

# Design

T8

**200310548 이정우**  
**200911372 김민하**

**200911364 곽수미**  
**200911373 김바울**

# 목차

---

- ▶ 2141 Design Real Use Case
- ▶ 2142 Define Reports, UI, and Storyboards
- ▶ 2143 Refine System Architecture
- ▶ 2144 Define Interaction Diagrams
- ▶ 2145 Define Design Class Diagrams
- ▶ 2146 Define Database Schema



# 2141 Design Real Use Case

---

- 사용자 인증

Use Case	사용자 인증
Actor	학생, 교수
Purpose	포탈에 가입이 되어 있어야 한다.
Overview	사용자는 아이디와 비밀번호를 입력하여 로그인을 누른다.
Type	
Cross Reference	Function : R1, R8 Use Cases : Log In use case
Pre-Requisites	N/A
UI Widgets	Window <그림 1>
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 프로그램을 실행시킨다. 2. (A) 아이디와 비밀번호를 입력하여 로그인 버튼을 누른다. 3. (S) 시스템이 서버에 아이디와 비밀번호가 정확한지 검색하여 사용자가 확인되면 시간표 서비스를 실행하게 된다.
Alternative Courses of Events	...
Exceptional Courses of Events	사용자 인증에 실패할 경우 에러메시지를 출력하고 로그인 화면으로 넘어간다.



# 2141 Design Real Use Case

---

- join(회원 가입)

Use Case	join
Actor	사용자
Purpose	Join 버튼을 눌러 가입 창을 띄운다.
Overview	사용자가 시스템에 가입을 한다.
Type	
Cross Reference	Function : ... Use Cases : join use case
Pre-Requisites	N/A
UI Widgets	Window <그림 1>, <그림 2>
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 아이디를 입력하고 중복확인 버튼을 누른다. 2. (S) 입력받은 아이디를 전체 사용자 계정에서 있는지 확인하고 메시지를 출력한다. 3. (A) 사용가능 하다는 메시지를 출력할 경우 비밀번호와 비밀번호확인에 비밀번호를 입력하고 학생 또는 교수란에 체크를 하고 학생의 경우 학번과 이름을 입력하고 교수의 경우 교수의 이름만 입력 받는다. 4. (A) 아이디 생성버튼을 누른다. 5. (S) 아이디가 중복된 것인지 체크하고 비밀번호와 비밀번호확인이 같은지를 체크한 후에 아이디를 생성하여 저장한다.
Alternative Courses of Events	...
Exceptional Courses of Events	...



# 2141 Design Real Use Case

---

- 강의 조회

Use Case	강의 조회
Actor	학생
Purpose	학생 로그인, 서버와의 연결이 되어있어야 한다.
Overview	학생은 강의 번호를 입력하여 강의를 조회한다.
Type	
Cross Reference	Function : R2 Use Cases : 학생시간표 서비스 use case
Pre-Requisites	N/A
UI Widgets	Window <그림 3>, <그림 4>
Typical Courses of Events	(A) :Actor, (S) :System 1. (A) 강의번호 란에 강의 번호를 입력한다. 강의 조회를 눌러서 강의 조회를 시작한다 2. (S) 강의 번호를 검색하여 강의 리스트화면에 출력한다.
Alternative Courses of Events	...
Exceptional Courses of Events	강의 번호가 없을 경우 에러메시지를 출력한다.



# 2141 Design Real Use Case

---

- 강의 저장

Use Case	강의 저장
Actor	학생
Purpose	학생 로그인, 서버와의 연결이 되어있어야 한다. 화면에 강의 리스트가 출력되어 있어야 한다.
Overview	출력된 강의를 선택하여 저장한다.
Type	
Cross Reference	Function : R3 Use Cases : 학생시간표 서비스 use case
Pre-Requisites	N/A
UI Widgets	Window <그림 3>, <그림 4>
Typical Courses of Events	(A) :Actor, (S) :System 1. (A) 강의 리스트에 출력된 강의에서 하나를 선택한 후 강의 저장버튼을 눌러 강의를 저장한다. 2. (S) 저장하려는 강의를 현재 저장된 강의 화면에 갱신한다. 3. (S) 현재 저장된 강의의 나타난 강의 정보를 DB에 저장한다.
Alternative Courses of Events	...
Exceptional Courses of Events	강의가 이미 저장되어 있을 경우 에러메시지를 출력한다.



# 2141 Design Real Use Case

---

- 강의 삭제

Use Case	강의 삭제
Actor	학생
Purpose	학생 로그인, 서버와의 연결이 되어있어야 한다. 강의가 저장되어 있어야 한다.
Overview	선택된 강의를 삭제한다.
Type	
Cross Reference	Function : R4 Use Cases : 학생시간표 서비스 use case
Pre-Requisites	N/A
UI Widgets	Window <그림 3>, <그림 4>
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 현재 저장된 강의 화면에 출력된 현재 저장된 강의에서 삭제하려는 강의를 선택한 후 강의 삭제버튼을 눌러 강의를 삭제한다. 2. (S) 삭제한 강의를 현재 저장된 강의 화면에서 제거한다. 3. (S) 현재 저장된 강의 화면에 나타난 강의 정보를 DB에 저장한다.
Alternative Courses of Events	...
Exceptional Courses of Events	...



# 2141 Design Real Use Case

---

- 시간표 출력

Use Case	시간표 출력
Actor	학생
Purpose	학생 로그인, 서버와의 연결이 되어있어야 한다. 강의가 저장되어 있어야 한다.
Overview	저장되어 있는 시간표를 출력한다.
Type	
Cross Reference	Function : R5 Use Cases : 학생시간표 서비스 use case
Pre-Requisites	N/A
UI Widgets	Window <그림 3>, <그림 5>
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 시간표 출력버튼을 눌러 현재 저장되어 있는 시간표를 출력한다. . 2. (S) 저장되어 있는 시간표 정보를 서버에 요청하여 전송 받은 시간표 정보를 화면에 출력한다.
Alternative Courses of Events	...
Exceptional Courses of Events	...



# 2141 Design Real Use Case

---

- 친구 찾기

Use Case	친구 찾기
Actor	학생
Purpose	학생 로그인, 서버와의 연결이 되어있어야 한다. 강의가 저장되어 있어야 한다.
Overview	친구의 학번을 입력하여 나의 시간표에서 같이 듣는 시간을 출력한다.
Type	
Cross Reference	Function : R6, R7 Use Cases : 학생시간표 서비스 use case, 학생부가기능 use case
Pre-Requisites	N/A
UI Widgets	Window <그림 3>, <그림 6>
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 친구 찾기 버튼을 눌러 부가기능 창을 띄운다. 2. (A) 친구 학번 입력란에 친구의 학번을 입력하고 친구찾기 버튼을 눌러 친구찾기를 시작한다. 3. (S) 입력 받은 학번의 시간표와 나의 시간표를 비교하여 같은 강의 시간을 찾는다. 4. (S) 같은 강의 시간을 화면에 출력한다.
Alternative Courses of Events	...
Exceptional Courses of Events	...



# 2141 Design Real Use Case

---

- 강의 시간 조회

Use Case	강의시간조회
Actor	교수
Purpose	교수 로그인, 서버와의 연결이 되어있어야 한다.
Overview	자신이 강의하는 시간을 조회한다.
Type	
Cross Reference	Function : R10 Use Cases : 교수시간표 서비스 use case
Pre-Requisites	N/A
UI Widgets	Window <그림 7>, <그림 8>
Typical Courses of Events	(A) :Actor, (S) :System 1. (A) 강의 시간 조회버튼을 눌러 강의시간을 조회한다. 2. (S) 현재 교수가 강의하는 강의시간을 검색하여 화면에 출력한다.
Alternative Courses of Events	...
Exceptional Courses of Events	...



# 2141 Design Real Use Case

---

## - 학생 조회

Use Case	학생조회
Actor	교수
Purpose	교수 로그인, 서버와의 연결이 되어있어야 한다. 현재 자신이 강의하는 강의를 듣는 학생이어야 한다.
Overview	현재 자신이 하는 강의를 듣는 학생들을 출력한다.
Type	
Cross Reference	Function : R9 Use Cases : 교수시간표 서비스 use case
Pre-Requisites	N/A
UI Widgets	Window <그림 7>, <그림 9>
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 학생 조회버튼을 눌러 학생을 조회한다. 2. (S) 현재 자신이 하는 강의를 듣는 학생의 정보를 검색하여 화면에 출력한다.
Alternative Courses of Events	...
Exceptional Courses of Events	...



# 2141 Design Real Use Case

---

- 시간표 조회

Use Case	시간표조회
Actor	교수
Purpose	교수 로그인, 서버와의 연결이 되어있어야 한다. <그림 7>의 시험시간 설정버튼을 누른다.
Overview	학생들의 시간표를 조회하여 빈 강의시간을 출력한다.
Type	
Cross Reference	Function : R I I Use Cases : 교수시간표 서비스 use case, 교수부가기능 use case
Pre-Requisites	N/A
UI Widgets	Window <그림 7>, <그림 10>
Typical Courses of Events	(A) :Actor, (S) :System 1. (A) 시간표 조회버튼을 눌러 시간표를 조회한다. 2. (S) 자신의 강의를 듣는 학생들의 시간표를 검색하여 빈 강의 시간을 찾아 화면에 출력한다.
Alternative Courses of Events	...
Exceptional Courses of Events	...



# 2141 Design Real Use Case

---

- 시험 시간 입력

Use Case	시험시간입력
Actor	교수
Purpose	교수 로그인, 서버와의 연결이 되어있어야 한다. <그림 7>의 시간표 조회버튼을 누른다. 시간표 조회버튼을 눌러서 시간표가 조회되어 있어야 한다.
Overview	출력된 시간표에서 빈 강의시간을 선택하여 시험시간으로 저장한다.
Type	
Cross Reference	Function : R12 Use Cases : 교수시간표 서비스 use case, 교수부가기능 use case
Pre-Requisites	N/A
UI Widgets	Window <그림 7>, <그림 10>
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 화면에 출력된 시간표에서 빈 강의 시간을 선택한 후 시험시간 입력버튼을 눌러서 시험시간을 저장한다. 2. (S) 선택한 시험시간을 서버에 저장한 뒤 화면을 갱신한다.
Alternative Courses of Events	...
Exceptional Courses of Events	선택한 시간이 다른 시간에 겹쳐져 있을 경우 에러메시지를 출력한다.



# 2141 Design Real Use Case

---

- 시험 시간 삭제

Use Case	시험시간삭제
Actor	교수
Purpose	교수 로그인, 서버와의 연결이 되어있어야 한다. 화면에 시험 시간표가 있어야 하고 시험시간이 등록되어 있어야 한다.
Overview	선택한 시험시간을 삭제한다.
Type	
Cross Reference	Function : R13 Use Cases : 교수시간표 서비스 use case, 교수부가기능 use case
Pre-Requisites	N/A
UI Widgets	Window <그림 7>, <그림 10>
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 화면에 출력된 시간표에서 시험시간을 선택한 후 시험시간 삭제 버튼을 눌러 시험시간을 삭제한다. 2. (S) 선택된 시험시간을 삭제하고 현재의 상태를 서버에 저장한 뒤 화면을 갱신한다.
Alternative Courses of Events	...
Exceptional Courses of Events	...



