

# C# 1번째 개인 과제

학과, 학번 : 컴퓨터공학과 21011575

이름 : 박준형

## 1. 명령어 입력과 결과 화면

```
//StreamReader로 input.txt에서 입력받는다.  
//필자는 해당 컴퓨터의 경로를 입력받아 경로를 아래와 같이 작성하였다.  
StreamReader sr = new StreamReader("C:\\Users\\박준형\\source\\repos\\FirstProgram\\FirstProgram\\input.txt");
```

StreamReader를 사용하여 컴퓨터에 있는 input.txt에 있는 문장들을 한줄씩 입력받아 명령어를 처리하는 과정으로 코딩을 하였고, 해당 명령어들을 통한 결과가 콘솔창을 통하여 보이도록 하였다.

```
선택 Microsoft Visual Studio 디버그 콘솔  
Total Cost: 0  
Computer List:  
(1) type: Netbook, ComId: 1, NetId: 1, Used for: internet, Avail: Y  
(2) type: Netbook, ComId: 2, NetId: 2, Used for: internet, Avail: Y  
(3) type: Netbook, ComId: 3, NetId: 3, Used for: internet, Avail: Y  
(4) type: Notebook, ComId: 4, NetId: 1, Used for: internet, scientific, Avail: Y  
(5) type: Notebook, ComId: 5, NetId: 2, Used for: internet, scientific, Avail: Y  
(6) type: Desktop, ComId: 6, DeskId: 1, Used for: internet, scientific, game, Avail: Y  
User List:  
(1) type: Students, Name: Kim, UserId: 1, Stuld: 1, Used for: internet, scientific, Rent: N  
(2) type: Gamers, Name: Lee, UserId: 2, GamerId: 1, Used for: internet, game, Rent: N  
(3) type: OfficeWorkers, Name: Choi, UserId: 3, WorkerId: 1, Used for: internet, Rent: N  
(4) type: Students, Name: Park, UserId: 4, Stuld: 2, Used for: internet, scientific, Rent: N  
=====  
Computer #6 has been assigned to User #2  
=====  
Computer #1 has been assigned to User #3  
=====  
Total Cost: 0  
Computer List:  
(1) type: Netbook, ComId: 1, NetId: 1, Used for: internet, Avail: N (UserId:3,DR:2,DL:2,DU:0)  
(2) type: Netbook, ComId: 2, NetId: 2, Used for: internet, Avail: Y  
(3) type: Netbook, ComId: 3, NetId: 3, Used for: internet, Avail: Y  
(4) type: Notebook, ComId: 4, NetId: 1, Used for: internet, scientific, Avail: Y  
(5) type: Notebook, ComId: 5, NetId: 2, Used for: internet, scientific, Avail: Y  
(6) type: Desktop, ComId: 6, DeskId: 1, Used for: internet, scientific, game, Avail: N (UserId:2,DR:3,DL:3,DU:0)  
User List:  
(1) type: Students, Name: Kim, UserId: 1, Stuld: 1, Used for: internet, scientific, Rent: N  
(2) type: Gamers, Name: Lee, UserId: 2, GamerId: 1, Used for: internet, game, Rent: Y (RentCompld: 6)  
(3) type: OfficeWorkers, Name: Choi, UserId: 3, WorkerId: 1, Used for: internet, Rent: Y (RentCompld: 1)  
(4) type: Students, Name: Park, UserId: 4, Stuld: 2, Used for: internet, scientific, Rent: N  
=====  
It has passed one day..  
=====  
It has passed one day..  
Time for Computer #1 has expired. User #3 has returned Computer #1 and paid 14000won.  
=====  
Total Cost: 14000  
Computer List:  
(1) type: Netbook, ComId: 1, NetId: 1, Used for: internet, Avail: Y  
(2) type: Netbook, ComId: 2, NetId: 2, Used for: internet, Avail: Y  
(3) type: Netbook, ComId: 3, NetId: 3, Used for: internet, Avail: Y  
(4) type: Notebook, ComId: 4, NetId: 1, Used for: internet, scientific, Avail: Y  
(5) type: Notebook, ComId: 5, NetId: 2, Used for: internet, scientific, Avail: Y  
(6) type: Desktop, ComId: 6, DeskId: 1, Used for: internet, scientific, game, Avail: N (UserId:2,DR:3,DL:1,DU:2)  
User List:  
(1) type: Students, Name: Kim, UserId: 1, Stuld: 1, Used for: internet, scientific, Rent: N  
(2) type: Gamers, Name: Lee, UserId: 2, GamerId: 1, Used for: internet, game, Rent: Y (RentCompld: 6)  
(3) type: OfficeWorkers, Name: Choi, UserId: 0, WorkerId: 1, Used for: internet, Rent: N  
(4) type: Students, Name: Park, UserId: 4, Stuld: 2, Used for: internet, scientific, Rent: N  
=====  
User #2 has returned Computer #6 and paid 26000 won.  
=====
```

```

Total Cost: 40000
Computer List:
(1) type: Netbook, ComId: 1, NetId: 1, Used for: internet, Avail: Y
(2) type: Netbook, ComId: 2, NetId: 2, Used for: internet, Avail: Y
(3) type: Netbook, ComId: 3, NetId: 3, Used for: internet, Avail: Y
(4) type: Notebook, ComId: 4, NoteId: 1, Used for: internet, scientific, Avail: Y
(5) type: Notebook, ComId: 5, NoteId: 2, Used for: internet, scientific, Avail: Y
(6) type: Desktop, ComId: 6, DeskId: 1, Used for: internet, scientific, game, Avail: Y
User List:
(1) type: Students, Name: Kim, UserId: 1, Stuld: 1, Used for: internet, scientific, Rent: N
(2) type: Gamers, Name: Lee, UserId: 2, GamerId: 1, Used for: internet, game, Rent: N
(3) type: OfficeWorkers, Name: Choi, UserId: 0, WorkerId: 1, Used for: internet, Rent: N
(4) type: Students, Name: Park, UserId: 4, Stuld: 2, Used for: internet, scientific, Rent: N
=====
C:\Users\박준형\source\repos\FirstProgram\FirstProgram\bin\Debug\net7.0\FirstProgram.exe(프로세스 31248개)이(가) 종료되
었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요...

