

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Windows.Forms;
5
6 namespace Week04Homework
7 {
8     public partial class FormManager : Form
9     {
10         Department[] departments;
11         List<Professor> professors;
12         Dictionary<string, Student> students;
13
14         List<Grade> testGrades;
15         TextBox[] tbxTestScores;
16
17         public FormManager()
18         {
19             InitializeComponent();
20
21             departments = new Department[100];
22             professors = new List<Professor>();
23             students = new Dictionary<string, Student>();
24
25             for (int i = 1; i <= 4; i++) {
26                 cmbYear.Items.Add(i);
27             }
28
29             cmbClass.Items
30                 .AddRange(new object[] { "A", "B", "C" });
31
32             cmbRegStatus.Items.Add("재 학");
33             cmbRegStatus.Items.Add("졸 업");
34             cmbRegStatus.Items.Add("휴 학");
35             cmbRegStatus.Items.Add("퇴 학");
36
37             testGrades = new List<Grade>();
38             tbxTestScores = new TextBox[] {
39                 tbxTestScore1,
40                 tbxTestScore2,
41                 tbxTestScore3,
42                 tbxTestScore4,
43                 tbxTestScore5,
44                 tbxTestScore6,
45                 tbxTestScore7,
46                 tbxTestScore8,
47                 tbxTestScore9,
48             };
49
50             departments[0] = new Department() {
51                 Code = "A001",
52                 Name = "컴퓨터정보"
53             };
54
55             departments[1] = new Department() {
56                 Code = "A002",
```

```
57         Name = "컴퓨터시스템"
58     };
59
60     for (int i = 0; i < departments.Length; i++) {
61         if (departments[i] != null) {
62             lbxDepartment.Items.Add(departments[i]);
63         }
64     }
65
66     professors.Add(new Professor() {
67         DepartmentCode = departments[0].Code,
68         Number = "2020001",
69         Name = "김인하"
70     });
71
72     professors.Add(new Professor() {
73         DepartmentCode = departments[0].Code,
74         Number = "2023003",
75         Name = "김정석"
76     });
77
78
79     professors.Add(new Professor() {
80         DepartmentCode = departments[1].Code,
81         Number = "2023004",
82         Name = "이공전"
83     });
84
85     students.Add("20240001", new Student() {
86         Number = "20240001",
87         Name = "김미영",
88         RegStatus = "재학",
89         Year = 1,
90         BirthInfo = new DateTime(2004, 01, 01),
91         DepartmentCode = "A001",
92         AdvisorNumber = "2020001",
93         Class = "B",
94         Address = "인천 남구 용현동 100",
95         Contact = "032-870-0000"
96     });
97
98     foreach (var student in students) {
99         if (student.Value != null) {
100             lbxDictionay.Items.Add(student.Value);
101         }
102     }
103 }
104
105 private void btnRegisterDepartment_Click(object sender, EventArgs e)
106 {
107     if (string.IsNullOrEmpty(tbxDepartmentCode.Text)) {
108         MessageBox.Show("학과코드 입력");
109         tbxDepartmentCode.Focus();
110         return;
111     }
112 }
```

```
113         if (string.IsNullOrEmpty(tbxDepartmentName.Text)) {
114             MessageBox.Show("학과이름 입력");
115             tbxDepartmentName.Focus();
116             return;
117         }
118
119         int index = -1;
120         for (int i = 0; i < departments.Length; i++) {
121             if (departments[i] == null) {
122                 if (index < 0) {
123                     index = i;
124                 }
125                 //break;
126             } else {
127                 if (departments[i].Code == tbxDepartmentCode.Text) {
128                     MessageBox.Show("중복 학과코드");
129                     tbxDepartmentCode.Focus();
130                     return;
131                 }
132             }
133         }
134
135         if (index < 0) {
136             MessageBox.Show("신규 학과를 개설할 수 없습니다.");
137             return;
138         }
139
140         Department dept = new Department();
141         dept.Code = tbxDepartmentCode.Text;
142         dept.Name = tbxDepartmentName.Text;
143
144         departments[index] = dept;
145
146         lbxDepartment.Items.Add(dept);
147     }
148
149     private void btnRemoveDepartment_Click(object sender, EventArgs e)
150     {
151         if (lbxDepartment.SelectedIndex < 0) {
152             MessageBox.Show("삭제할 학과를 선택");
153             return;
154         }
155
156         if (lbxDepartment.SelectedItem is Department) {
157             var target = (Department)lbxDepartment.SelectedItem;
158
159             for (int i = 0; i < departments.Length; i++) {
160                 if (departments[i] != null && departments[i] == target) {
161                     departments[i] = null;
162                     break;
163                 }
164             }
165
166             lbxDepartment.Items.RemoveAt(lbxDepartment.SelectedIndex);
167             lbxDepartment.SelectedIndex = -1;
168         }
```