# 2141 Design Real Use Case

---

- 전단계

Use Case	전 단계
Actor	사용자
Purpose	로그인이 되어있어야 한다.
Overview	현재의 서비스 사용에서 메뉴 선택 화면으로 이동한다.
Type	
Cross Reference	Function : R14 Use Cases : 전 단계 use case
Pre-Requisites	N/A
UI Widgets	Window <그림 4> ~<그림 10>
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 전 단계 버튼을 누른다. 2. (S) 각 사용자에게 맞는 서비스 메뉴 화면을 출력한다.
Alternative Courses of Events	...
Exceptional Courses of Events	...



# 2141 Design Real Use Case

---

- 로그아웃

Use Case	로그아웃
Actor	사용자
Purpose	로그인이 되어 있어야 한다.
Overview	사용자의 정보를 DB에 저장한 후 접속을 종료하고 로그인 화면을 띄운다.
Type	
Cross Reference	Function : R15 Use Cases : 로그아웃 use case
Pre-Requisites	N/A
UI Widgets	Window <그림 4> ~<그림 10>
Typical Courses of Events	(A) :Actor, (S) :System 1. (A) 로그아웃버튼을 누른다. 2. (S) 현재의 정보를 DB에 저장하고 접속을 해제한다. 3. (S) 로그인 화면을 출력한다.
Alternative Courses of Events	...
Exceptional Courses of Events	...



# 2142 Define Reports, UI, and Storyboards

학생 시간표 시스템 **그림1**

아이디   D

비밀번호   E

학생  교수 C

## <그림 1> 전체 프로그램 초기화면

1. 사용자가 이미 account가 있는 경우,  
A에 사용자의 id, B에는 password를 입력하고 사용자가 학생과 교수 중에 선택하고 로그인한다.
2. Account가 없는 경우 경우에는 D를 클릭해 새 계정을 만든다.

학생 시간표 시스템 **그림2**

아이디   D2

비밀번호  D2

비밀번호 재확인  D5

학생  교수

학생  
학번

이름

D8

## <그림 2> 회원가입

사용자는 D1에 새로운 아이디를 적은 후, D2를 클릭하여 중복확인을 한다. 이 때, 다른 사용자의 아이디와 같을 경우 새로운 아이디를 만들라는 화면 창을 띄운다. 중복확인을 통하여 사용할 수 있는 아이디일 경우에는, D3에 비밀번호를 지정한 후, D4에 재확인을 위해 다시 한 번 적는다. 학생이냐 교수이냐에 따라 D5에 체크한다. 학생일 경우, D6에는 학번을, D7에는 이름을 적은 후, D8을 클릭하여 가입을 마친다.

# 2142 Define Reports, UI, and Storyboards

---



## <그림 3>학생 시간표 시스템 – 초기화면

- ▶ 학생으로 가입 또는 로그인을 하면 <그림 3>과 같은 창이 뜬다.
- ▶ 사용자는 A를 클릭하여 강의 조회, B를 클릭하여 시간표 출력, C를 클릭하여 친구 찾기 기능을 실행할 수 있다.
- ▶ 이 프로그램을 종료하기 위해서는 D를 클릭하여 로그아웃 한다.



# 2142 Define Reports, UI, and Storyboards



## <그림 4> 학생 시간표 시스템 - 강의 조회

- ▶ <그림 3>에서 A(강의 조회)를 클릭했을 경우, <그림 4>와 같은 창이 뜬다. A1의 화살표를 클릭하면, 오른쪽과 같이 '강의번호', '교과목' 이라는 2개의 리스트가 나타나는데, 사용자의 편의에 맞게 선택한다. A2에는 사용자의 선택이 '강의번호'일 경우에는 해당 강의의 번호를, '교과목'일 경우에는 해당 강의의 이름을 적은 후, A3을 클릭한다. 그러면, A4에 해당 강의의 내역이 나타난다.
- ▶ 사용자가 A5를 클릭하여 A4에 나타나있는 과목을 저장할 경우, 사용자의 데이터베이스에 저장되며 A9에 나타난다. 또한, A6을 클릭하면 해당 과목이 삭제되면서 데이터베이스와 A9에서도 지워진다. 이때 정말로 지울 것인지를 묻는 경고문을 띄운다.
- ▶ A7을 클릭하여 시간표를 출력하거나 A8을 클릭하여 친구 찾기 기능을 실행할 수 있다.
- ▶ A10을 클릭하면 전 단계로, A11을 클릭하면 로그아웃 되어 프로그램을 마친다.

# 2142 Define Reports, UI, and Storyboards

---



## <그림 5> 학생 시간표 시스템 - 시간표 출력

- ▶ <그림 3>에서 B(시간표 출력)를 클릭하거나 <그림 4>에서 A7(시간표 출력)을 클릭했을 경우, <그림 5>와 같은 창이 뜬다. B2를 클릭하면 사용자의 데이터베이스에서 불러온 시간표의 내용이 B1에 나타난다.



# 2142 Define Reports, UI, and Storyboards

---

학생 부가기능 **그림6** 로그아웃

친구 학번  친구찾기 C2

교과목	이름	시간
C3		

## <그림 6> 부가기능 - 친구 찾기

- ▶ <그림 3>에서 C(친구 찾기)를 클릭하거나 <그림 4>에서 A8(친구 찾기)을 클릭했을 경우, <그림 6>과 같은 창이 뜬다. C1에 친구의 학번을 입력한 후, C2를 클릭하면 C3에는 사용자와 같은 시간에 같은 수업을 듣는 친구의 이름과 그 과목의 이름, 강의 시간이 나타난다.



# 2142 Define Reports, UI, and Storyboards

---



## <그림 7> 교수 시간표 시스템 – 초기화면

- ▶ 교수로 가입 또는 로그인이 끝나면 <그림 7>과 같은 창이 뜬다. A를 클릭하여 강의 시간 조회 기능을, B를 클릭하여 학생 조회 기능을, C를 클릭하여 시험 시간 설정 기능을 실행한다.



# 2142 Define Reports, UI, and Storyboards

---



## <그림 8> 교수 시간표 시스템 - 강의 시간 조회

- ▶ 사용자가 <그림 7>에서 A(강의 시간 조회)를 클릭했을 경우, <그림 8>과 같은 창이 뜬다. A1을 클릭하면 데이터베이스에서 읽어와 사용자가 해당하는 강의 시간이 A2에 나타난다.
- ▶ A3을 클릭하면 전 단계로 이동하며, A4를 클릭하면 로그아웃 할 수 있다.



# 2142 Define Reports, UI, and Storyboards

---



## <그림 9> 교수 시간표 시스템 - 학생 조회

- ▶ 사용자가 <그림 7>에서 B(학생 조회)를 클릭했을 경우, <그림 9>와 같은 창이 뜬다. B1을 클릭하면, 데이터베이스에서 읽어와 사용자가 강의하는 과목을 듣는 학생들의 정보가 B2에 나타난다. (과목은 하나 이상)



# 2142 Define Reports, UI, and Storyboards

---

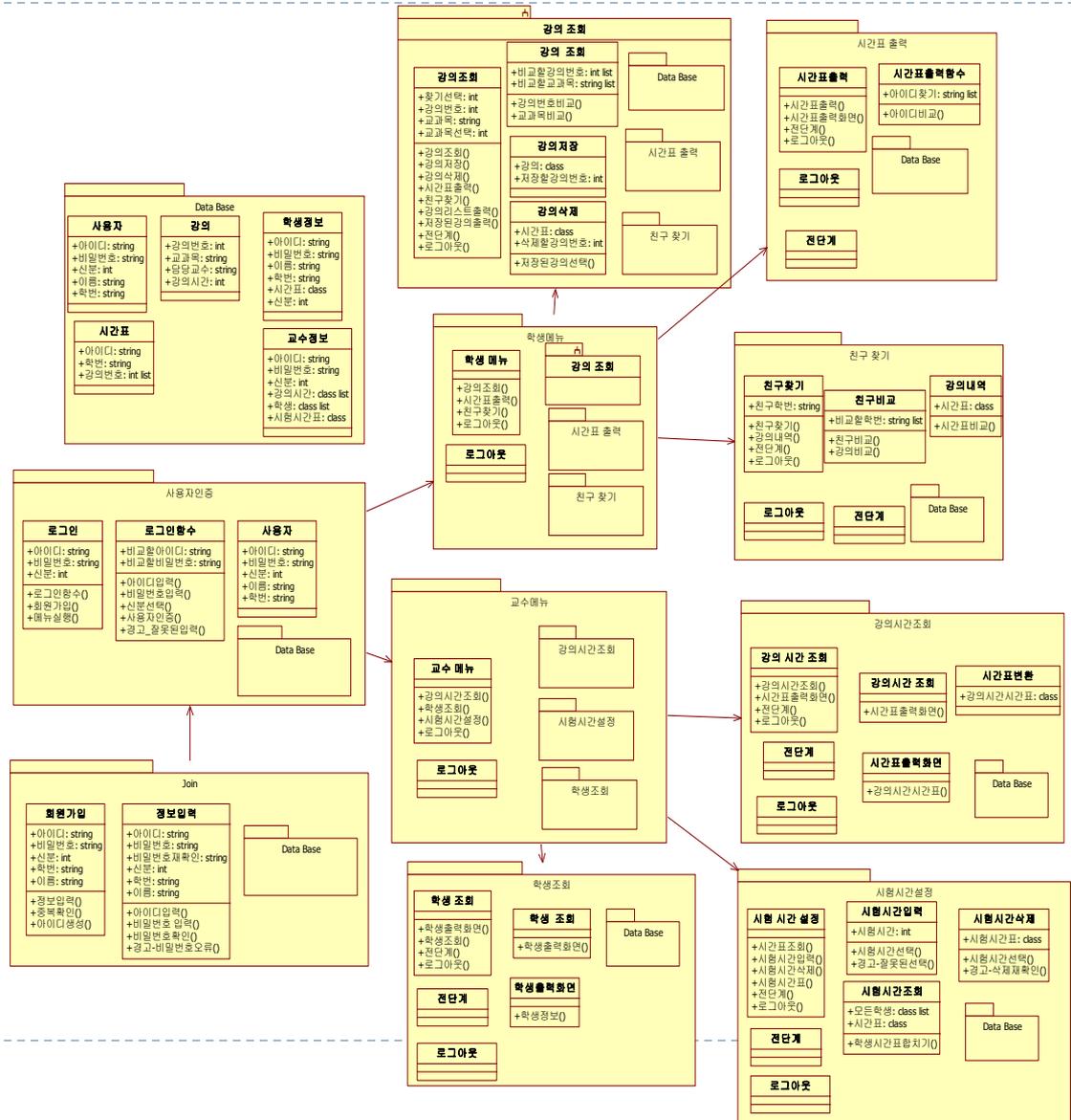


## <그림 10> 교수 부가 기능 - 시험 시간 설정

- ▶ 사용자가 <그림 7>에서 D(시험 시간 설정)를 클릭했을 경우, <그림 10>과 같은 창이 뜬다. D1을 클릭하면 사용자의 강의를 듣는 학생들의 시간표를 조회할 수 있고, 각 학생들의 시간표를 비교한 내용이 D4에 나타난다. D2를 클릭하여 정한 시험 시간을 입력하면 데이터베이스에 저장되고 D4에 그 내용이 나타나며, D4을 클릭하여 정한 시험 시간을 삭제할 경우에는 데이터베이스에서도 삭제되며 그 내용이 D4에 나타난다. 이때 정말로 지울 것인지를 묻는 경고문을 띄운다.