```

## 2. 상속성 사용하여 클래스 선언

```

//컴퓨터 정보 클래스
참조 9개
public class Computer
{
    public int ComId;
    public string Avail = "Y";
    public int UserId = 0;
    public int DR = 0;
    public int DL = 0;
    public int DU = 0;
}

//노트북 정보 클래스
참조 17개
public class Notebook : Computer
{
    public string type = "Notebook";
    public int NoteId;
    public string avail_internet = "Y";
    public string avail_scientific = "Y";
    public string avail_game = "N";
    public int day_cost = 10000;
}

//넷북 정보 클래스
참조 16개
public class Netbook : Computer
{
    public string type = "Netbook";
    public int NetId;
    public string avail_internet = "Y";
    public string avail_scientific = "N";
    public string avail_game = "N";
    public int day_cost = 7000;
}

//데스크탑 정보 클래스
참조 18개
public class Desktop : Computer
{
    public string type = "Desktop";
    public int DeskId;
    public string avail_internet = "Y";
    public string avail_scientific = "Y";
    public string avail_game = "Y";
    public int day_cost = 13000;
}

```

- 가장 위에 컴퓨터 클래스를 선언 후 클래스 아래에 해당 컴퓨터 정보를 필드로 입력하고, 이후, 노트북, 넷북, 데스크탑을 컴퓨터를 상속한 자식 클래스로 두고 해당 기기의 정보들을 필드로 설정해 두었다.

```

//User 정보 클래스
참조 9개
public class User
{
    public string name;
    public int userid;
    public int computerid = 0;
    public string rent = "N";
    public int lent_day = 0;
    public int use_day = 0;
}

//Students 정보 클래스
참조 9개
public class Students : User
{
    public string type = "Students";
    public int studid;
    public string avail_internet = "Y";
    public string avail_scientific = "Y";
    public string avail_game = "N";
}

//Workers 정보 클래스
참조 9개
public class Workers : User
{
    public string type = "OfficeWorkers";
    public int workerid;
    public string avail_internet = "Y";
    public string avail_scientific = "N";
    public string avail_game = "N";
}

//Gamers 정보 클래스
참조 9개
public class Gamers : User
{
    public string type = "Gamers";
    public int gamerid;
    public string avail_internet = "Y";
    public string avail_scientific = "N";
    public string avail_game = "Y";
}

```

- User 클래스를 선언한후 유저 정보를 담은 필드를 작성하였고, 그 이후, Students, Workers, Gamers 클래스는 User 클래스를 상속한 자식 클래스로 설정하였다.

3. ComputerManager 클래스 구현 (코드가 길어 코드 사진은 제외하였고, 소스코드파일에서 볼 수 있습니다.)

- ComputerManager 클래스에서는 해당 컴퓨터 정보와 사용자 정보를 출력하여 주는 Print 메소드, 해당 사용자에게 이용할 수 있는 컴퓨터를 할당하여 주는 Allocation 메소드, 하루 날짜가 지난 것을 확인하여 주는 pass\_day메소드, 해당 사용자가 가진 컴퓨터를 반납하여 주는 returning 메소드가 포함되어 있어, 해당 명령어를 입력받았을 때 ComputerManager 클래스의 해당 메소드를 이용할 수 있도록 만들어 두었다.

