

[화면]

Form1

차량번호

입차

출차

전체현황

```
13 public partial class Form1 : Form
14 {
15     private Car[] _cars = new Car[10];
16     private int _currentCarCount = 0;
17
18     public Form1()
19     {
20         InitializeComponent();
21     }
22
23     private void btnIn_Click(object sender, EventArgs e)
24     {
25         string number = tbxNumber.Text.Trim();
26         if (true == string.IsNullOrEmpty(number))
27         {
28             MessageBox.Show("차량번호를 반드시 넣어주세요.");
29             return;
30         }
31
32         for(int i=0; i < this._currentCarCount ; i++)
33         {
34             if(_cars[i].CarNumber == number)
35             {
36                 MessageBox.Show("현재 입고되어 있는 차량 번호입니다.");
37                 return;
38             }
39         }
40
41         this._cars[this._currentCarCount] = new Car();
42         this._cars[this._currentCarCount].CarNumber = number;
43
44         var now_datetime = (DateTime.Now).AddMinutes(-47);
45
46         this._cars[this._currentCarCount].InTime = now_datetime;
47         _currentCarCount++;
48
49         MessageBox.Show($"{number} 차량 입고완료");
50     }
51 }
```

```

private void btnOut_Click(object sender, EventArgs e)
{
    string number = tbxNumber.Text.Trim();
    if (true == string.IsNullOrEmpty(number))
    {
        MessageBox.Show("차량번호를 반드시 넣어주세요.");
        return; //메소드 탈출문
    }

    for (int i = 0; i < this._currentCarCount; i++)
    {
        if (_cars[i].CarNumber == number)
        {
            _cars[i].OutTime = DateTime.Now;
            int charge = _cars[i].CalcParkingCharge();
            int min = _cars[i].ParkingTime();

            if(charge == 0)
            {
                MessageBox.Show($"회차 차량입니다.{min}분");
            }
            else
            {
                MessageBox.Show(string.Format("{0}분 주차 / {1}원", min, charge));
            }

            for(int j=i; j < this._currentCarCount - 1; j++)
            {
                this._cars[j] = this._cars[j + 1];
            }

            _cars[--this._currentCarCount] = null;

            return;
        }
    }

    MessageBox.Show("현재 입고되어 있는 차량이 없습니다.");
}

private void btnView_Click(object sender, EventArgs e)
{
    string message = string.Empty;
    for(int i=0; i < _currentCarCount; i++)
    {
        message += string.Format("[{0}] 입고시간:{1} 주차시간:{2}{3}"
            , _cars[i].CarNumber
            , _cars[i].InTime
            , _cars[i].ParkingTime()
            , Environment.NewLine
        );
    }
    tbxView.Text = message;
}