# 2143 Refine System Architecture



# 2143 Refine System Architecture

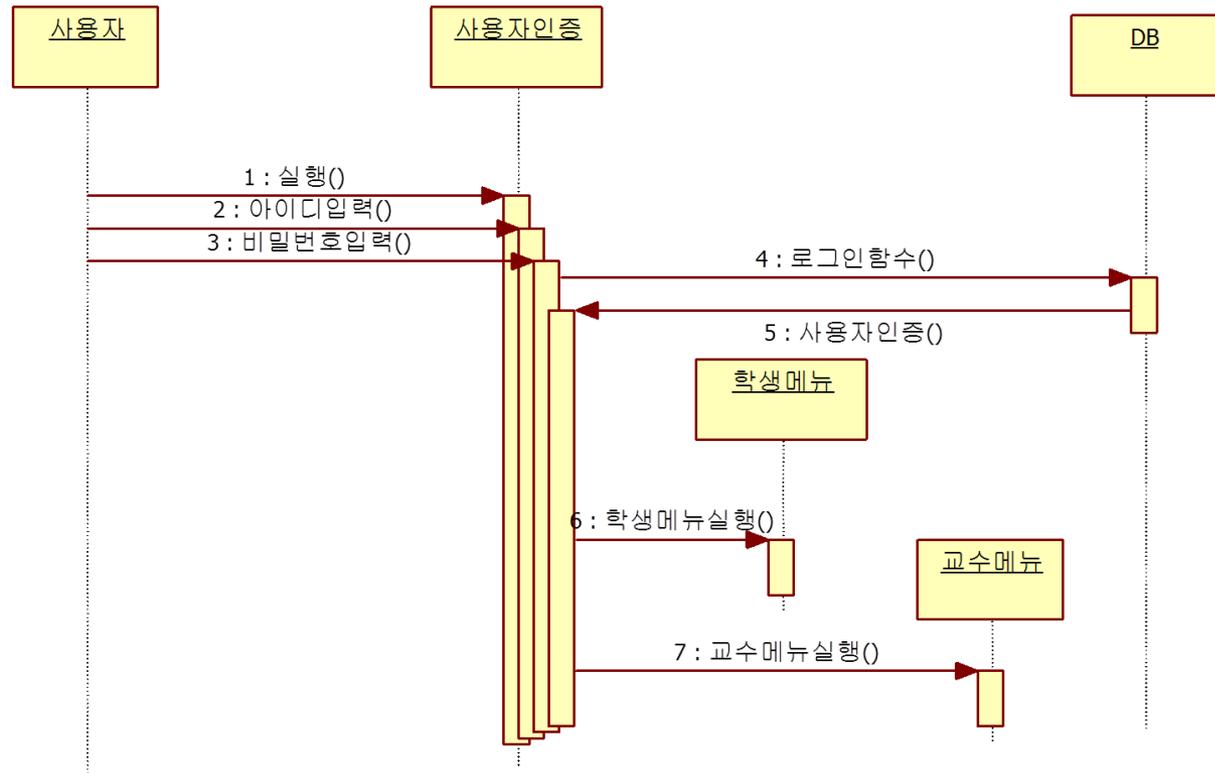
---

- ▶ 패키지 다이어그램
- ▶
- ▶ 패키지 다이어그램의 패키지들로는 Data Base, 사용자인증, Join, 학생메뉴, 강의조회, 시간표출력, 친구 찾기, 교수 메뉴, 강의시간조회, 시험시간설정, 학생조회가 있으며 각각의 패키지마다 그림에서 보는 것과 같이 각각의 class 와 함수 또는 같은 패키지들을 포함하고 있습니다.
- ▶ 먼저 시스템을 사용하기 위해서 사용자 인증을 해야 하는데 회원가입이 되어있지 않다면 먼저 회원 가입을 하여야 사용자 인증을 할 수 있습니다. 사용자 인증 후 사용자의 신분에 따라서 학생메뉴와 교수메뉴로 나뉘지게 되는데 먼저 학생메뉴에는 강의조회, 시간표출력, 친구 찾기의 패키지와 로그아웃 함수가 포함되어있어 여기서의 선택에 따라서 기능을 실행하거나 종료할 수 있습니다. 강의 조회, 시간표 출력, 친구 찾기의 기능은 학생메뉴를 통하여야만이 실행시킬 수 있습니다. 메뉴를 실행하게 되면 메뉴에 따라서 여러 가지 기능이 실행되고 전 단계라는 함수가 각각의 패키지에 포함되어있는데 이 함수를 통하여 메뉴선택으로 돌아갈 수 있습니다. 교수메뉴에는 강의시간조회, 시험시간설정, 학생조회 패키지와 로그아웃 함수가 포함되어있어 여기서도 선택에 따라서 기능을 실행하거나 종료할 수 있습니다. 강의시간조회, 시험시간설정, 학생조회 기능은 교수메뉴를 통하여야만이 실행시킬 수 있고 메뉴를 실행하게 되면 각각의 메뉴에 따라서 각각의 기능이 실행되고 여기서도 마찬가지로 포함된 전 단계 함수를 실행하므로 교수메뉴선택 화면으로 이동하게 됩니다.