```

169     }
170
171     private void tabMain_SelectedIndexChanged(object sender, EventArgs e)
172     {
173         switch (tabMain.SelectedIndex) {
174             case 0:
175                 break;
176             case 1:
177                 cmbProfessorDepartment.Items.Clear();
178                 foreach (var department in departments) {
179                     if (department != null) {
180                         cmbProfessorDepartment.Items.Add(department);
181                     }
182                 }
183
184                 cmbProfessorDepartment.SelectedIndex = -1;
185                 lbxProfessor.Items.Clear();
186                 break;
187             case 2:
188                 cmbDepartment.Items.Clear();
189                 foreach (var department in departments) {
190                     if (department != null) {
191                         cmbDepartment.Items.Add(department);
192                     }
193                 }
194
195                 ClearStudentInfo();
196                 break;
197             case 3:
198                 //(숙제) 성적처리(TEST)의 모든 Control을 지우는 내용을 만 ➤
199                 들 것
200                 ClearTestScoreInfo();
201                 break;
202             default:
203                 break;
204         }
205
206         private void ClearTestScoreInfo()
207         {
208             lblTestName.Text = string.Empty;
209             foreach (var textbox in tbxTestScores) {
210                 textbox.Text = string.Empty;
211             }
212         }
213
214         private void cmbProfessorDepartment_SelectedIndexChanged(object ➤
215             sender, EventArgs e)
216         {
217             if (cmbProfessorDepartment.SelectedIndex < 0) {
218                 return;
219             }
220
221             lbxProfessor.Items.Clear();
222
223             var department = cmbProfessorDepartment.SelectedItem as ➤

```

```

        Department;
223
224         if (department != null) {
225             foreach (var professor in professors) {
226                 if (professor != null && professor.DepartmentCode == department.Code) {
227                     lbxProfessor.Items.Add(professor);
228                 }
229             }
230         }
231     }
232
233     private void btnRegisterProfessor_Click(object sender, EventArgs e)
234     {
235         if (cmbProfessorDepartment.SelectedIndex < 0) {
236             MessageBox.Show("학과를 선택");
237             cmbProfessorDepartment.Focus();
238             return;
239         }
240
241         if (false == cmbProfessorDepartment.SelectedItem is Department dept) {
242             MessageBox.Show("학과정보에 이상 발생");
243             cmbProfessorDepartment.Focus();
244             return;
245         }
246
247         if (string.IsNullOrEmpty(tbxProfessorNumber.Text)) {
248             MessageBox.Show("교수번호 입력");
249             tbxProfessorNumber.Focus();
250             return;
251         }
252
253         if (string.IsNullOrEmpty(tbxProfessorName.Text)) {
254             MessageBox.Show("교수이름 입력");
255             tbxProfessorName.Focus();
256             return;
257         }
258
259         for (int i = 0; i < professors.Count; i++) {
260             if (professors[i].Number == tbxProfessorNumber.Text) {
261                 MessageBox.Show("중복 교수코드");
262                 tbxProfessorNumber.Focus();
263                 return;
264             }
265         }
266
267         Professor professor = new Professor();
268         professor.DepartmentCode = dept.Code;
269         professor.Number = tbxProfessorNumber.Text;
270         professor.Name = tbxProfessorName.Text;
271
272         professors.Add(professor);
273         lbxProfessor.Items.Add(professor);
274     }
275

```