```

```
9      class Car
10     {
11         public static int HalfHour_Charge = 1000;
12         public static int TurningTime = 10;
13
14         public string CarNumber;
15         public DateTime InTime;
16         public DateTime OutTime;
17
18         public int ParkingTime()
19         {
20             if(this.OutTime == DateTime.MinValue)
21             {
22                 return (int)(DateTime.Now - this.InTime).TotalMinutes;
23             }
24             else
25             {
26                 return (int)(this.OutTime - this.InTime).TotalMinutes;
27             }
28         }
29
30         public string PrintState()
31         {
32             int parkingTime = ParkingTime();
33             string message;
34             if (parkingTime > 0)
35             {
36                 message = string.Format("[{0}] {1}분 주차", this.CarNumber, parkingTime);
37             }
38             else
39             {
40                 message = string.Format("[{0}] 주차중", this.CarNumber);
41             }
42             return message;
43         }
44
45         public int CalcParkingCharge()
46         {
47             int parkingTime = ParkingTime();
48             if( parkingTime <= Car.TurningTime)
49             {
50                 return 0;
51             }
52             else
53             {
54                 var halfHour = parkingTime / 30;
55                 var remainderHour = parkingTime % 30;
56                 var totalcharge = (halfHour * Car.HalfHour_Charge)
57                     + (remainderHour > 0 ? Car.HalfHour_Charge : 0);
58
59                 return totalcharge;
60             }
61         }
62     }
```

```
13 public partial class Form1 : Form
14 {
15     List<Car> _cars = new List<Car>();
16
17     public Form1()
18     {
19         InitializeComponent();
20     }
21
22     private void btnIn_Click(object sender, EventArgs e)
23     {
24         string number = tbxNumber.Text.Trim();
25         if (true == string.IsNullOrEmpty(number))
26         {
27             MessageBox.Show("차량번호를 반드시 넣어주세요.");
28             return;
29         }
30
31         foreach(Car car in _cars)
32         {
33             if(car.CarNumber == number)
34             {
35                 MessageBox.Show("현재 입고되어 있는 차량 번호입니다.");
36                 return;
37             }
38         }
39
40         Car newCar = new Car();
41         newCar.CarNumber = number;
42         newCar.InTime = DateTime.Now.AddMinutes(-47);
43
44         _cars.Add(newCar);
45
46         MessageBox.Show($"{number} 차량 입고완료");
47     }
48 }
```

```

private void btnOut_Click(object sender, EventArgs e)
{
    string number = tbxNumber.Text.Trim();
    if (true == string.IsNullOrEmpty(number))
    {
        MessageBox.Show("차량번호를 반드시 넣어주세요.");
        return;
    }

    foreach(var car in _cars){
        if(car.CarNumber == number)
        {
            car.OutTime = DateTime.Now;
            int charge = car.CalcParkingCharge();
            int min = car.ParkingTime();

            if (charge == 0)
            {
                MessageBox.Show($"회차 차량입니다.{min}분");
            }
            else
            {
                MessageBox.Show(string.Format("{0}분 주차 / {1}원", min, charge));
            }

            this._cars.Remove(car);
        }
    }
    MessageBox.Show("현재 입고되어 있는 차량이 없습니다.");
}

private void btnView_Click(object sender, EventArgs e)
{
    string message = string.Empty;
    for (int i = 0; i < _cars.Count ; i++)
    {
        message += string.Format("[{0}] 입고시간:{1} 주차시간:{2}{3}"
            , _cars[i].CarNumber
            , _cars[i].InTime
            , _cars[i].ParkingTime()
            , Environment.NewLine
        );
    }
    tbxView.Text = message;
}
}

```

← return; 추가

```
9      class Car
10     {
11         public static int HalfHour_Charge = 1000;
12         public static int TurningTime = 10;
13
14         public string CarNumber;
15         public DateTime InTime;
16         public DateTime OutTime;
17
18         public int ParkingTime()
19         {
20             if(this.OutTime == DateTime.MinValue)
21             {
22                 return (int)(DateTime.Now - this.InTime).TotalMinutes;
23             }
24             else
25             {
26                 return (int)(this.OutTime - this.InTime).TotalMinutes;
27             }
28         }
29
30         public string PrintState()
31         {
32             int parkingTime = ParkingTime();
33             string message;
34             if (parkingTime > 0)
35             {
36                 message = string.Format("[{0}] {1}분 주차", this.CarNumber, parkingTime);
37             }
38             else
39             {
40                 message = string.Format("[{0}] 주차중", this.CarNumber);
41             }
42             return message;
43         }
44
45         public int CalcParkingCharge()
46         {
47             int parkingTime = ParkingTime();
48             if( parkingTime <= Car.TurningTime)
49             {
50                 return 0;
51             }
52             else
53             {
54                 var halfHour = parkingTime / 30;
55                 var remainderHour = parkingTime % 30;
56                 var totalcharge = (halfHour * Car.HalfHour_Charge)
57                     + (remainderHour > 0 ? Car.HalfHour_Charge : 0);
58
59                 return totalcharge;
60             }
61         }
62     }
```

[Week6_3]