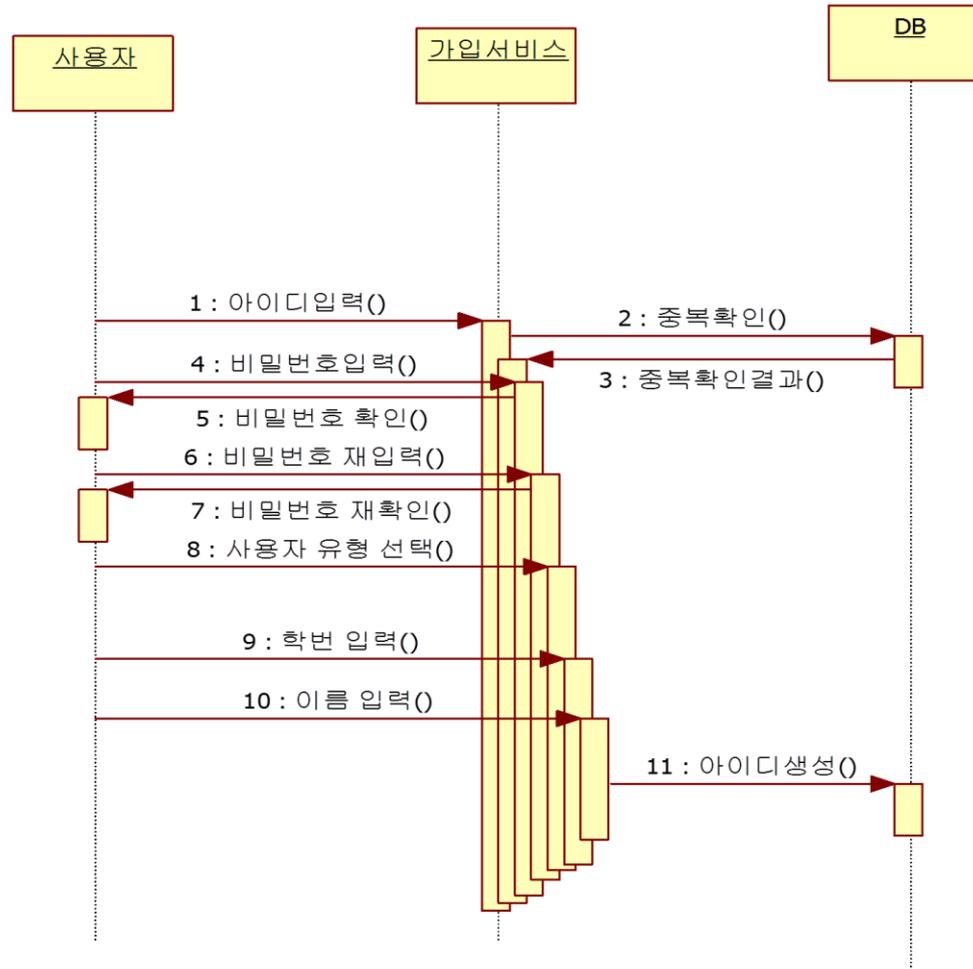
# 2144 Define Interaction Diagrams

- 로그인



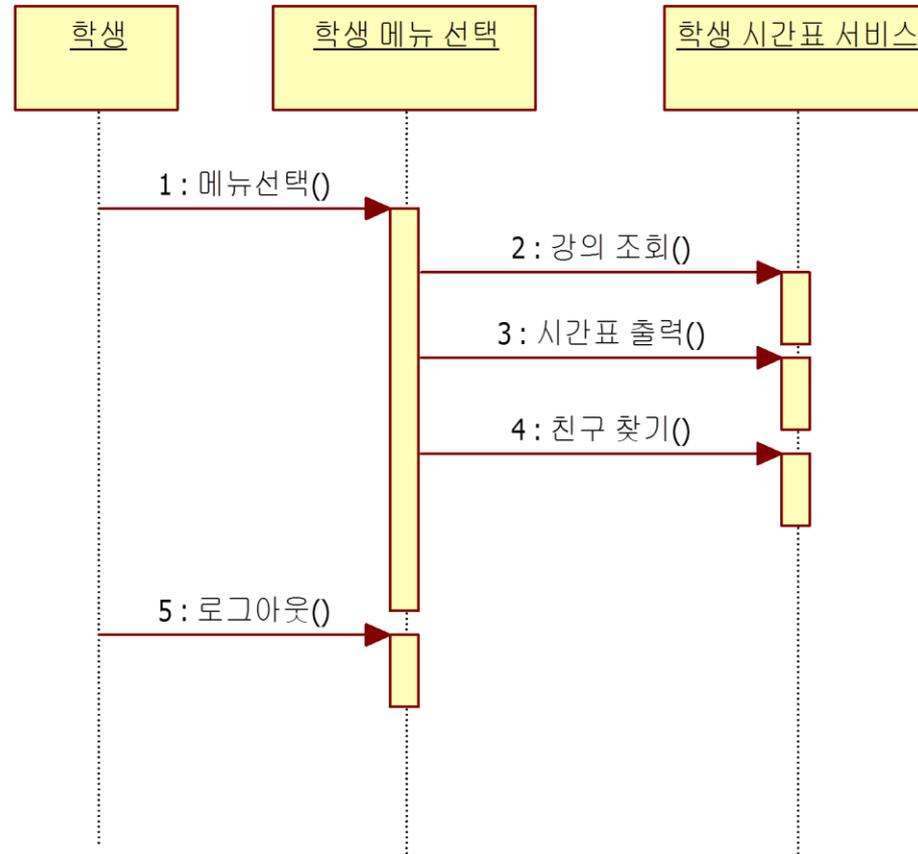
# 2144 Define Interaction Diagrams

- join



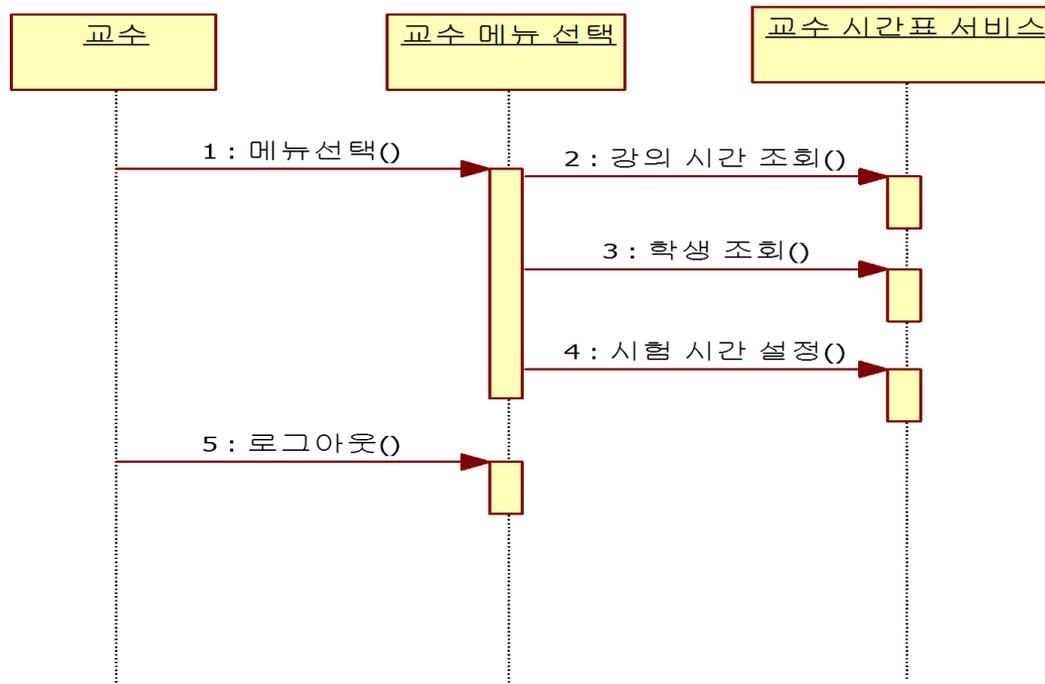
# 2144 Define Interaction Diagrams

- 학생 메뉴



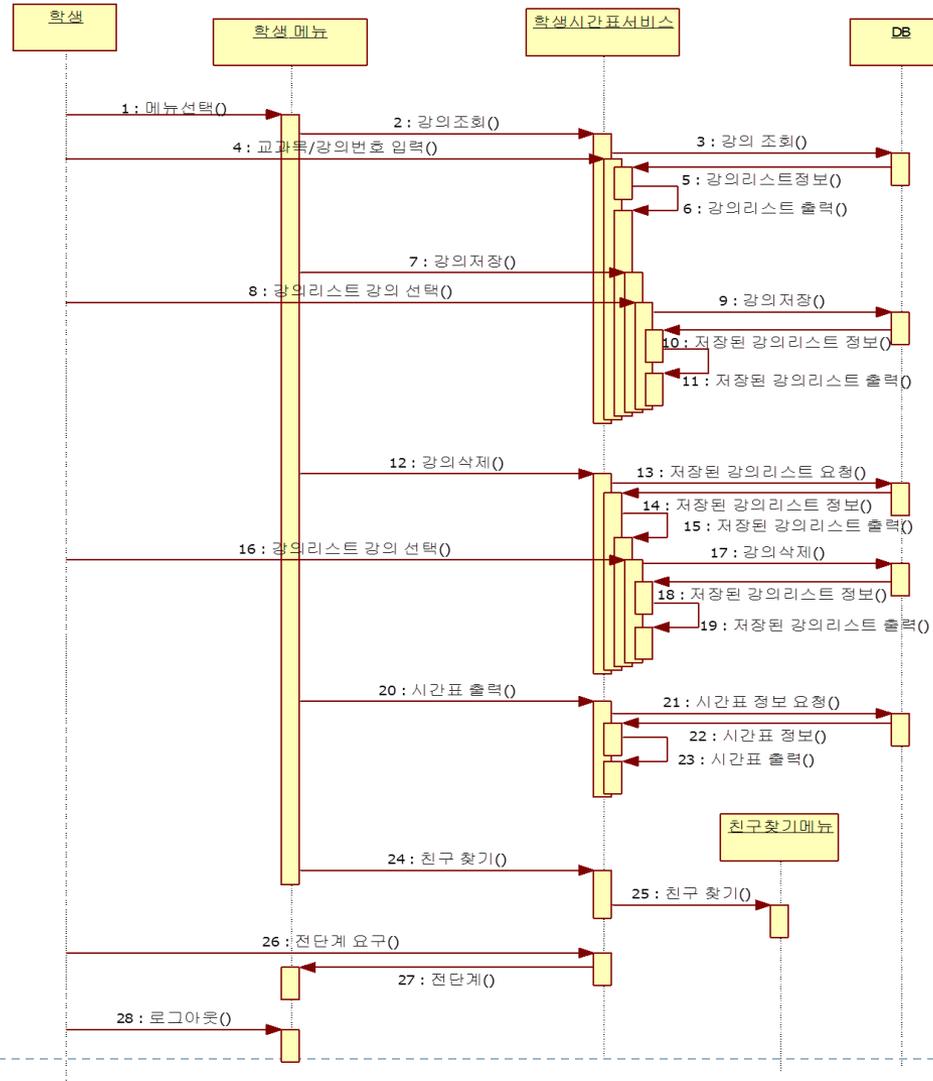
# 2144 Define Interaction Diagrams

- 교수 메뉴



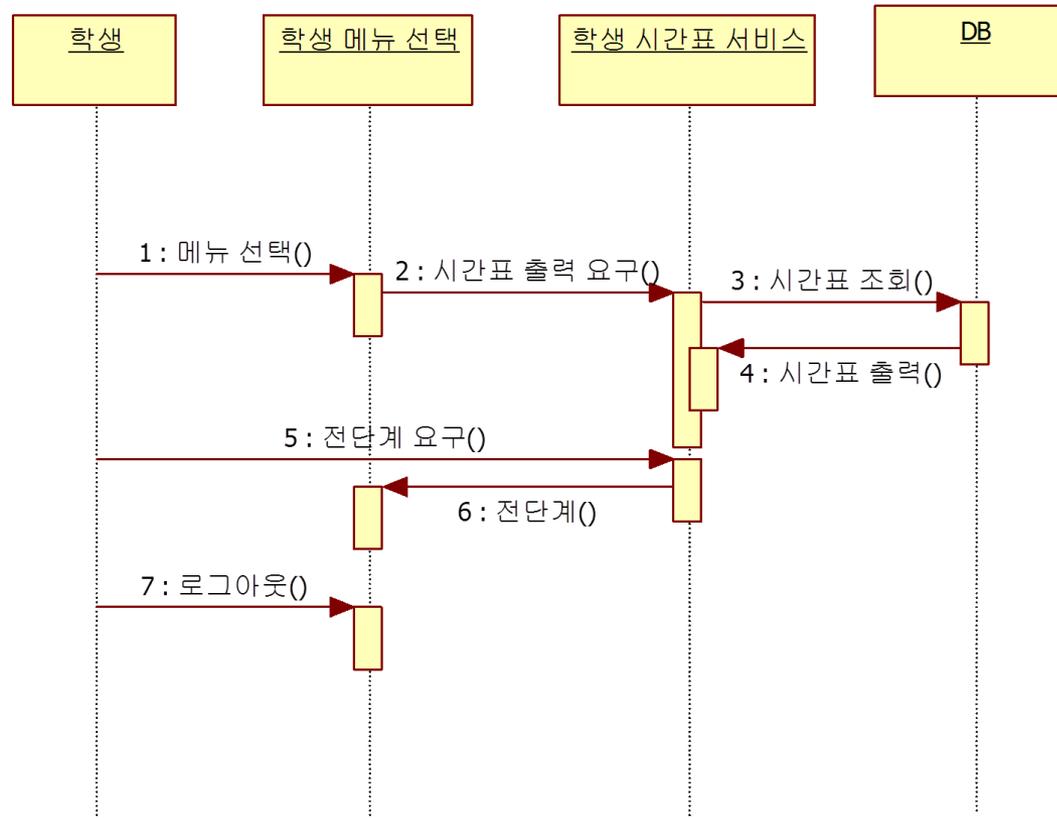