```
276     private void btnRemoveProfessor_Click(object sender, EventArgs e)
277     {
278         if (lbxProfessor.SelectedIndex < 0) {
279             MessageBox.Show("삭제할 교수를 선택");
280             return;
281         }
282
283         if (lbxProfessor.SelectedItem is Professor) {
284             var target = (Professor)lbxProfessor.SelectedItem;
285
286             for (int i = 0; i < professors.Count; i++) {
287                 if (professors[i] != null && professors[i] == target) {
288                     professors.RemoveAt(i);
289                     break;
290                 }
291             }
292
293             lbxProfessor.Items.Remove(target);
294             lbxProfessor.SelectedIndex = -1;
295         }
296     }
297
298     private void cmbDepartment_SelectedIndexChanged(object sender,      ↗
299     EventArgs e)
300     {
301         cmbAdvisor.Items.Clear();
302
303         if (cmbDepartment.SelectedIndex < 0) {
304             return;
305         }
306
307         if (false == cmbDepartment.SelectedItem is Department dept) {
308             return;
309         }
310
311         foreach (var professor in professors) {
312             if (professor != null && professor.DepartmentCode ==      ↗
313                 dept.Code) {
314                 cmbAdvisor.Items.Add(professor);
315             }
316         }
317
318         cmbAdvisor.SelectedIndex = -1;
319
320     }
321
322     private void btnNew_Click(object sender, EventArgs e)
323     {
324         ClearStudentInfo();
325         lbxDictionary.SelectedIndex = -1;
326     }
327
328     private void ClearStudentInfo()
329     {
330         tbxNumber.Text = "20";
331         tbxName.Text = string.Empty;
332         tbxBirthYear.Text = "20";
333     }
```

```

330         tbxBirthMonth.Text = "";
331         tbxBirthDay.Text = "";
332         cmbDepartment.SelectedIndex = -1;
333         cmbAdvisor.SelectedIndex = -1;
334         cmbYear.SelectedIndex = -1;
335         cmbClass.SelectedIndex = -1;
336         cmbRegStatus.SelectedIndex = -1;
337         tbxAddress.Text = string.Empty;
338         tbxContact.Text = string.Empty;
339
340         tbxNumber.ReadOnly = false;
341         selectedStudent = null;
342         btnRegister.Text = "등록";
343     }
344
345     Student selectedStudent = null;
346     private void btnRegister_Click(object sender, EventArgs e)
347     {
348         if (selectedStudent == null) {
349             RegisterStudent();
350         } else {
351             UpdateStudent();           //call
352         }
353     }
354
355     private void RegisterStudent()
356     {
357         var number = tbxNumber.Text.Trim();
358
359         if (string.IsNullOrEmpty(number) || number.Length != 8) {
360             tbxNumber.Focus();
361             return;
362         }
363
364         if (string.IsNullOrEmpty(tbxName.Text) || tbxName.Text.Trim
365             ().Length < 2) {
366             tbxName.Focus();
367             return;
368         }
369
370         //for, 성능 떨어짐
371         for (int i = 0; i < students.Count; i++) {
372             var pair = students.ElementAt(i);
373             if (pair.Key == number) {
374                 tbxNumber.Focus();
375                 return;
376             }
377         }
378
379         //foreach
380         foreach (var pair in students) {
381             if (pair.Key == number) {
382                 tbxNumber.Focus();
383                 return;
384             }
385         }

```