Form1.cs

```
14 public partial class Form1 : Form
15 {
16     List<Car> _cars = new List<Car>();
17
18     public Form1()
19     {
20         InitializeComponent();
21     }
22
23     private void btnIn_Click(object sender, EventArgs e)
24     {
25         string number = tbxNumber.Text.Trim();
26
27         if (!CheckCarNumber(number))
28         {
29             MessageBox.Show("차량번호가 형식에 어긋납니다.");
30             return;
31         }
32
33         if(SearchCar(number) != null)
34         {
35             MessageBox.Show("현재 입고되어 있는 차량 번호입니다.");
36             return;
37         }
38
39         Car newCar = new Car();
40         newCar.CarNumber = number;
41         newCar.InTime = DateTime.Now.AddMinutes(MakeTestTime());
42
43         _cars.Add(newCar);
44
45         MessageBox.Show($"{number} 차량 입고완료");
46     }
47 }
```



```

private void btnOut_Click(object sender, EventArgs e)
{
    string number = tbxNumber.Text.Trim();

    if (!CheckCarNumber(number))
    {
        MessageBox.Show("차량번호가 형식에 어긋납니다.");
        return;
    }

    var car = SearchCar(number);

    if (car != null)
    {
        car.OutTime = DateTime.Now;
        int charge = car.CalcParkingCharge();
        int min = car.ParkingTime();

        if (charge == 0)
        {
            MessageBox.Show($"회차 차량입니다.{min}분");
        }
        else
        {
            MessageBox.Show(string.Format("{0}분 주차 / {1}원", min, charge));
        }

        this._cars.Remove(car);
    }
    else
    {
        MessageBox.Show("현재 입고되어 있는 차량이 없습니다.");
    }
}

private void btnView_Click(object sender, EventArgs e)
{
    string message = string.Empty;

    for (int i = 0; i < _cars.Count; i++)
    {
        message += string.Format("[{0}] 입고시간:{1} 주차시간:{2}{3}"
            , _cars[i].CarNumber
            , _cars[i].InTime
            , _cars[i].ParkingTime()
            , Environment.NewLine
        );
    }

    tbxView.Text = message;
}

```

```
101 private bool CheckCarNumber(string number)
102 {
103     Regex regx = new Regex(@"^[0-9]{2,3}[가-힣]{1}[0-9]{4}$");
104     Match m = regx.Match(number);
105     return m.Success;
106 }
107
108 private int MakeTestTime()
109 {
110     Random rand = new Random();
111     return rand.Next(-70, 0);
112 }
113
114 private Car SearchCar(string number)
115 {
116     foreach(var car in _cars)
117     {
118         if(car.CarNumber == number)
119         {
120             return car;
121         }
122     }
123     return null;
124 }
125 }
```

```
9      class Car
10     {
11         public static int HalfHour_Charge = 1000;
12         public static int TurningTime = 10;
13
14         public string CarNumber;
15         public DateTime InTime;
16         public DateTime OutTime;
17
18         public int ParkingTime()
19         {
20             if(this.OutTime == DateTime.MinValue)
21             {
22                 return (int)(DateTime.Now - this.InTime).TotalMinutes;
23             }
24             else
25             {
26                 return (int)(this.OutTime - this.InTime).TotalMinutes;
27             }
28         }
29
30         public string PrintState()
31         {
32             int parkingTime = ParkingTime();
33             string message;
34             if (parkingTime > 0)
35             {
36                 message = string.Format("[{0}] {1}분 주차", this.CarNumber, parkingTime);
37             }
38             else
39             {
40                 message = string.Format("[{0}] 주차중", this.CarNumber);
41             }
42             return message;
43         }
44
45         public int CalcParkingCharge()
46         {
47             int parkingTime = ParkingTime();
48             if( parkingTime <= Car.TurningTime)
49             {
50                 return 0;
51             }
52             else
53             {
54                 var halfHour = parkingTime / 30;
55                 var remainderHour = parkingTime % 30;
56                 var totalcharge = (halfHour * Car.HalfHour_Charge)
57                     + (remainderHour > 0 ? Car.HalfHour_Charge : 0);
58
59                 return totalcharge;
60             }
61         }
62     }
```

[Week6_4]