# 2144 Define Interaction Diagrams

- 강의 시간 조회



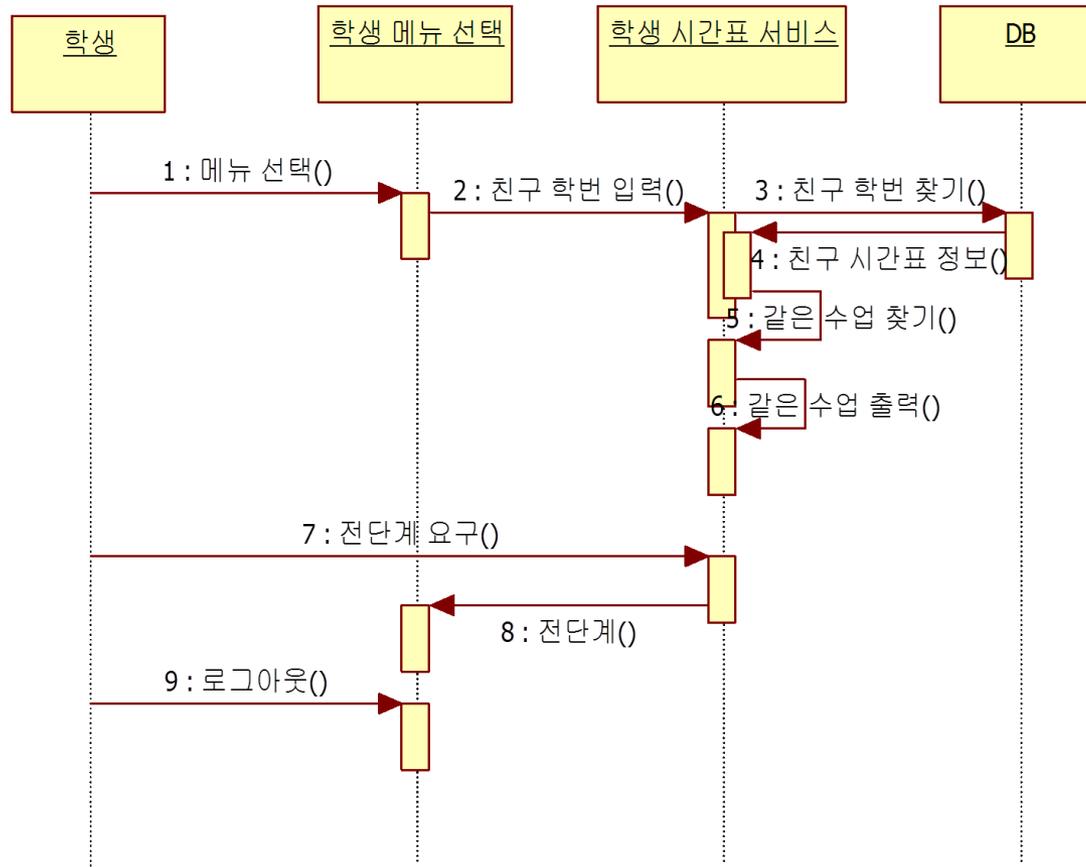
# 2144 Define Interaction Diagrams

- 시간표 출력



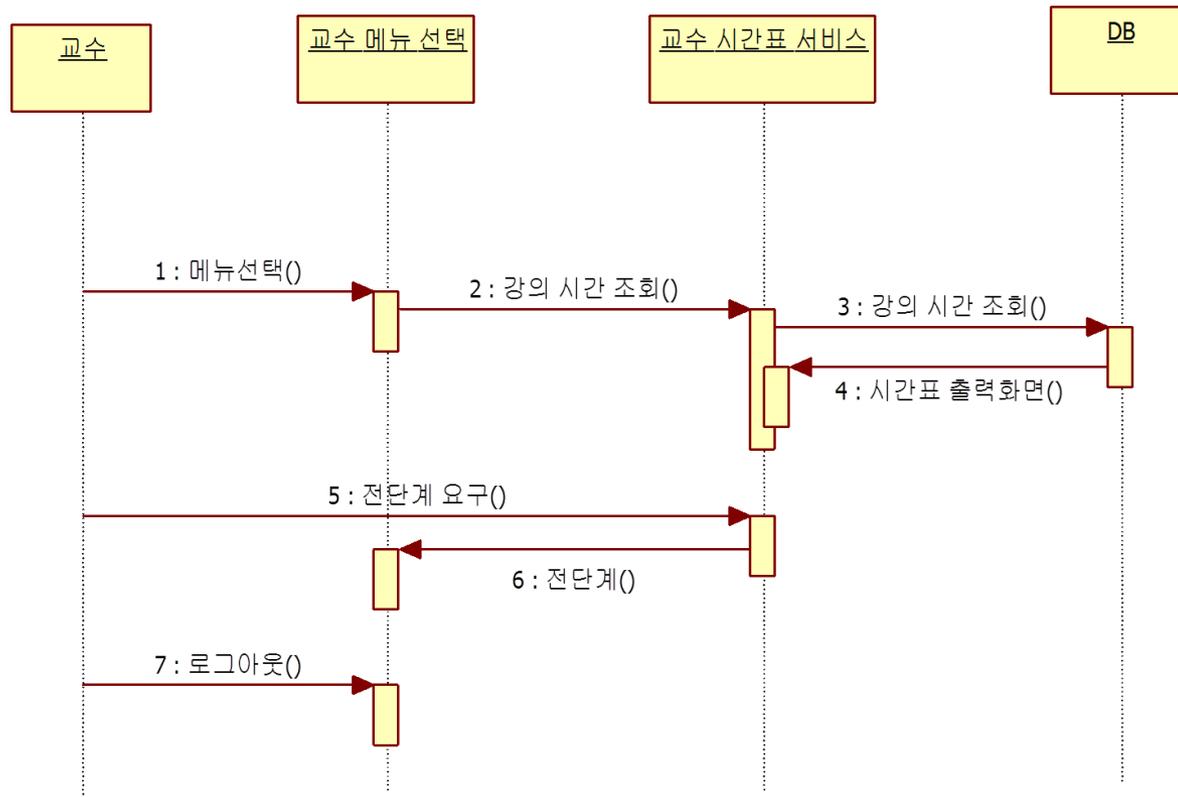
# 2144 Define Interaction Diagrams

- 친구 찾기



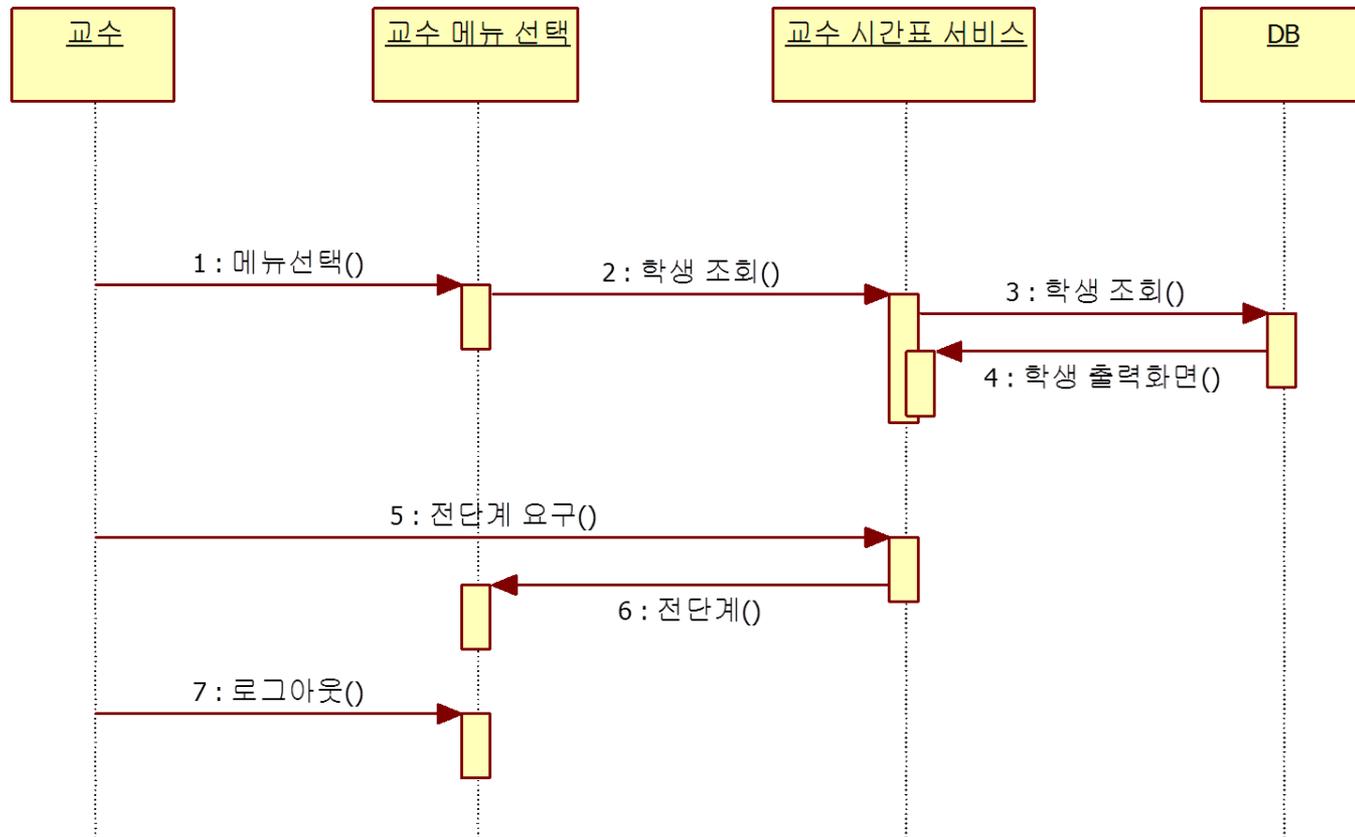
# 2144 Define Interaction Diagrams

- 강의 시간 조회



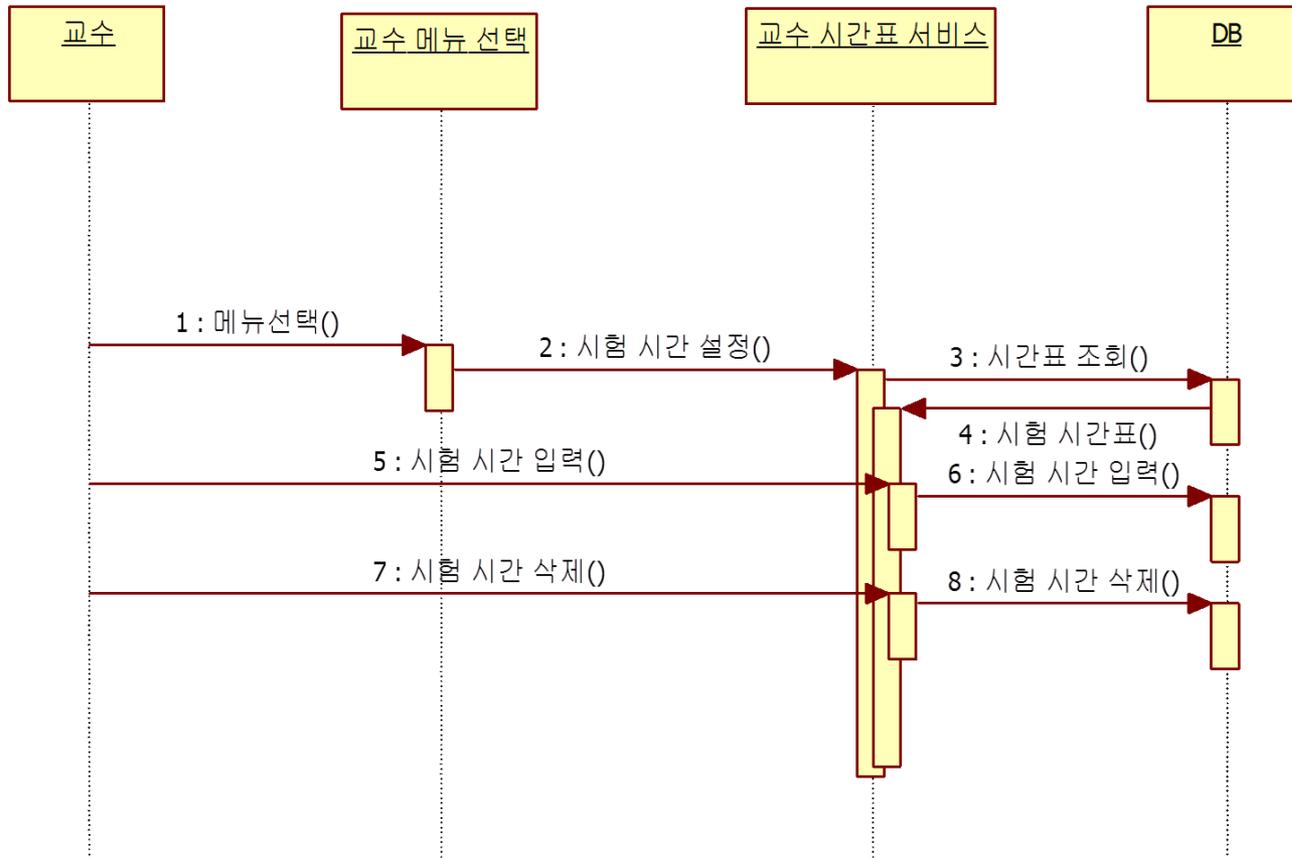
# 2144 Define Interaction Diagrams

- 학생 조회



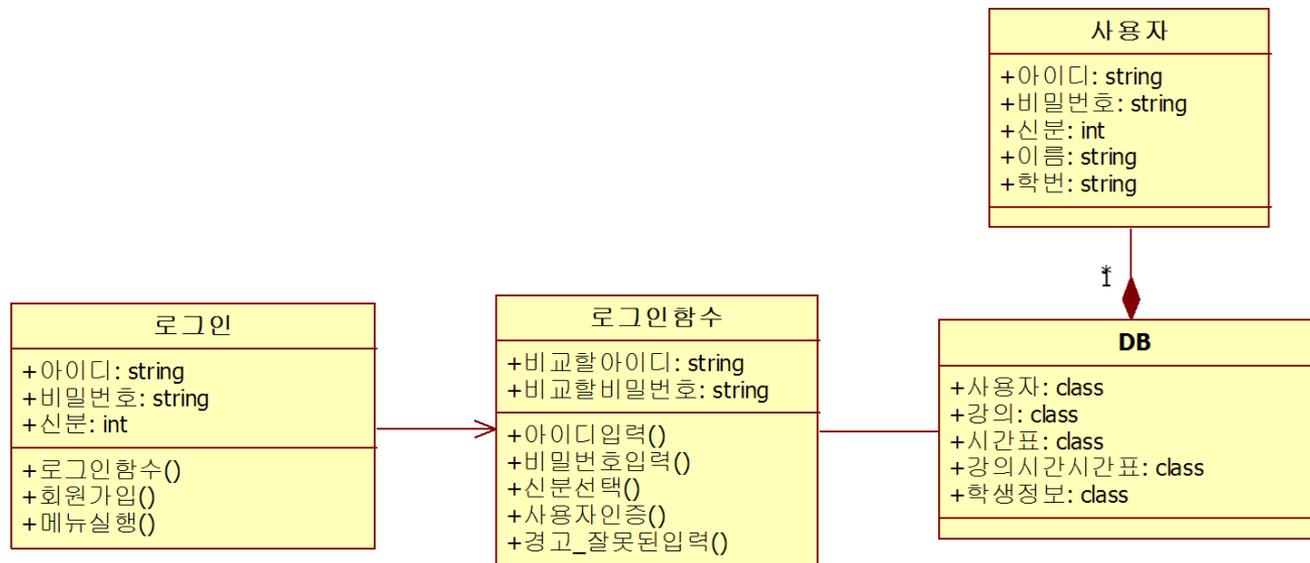