```
385
386     //실제 많이 사용하는 방법1
387     if (true == students.ContainsKey(number)) {
388         tbxNumber.Focus();
389         return;
390     }
391
392     Student student = new Student();
393     student.Number = number;
394     student.Name = tbxName.Text.Trim();
395
396     int birthYear, birthMonth; // birthDay;
397     if (true == int.TryParse(tbxBirthYear.Text, out birthYear)) {
398         if (birthYear < 1900 || 9000 < birthYear) {
399             tbxBirthYear.Focus();
400             return;
401         }
402     } else {
403         tbxBirthYear.Focus();
404         return;
405     }
406
407     if (true == int.TryParse(tbxBirthMonth.Text, out birthMonth)) {
408         if (birthMonth < 1 || 12 < birthMonth) {
409             tbxBirthMonth.Focus();
410             return;
411         }
412     } else {
413         tbxBirthMonth.Focus();
414         return;
415     }
416
417     if (true == int.TryParse(tbxBirthDay.Text, out int birthDay)) {
418         //2월, 달 처리등은 추후 해볼것
419         if (birthDay < 0 || 31 < birthDay) {
420             tbxBirthDay.Focus();
421             return;
422         }
423     } else {
424         tbxBirthDay.Focus();
425         return;
426     }
427
428     student.BirthInfo = new DateTime(birthYear, birthMonth, birthDay);
429
430     if (cmbDepartment.SelectedIndex < 0) {
431         //cmbDepartment.Focus();
432         //return;
433         student.DepartmentCode = null;
434     } else {
435         student.DepartmentCode
436             = (cmbDepartment.SelectedItem as Department).Code;
437     }
438
439     if (cmbAdvisor.SelectedIndex < 0) {
440         student.AdvisorNumber = null;
```



```
441         } else {
442             student.AdvisorNumber = (cmbAdvisor.SelectedItem as
443                 Professor).Number;
444         }
445         if (false == int.TryParse(cmbYear.SelectedItem.ToString(), out
446             student.Year)) {
447             cmbYear.Focus();
448             return;
449         }
450         if (cmbClass.SelectedIndex < 0) {
451             cmbClass.Focus();
452             return;
453         }
454         student.Class = cmbClass.SelectedItem.ToString();
455
456         if (cmbRegStatus.SelectedIndex < 0) {
457             cmbRegStatus.Focus();
458             return;
459         }
460         student.RegStatus = cmbRegStatus.SelectedItem.ToString();
461
462         student.Address = tbxAddress.Text.Trim();
463         student.Contact = tbxContact.Text.Trim();
464
465         students[number] = student;
466         lbxDictionary.Items.Add(student);
467     }
468
469     private void UpdateStudent() //define
470     {
471         if (string.IsNullOrEmpty(tbxName.Text) || tbxName.Text.Trim
472             ().Length < 2) {
473             tbxName.Focus();
474             return;
475         }
476         if (true == int.TryParse(tbxBirthYear.Text, out int birthYear)) {
477             if (birthYear < 1900 || 9000 < birthYear) {
478                 tbxBirthYear.Focus();
479                 return;
480             }
481         } else {
482             tbxBirthYear.Focus();
483             return;
484         }
485
486         if (true == int.TryParse(tbxBirthMonth.Text, out int birthMonth))
487         {
488             if (birthMonth < 1 || 12 < birthMonth) {
489                 tbxBirthMonth.Focus();
490                 return;
491             }
492         } else {
```

```

493         tbxBirthMonth.Focus();
494         return;
495     }
496
497     if (true == int.TryParse(tbxBirthDay.Text, out int birthDay)) {
498         //2월, 달 처리등은 추후 해볼것
499         if (birthDay < 0 || 31 < birthDay) {
500             tbxBirthDay.Focus();
501             return;
502         }
503     } else {
504         tbxBirthDay.Focus();
505         return;
506     }
507
508     if (cmbDepartment.SelectedIndex < 0) {
509         //cmbDepartment.Focus();
510         return;
511     } else {
512
513     }
514
515     if (cmbYear.SelectedIndex < 0) {
516         cmbYear.Focus();
517         return;
518     }
519
520     if (false == int.TryParse(cmbYear.SelectedItem.ToString(), out
521         int year)) {
522         cmbYear.Focus();
523         return;
524     }
525
526     if (cmbClass.SelectedIndex < 0) {
527         cmbClass.Focus();
528         return;
529     }
530
531     if (cmbRegStatus.SelectedIndex < 0) {
532         cmbRegStatus.Focus();
533         return;
534     }
535
536     //모든 사항의 확인 및 수정이 끝나면
537     //수정완료 메시지를 띄운 후
538     //화면을 초기화 상태로 만든다.
539
540     selectedStudent.Name = tbxName.Text.Trim();
541     selectedStudent.BirthInfo = new DateTime(birthYear, birthMonth,
542         birthDay);
543
544     if (cmbDepartment.SelectedIndex < 0) {
545         selectedStudent.DepartmentCode = null;
546     } else {
547         selectedStudent.DepartmentCode = (cmbDepartment.SelectedItem

```

