 없습니다." << endl;
    }

    void showlaserprinter(int pages) {
        remain(pages);
        cout << " ,남은 토너 " << availabletoner << endl;
    }
};

int main() {
    string answer = "y";
    int type, pages;

    cout << "현재 작동중인 2 대의 프린터는 아래와 같다" << endl;

    Inkjetprinter inkp("Officejet V40", "HP", 5, 10);
    cout << "잉크젯 : ";
    inkp.showinkprinter(0);

    laserprinter laserp("SCX-6x45", "삼성전자", 3, 20);
    cout << "레이저 : ";
    laserp.showlaserprinter(0);

```

```

do {

    cout << endl << "프린터(1:잉크젯, 2:레이저)와 매수 입력>>";

    cin >> type >> pages;

    switch (type) {

    case 1:

        inkp.printInkJet(pages);

        break;

    case 2:

        laserp.printlaser(pages);

        break;

    default:

        break;

    }

    inkp.showinkprinter(pages);

    laserp.showlaserprinter(pages);

    cout << "계속 프린트 하시겠습니까(y/n)>>";

    cin >> answer;

    } while (answer == "y");

}

```

----문제 정의----

서로 다른 두 종류의 프린터에서 지정한 매수만큼 문서를 출력하는 프로그램을 작성하는 문제이다. 사용자 입력에 따라 잉크젯 혹은 레이저 프린터 중 하나를 선택한 후 프린트가 가능한지에 대한 여부를 확인하고, 프린트를 완료하면 잔여 용지 수와 잉크 또는 토너를 출력하는 프로그램이다.

----문제 해결 방법, 아이디어들----

Printer 기본 클래스를 만들고, 이 클래스를 상속받아 잉크젯, 레이저 프린터 클래스를 각각 구현한다. 두 프린터는 공통적인 기능들을 가지고 있지만, 사용되는 소모품이 다르기 때문에 각각 잉크, 토너를 검사하는 기능을 구현하였다. 사용자가 입력한 매수와 프린터 종류에 따라 조건을 확인하고, 프린터가 가능한 경우에만 매수를 차감한다.

----아이디어 평가----

객체 지향적인 설계를 통한 상속을 통해 코드의 재사용성을 높일 수 있다.

프린터마다 다른 부분의 파생 클래스를 각각 구현하여 코드의 유연성과 가독성을 높인다.

----문제를 해결한 키 아이디어 또는 알고리즘 설명----

1. 상속 구조 활용: Printer클래스를 통한 기본 데이터를 관리하고, 잉크젯, 레이저 프린터기에 상속하여 다른 요소를 따로 추가하였다.
2. 프린트 작업을 수행하기 전에 isPrint() 함수를 통해 용지가 충분한 지, isPrinterInkjet() 혹은 isPrinterLaser()함수로 잉크와 토너가 충분한 지에 대해 확인한다.
3. Main 함수에서는 프린터 종류와 매수를 입력 받아 적절한 프린터를 선택하고, 출력 작업을 한다.