# 2144 Define Interaction Diagrams

- 시험 시간 설정



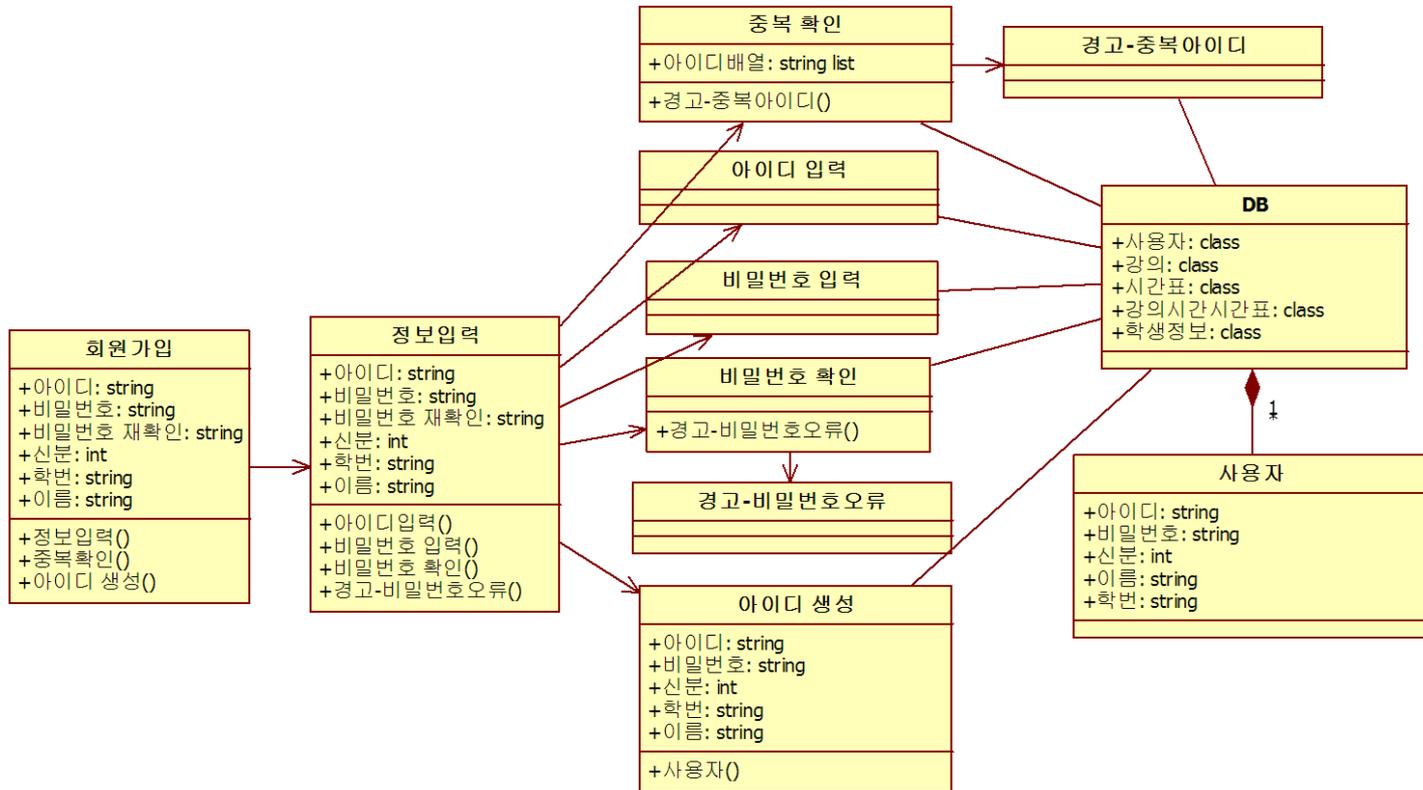
# 2145 Define Design Class Diagrams

- 로그인



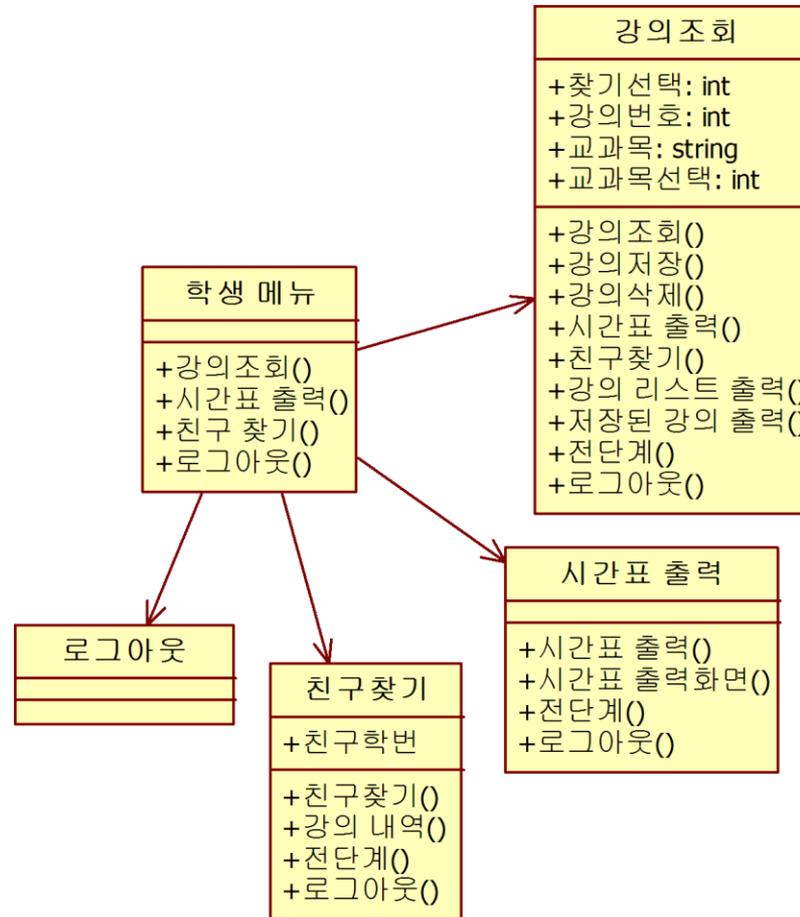
# 2145 Define Design Class Diagrams

## - 회원가입



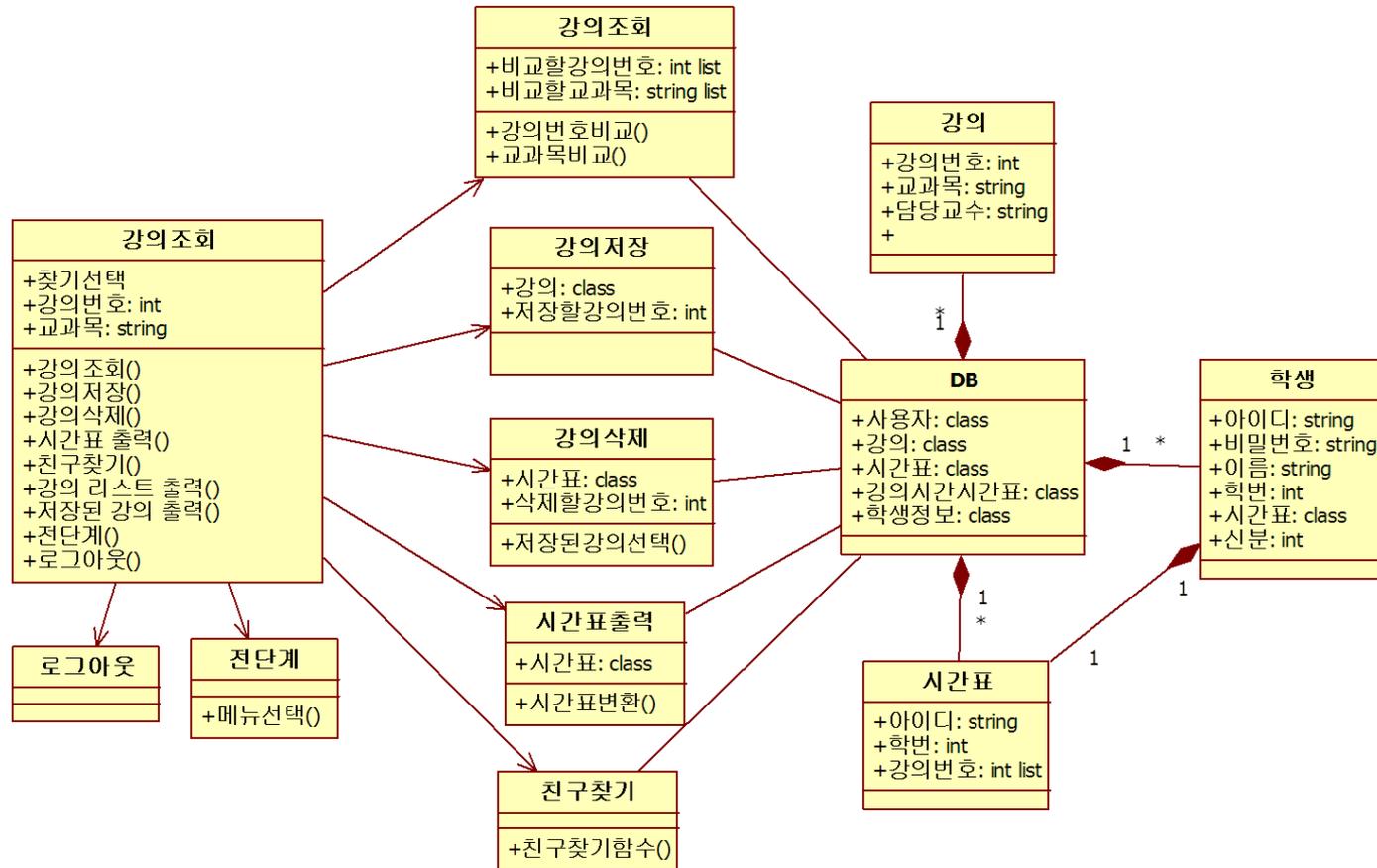
# 2145 Define Design Class Diagrams

-학생메뉴



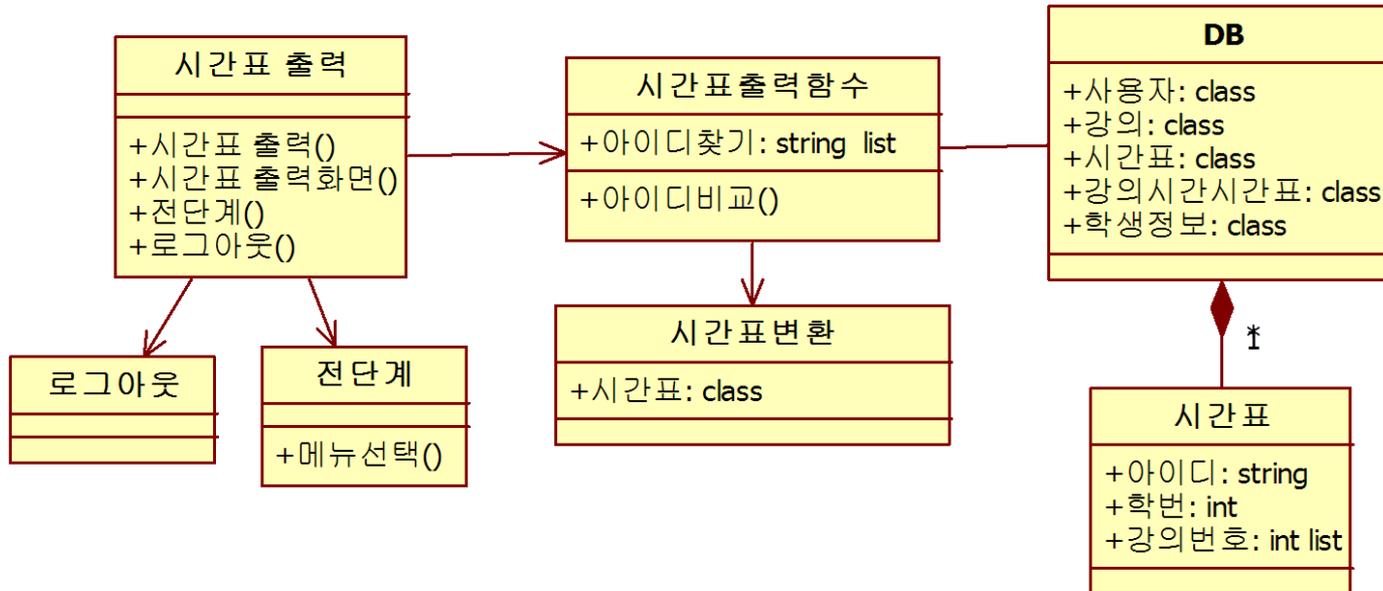
# 2145 Define Design Class Diagrams

- 강의 조회