#### 4. 이외 구현사항(메인 함수)

```
int line = 0;
int[] type_computer = new int[3];
int num_computer = 0;
int com_num = 1;
int net_num = 1;
int note_num = 1;
int desk_num = 1;

//is,as 사용을 위해 빈 변수 만들어 내기
Netbook a;
Notebook b;
Desktop c;

Computer[] arrComp = null;
int user_num = 1;
int stu_num = 1;
int worker_num = 1;
int gamer_num = 1;

//is,as 사용을 위해 빈 변수 만들어 내기
Students d;
Workers e;
Gamers f;

int num_user = 0;
User[] arrUser = null;
int cnt = 0;
int total = 0;
int gash = 0;
```

- 메인 함수에서는 컴퓨터와 사용자의 정보를 입력해 둘 수 있도록 위와 같이 변수들을 미리 만들어 두었다.

```
// 입력 파일에 더 이상 읽을 문자가 없을 때 까지 실행
while (sr.Peek() >= 0)
{
    line++;
}
```

- 이후 sr.Peek()를 통하여 input.txt의 모든 문장들을 입력값으로 입력받는 식으로 하였고,

1번째 줄에서는 총 컴퓨터의 개수를 입력받고,

2번째 줄은 컴퓨터의 종류 중 노트북, 데스크탑, 넷북 수를 각자 입력받는다. 또한, arrComp라는 컴퓨터 배열에 각자의 컴퓨터 종류를 할당하는 과정을 거치고 각각의 컴퓨터에 Comid와 같은 컴퓨터 정보를 입력해주었다.

3번째 줄은 user의 수를 입력받으며,

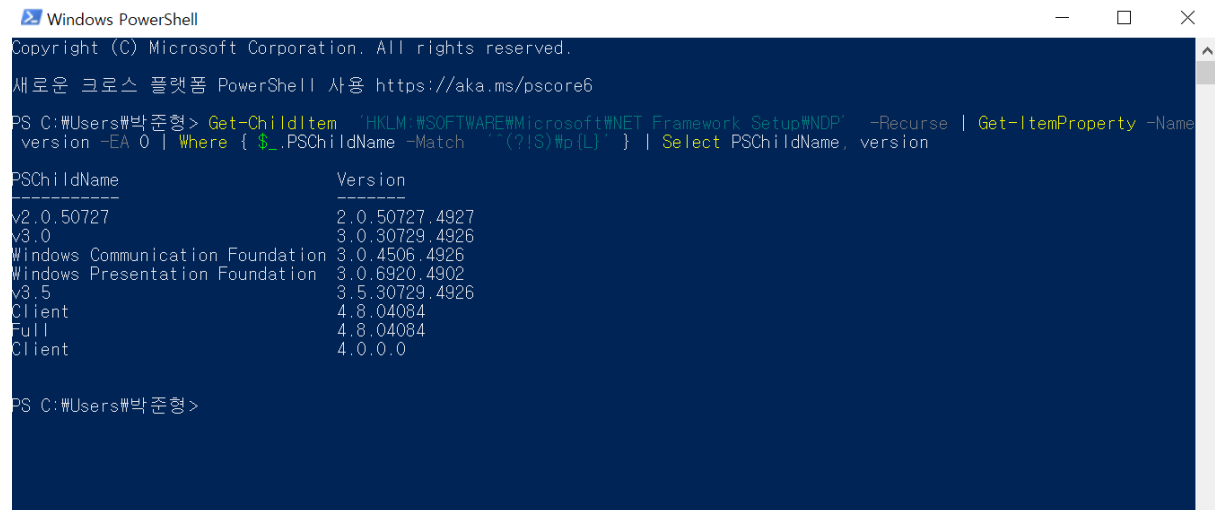
4번째 줄부터 시작하여 user의 개수만큼은 user의 직업과 이름을 입력받아 사용자 배열인 arrUser에 사용자의 정보를 입력해주는 과정을 거쳤다.

이후 줄부터는 Q가 입력될때까지 명령어를 입력받아 S를 입력받을 경우 컴퓨터, 사용자 정보 출력, A를 입력받았을 경우 해당 사용자에게 컴퓨터 할당, T가 입력받았을 경우 pass\_day메소드를 통해 하루가 지난 것을 확인해주는 과정, R을 입력받았을 경우 해당 사용자의 컴퓨터 반납과정을 거치

도록 구현을 하였다. (코드가 너무 길기 때문에 해당 코드는 소스코드를 통해서 확인할 수 있습니다.)

## 5. 개발한 환경

- Windows Powershell에서 명령어를 입력하여 해당 개발환경을 확인할 수 있었다.



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

새로운 크로스 플랫폼 PowerShell 사용 https://aka.ms/pscore6

PS C:\Users\#박준형> Get-Childitem 'HKLM:\SOFTWARE\Microsoft\NET Framework Setup\NDP' -Recurse | Get-ItemProperty -Name version -EA 0 | Where { $_.PSChildName -Match '(?!(S)#p[L])' } | Select PSChildName, version
```

PSChildName	Version
v2.0.50727	2.0.50727.4927
v3.0	3.0.30729.4926
Windows Communication Foundation	3.0.4506.4926
Windows Presentation Foundation	3.0.6920.4902
v3.5	3.5.30729.4926
Client	4.8.04084
Full	4.8.04084
Client	4.0.0.0

```
PS C:\Users\#박준형>
```