```

        as Department).Code;
547     }
548
549     if (cmbAdvisor.SelectedIndex < 0) {
550         selectedStudent.AdvisorNumber = null;
551     } else {
552         selectedStudent.AdvisorNumber = (cmbAdvisor.SelectedItem as  ➤
            Professor).Number;
553     }
554
555     selectedStudent.Year = year;
556     selectedStudent.Class = cmbClass.SelectedItem.ToString();
557     selectedStudent.RegStatus = cmbRegStatus.SelectedItem.ToString();
558     selectedStudent.Address = tbxAddress.Text.Trim();
559     selectedStudent.Contact = tbxContact.Text.Trim();
560
561     MessageBox.Show("수정 완료");
562     lbxDictionary.SelectedIndex = -1;
563     ClearStudentInfo();
564 }
565
566 private void lbxDictionary_SelectedIndexChanged(object sender,  ➤
    EventArgs e)
567 {
568     if (lbxDictionary.SelectedIndex < 0) {
569         return;
570     }
571
572     var student = (lbxDictionary.SelectedItem as Student);
573
574     ClearStudentInfo();
575
576     if (student != null) {
577         DisplaySelectedStudent(student);
578     }
579 }
580
581 private void DisplaySelectedStudent(Student student)
582 {
583     selectedStudent = student;
584     tbxNumber.ReadOnly = true;
585
586     tbxNumber.Text = student.Number;
587     tbxName.Text = student.Name;
588     tbxBirthYear.Text = student.BirthInfo.Year.ToString();
589     tbxBirthMonth.Text = student.BirthInfo.Month.ToString();
590     tbxBirthDay.Text = student.BirthInfo.Day.ToString();
591
592     for (int i = 0; i < cmbDepartment.Items.Count; i++) {
593         if ((cmbDepartment.Items[i] as Department).Code
594             == student.DepartmentCode) {
595             cmbDepartment.SelectedIndex = i;
596             break;
597         }
598     }
599

```

```
600         for (int i = 0; i < cmbAdvisor.Items.Count; i++) {
601             if ((cmbAdvisor.Items[i] as Professor).Number
602                 == student.AdvisorNumber) {
603                 cmbAdvisor.SelectedIndex = i;
604                 break;
605             }
606         }
607
608         for (int i = 0; i < cmbYear.Items.Count; i++) {
609             if (cmbYear.Items[i].ToString() == student.Year.ToString()) {
610                 cmbYear.SelectedIndex = i;
611                 break;
612             }
613         }
614
615         for (int i = 0; i < cmbClass.Items.Count; i++) {
616             if (cmbClass.Items[i].ToString() == student.Class) {
617                 cmbClass.SelectedIndex = i;
618                 break;
619             }
620         }
621
622         for (int i = 0; i < cmbRegStatus.Items.Count; i++) {
623             if (cmbRegStatus.Items[i].ToString() == student.RegStatus) {
624                 cmbRegStatus.SelectedIndex = i;
625                 break;
626             }
627         }
628
629         tbxAddress.Text = selectedStudent.Address;
630         tbxContact.Text = selectedStudent.Contact;
631
632         btnRegister.Text = "수정";
633     }
634
635     private void btnTestSearchStudent_Click(object sender, EventArgs e)
636     {
637         ClearTestScoreInfo();
638
639         var number = tbxTestNumber.Text.Trim();
640         if (string.IsNullOrEmpty(number) || number.Length != 8) {
641             tbxNumber.Focus();
642             return;
643         }
644
645         selectedStudent = SearchStudentByNumber(tbxTestNumber.Text);
646
647         if (selectedStudent == null) {
648             MessageBox.Show($"{number}학번의 학생 정보가 없음");
649             tbxTestNumber.Focus();
650             return;
651         }
652
653         lblTestName.Text = selectedStudent.Name;
654     }
655
```

```
656     private Student SearchStudentByNumber(string number)
657     {
658         if (students.ContainsKey(number)) {
659             return students[number];
660         } else {
661             return null;
662         }
663     }
664 }
665 }
666 }
```