Form1.cs – 강의 내용과 메소드 순서가 좀 다름 (순서 상관 없음)

```
14 public partial class Form1 : Form
15 {
16     List<Car> _cars = new List<Car>();
17
18     public Form1()
19     {
20         InitializeComponent();
21     }
22
23     private void btnIn_Click(object sender, EventArgs e)
24     {
25         string number = tbxNumber.Text.Trim();
26
27         if (!CheckCarNumber(number))
28         {
29             MessageBox.Show("차량번호가 형식에 어긋납니다.");
30             return;
31         }
32
33         if (SearchCar(number) != null)
34         {
35             MessageBox.Show("현재 입고되어 있는 차량 번호입니다.");
36             return;
37         }
38
39         Car newCar = new Car(number);
40
41         _cars.Add(newCar);
42
43         MessageBox.Show($"{number} 차량 입고완료");
44         ViewStatus();
45     }
46
47
48     private bool CheckCarNumber(string number)
49     {
50         Regex regx = new Regex(@"^[0-9]{2,3}[가-힣]{1}[0-9]{4}$");
51         Match m = regx.Match(number);
52         return m.Success;
53     }
54
55     private Car SearchCar(string number)
56     {
57         foreach (var car in _cars)
58         {
59             if (car.CarNumber == number)
60             {
61                 return car;
62             }
63         }
64         return null;
65     }
66 }
```

```

67 private void btnOut_Click(object sender, EventArgs e)
68 {
69     string number = tbxNumber.Text.Trim();
70
71     if (!CheckCarNumber(number))
72     {
73         MessageBox.Show("차량번호가 형식에 어긋납니다.");
74         return;
75     }
76
77     var car = SearchCar(number);
78     if (car != null)
79     {
80         car.OutTime = DateTime.Now;
81         //car.GoOut();
82         //car._outTime = DateTime.Now;
83         int charge = car.CalcParkingCharge();
84         int min = car.ParkingTime();
85
86         if (charge == 0)
87         {
88             MessageBox.Show($"회차 차량입니다.{min}분");
89         }
90         else
91         {
92             MessageBox.Show(string.Format("{0}분 주차 / {1}원", min, charge));
93         }
94         this._cars.Remove(car);
95     }
96     else
97     {
98         MessageBox.Show("현재 입고되어 있는 차량이 없습니다.");
99     }
100     ViewStatus();
101 }
102
103 private void btnView_Click(object sender, EventArgs e)
104 {
105     ViewStatus();
106 }
107
108 private void ViewStatus()
109 {
110     string message = string.Empty;
111     for (int i = 0; i < _cars.Count; i++)
112     {
113         message += string.Format("[{0}] 입고시간:{1} 주차시간:{2}{3}"
114             , _cars[i].CarNumber
115             , _cars[i].InTime
116             , _cars[i].ParkingTime()
117             , Environment.NewLine
118             );
119     }
120     tbxView.Text = message;
121 }
122 }

```

```
9  class Car
10 {
11     public static int HalfHour_Charge = 1000;
12     public static int TurningTime = 10;
13
14     private string _carNumber;
15     public string CarNumber //읽기전용 PROPERTY
16     {
17         get
18         {
19             return this._carNumber;
20         }
21     }
22
23     private DateTime _inTime;
24     public DateTime InTime
25     {
26         get
27         {
28             return this._inTime;
29         }
30     }
31
32     private DateTime _outTime;
33     public DateTime OutTime
34     {
35         set
36         {
37             this._outTime = value;
38         }
39     }
40
41     public Car(string number)
42     {
43         this._carNumber = number;
44         this._inTime = DateTime.Now.AddMinutes(Utility.MakeTestTime());
45     }
46
47     public int ParkingTime()
48     {
49         if (this._outTime == DateTime.MinValue)
50         {
51             return (int)(DateTime.Now - this._inTime).TotalMinutes;
52         }
53         else
54         {
55             return (int)(this._outTime - this._inTime).TotalMinutes;
56         }
57     }
58 }
59
```

```

60     public string PrintState()
61     {
62         int parkingTime = ParkingTime();
63         string message;
64         if (parkingTime > 0)
65         {
66             message = string.Format("[{0}] {1}분 주차", this._carNumber, parkingTime);
67         }
68         else
69         {
70             message = string.Format("[{0}] 주차중", this._carNumber);
71         }
72         return message;
73     }
74
75     public int CalcParkingCharge()
76     {
77         int parkingTime = ParkingTime(); //분 단위
78         if (parkingTime <= Car.TurningTime)
79         {
80             return 0;
81         }
82         else
83         {
84             var halfHour = parkingTime / 30;
85             var remainderHour = parkingTime % 30;
86             var totalcharge = (halfHour * Car.HalfHour_Charge)
87                 + (remainderHour > 0 ? Car.HalfHour_Charge : 0);
88             return totalcharge;
89         }
90     }
91
92     public void GoOut()
93     {
94         this._outTime = DateTime.Now;
95     }
96 }

```

Utility.cs

```

9     class Utility
10    {
11        public static int MakeTestTime()
12        {
13            Random rand = new Random();
14            return rand.Next(-70, 0);
15        }
16
17        public static int MakeTestTime(int min)
18        {
19            Random rand = new Random();
20            return rand.Next(min, 0);
21        }
22
23        public static int MakeTestTime(int min, int max)
24        {
25            Random rand = new Random();
26            return rand.Next(min, max);
27        }
28    }

```