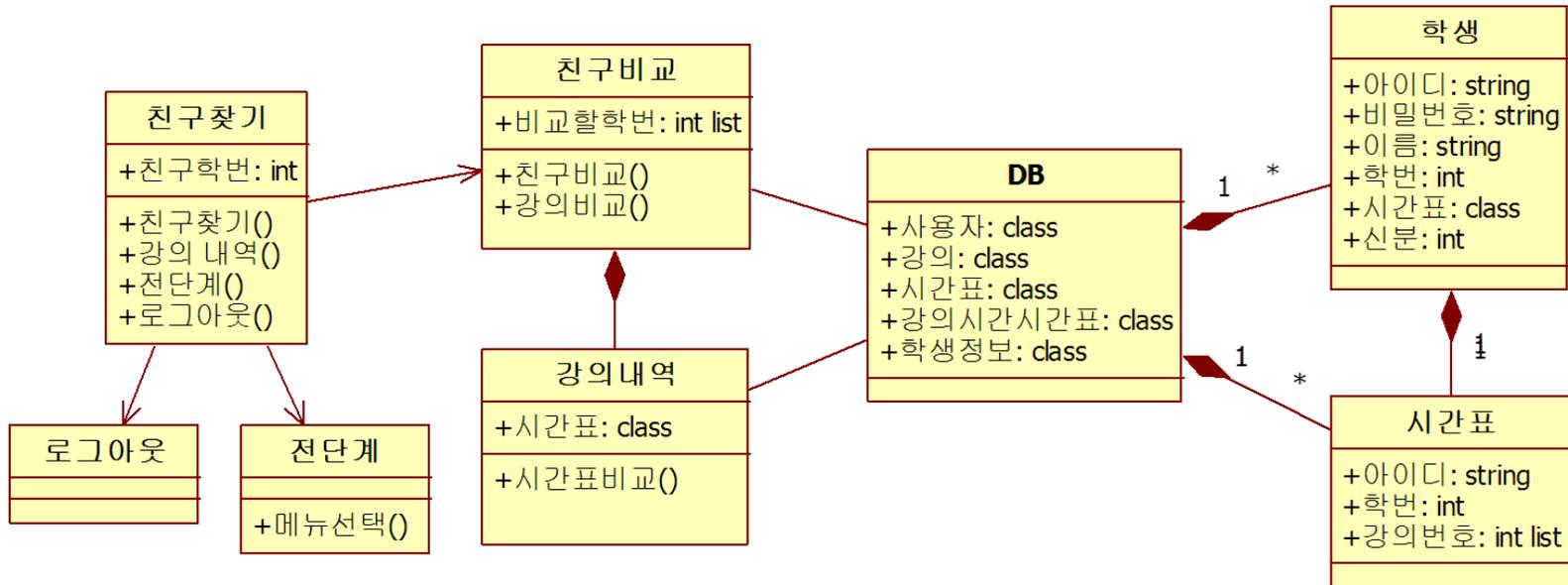
# 2145 Define Design Class Diagrams

- 시간표 출력



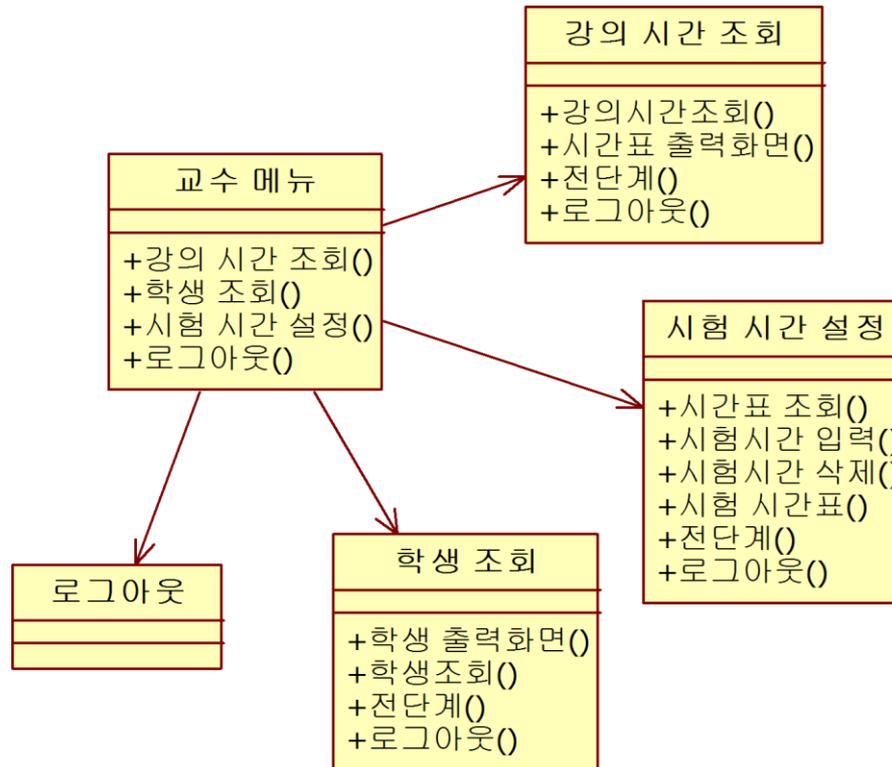
# 2145 Define Design Class Diagrams

- 친구 찾기



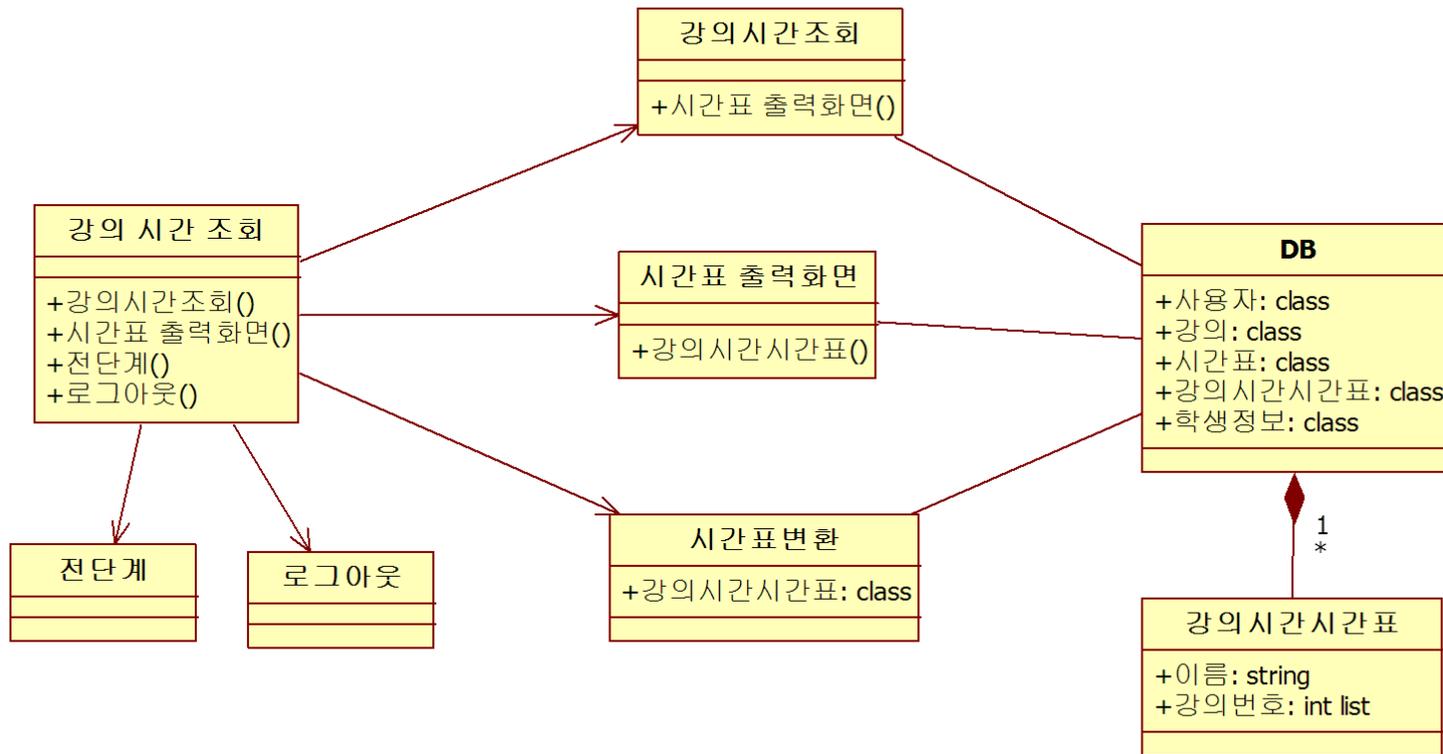
# 2145 Define Design Class Diagrams

- 교수 메뉴



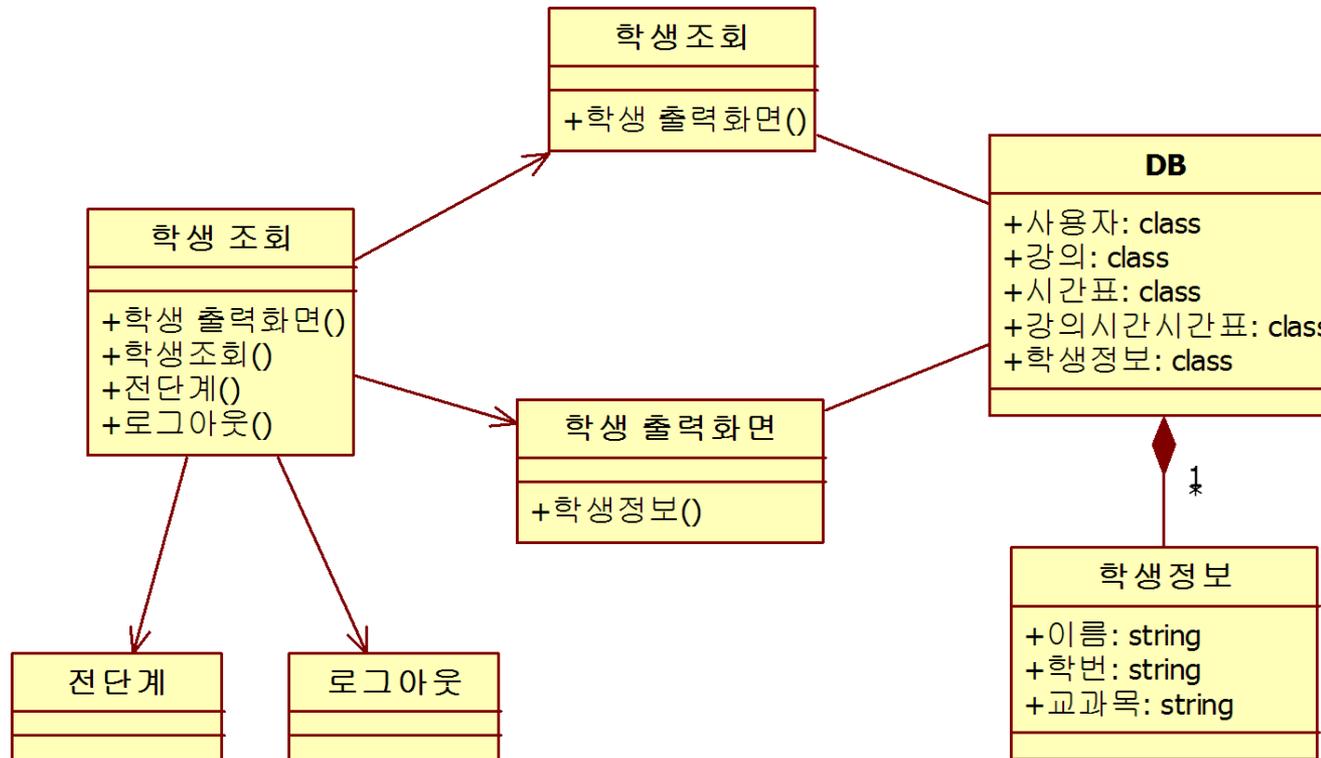
# 2145 Define Design Class Diagrams

- 강의 시간 조회



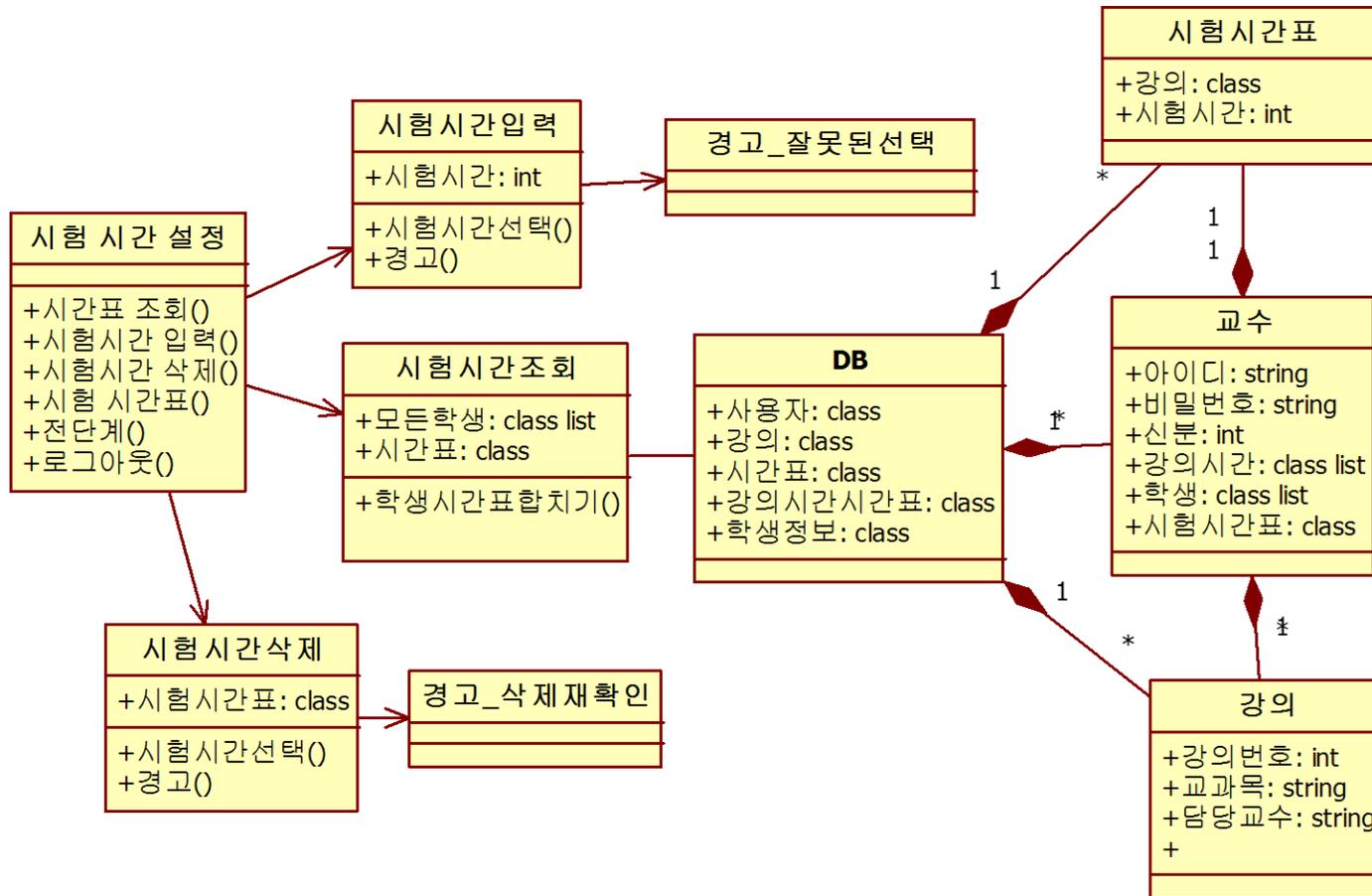
# 2145 Define Design Class Diagrams

- 학생 조회



# 2145 Define Design Class Diagrams

## - 시험 시간 설정



# 2146 Define Database Schema

---

- 사용자 테이블

아이디	비밀번호	신분	이름	학번
exam100	100wjaakswnj	2(학생)	김민하	200911372

- 강의 테이블

강의번호	교과목	담당교수	강의시간
1111	성적에대한고찰	나백점	11-13

- 시간표 테이블

아이디	학번	강의번호(list)	강의명(list)
exam100	200911372	1111	성적에대한고찰
exam100	200911372	1112	과제에대한고찰



# 2146 Define Database Schema

---

- 학생정보 테이블

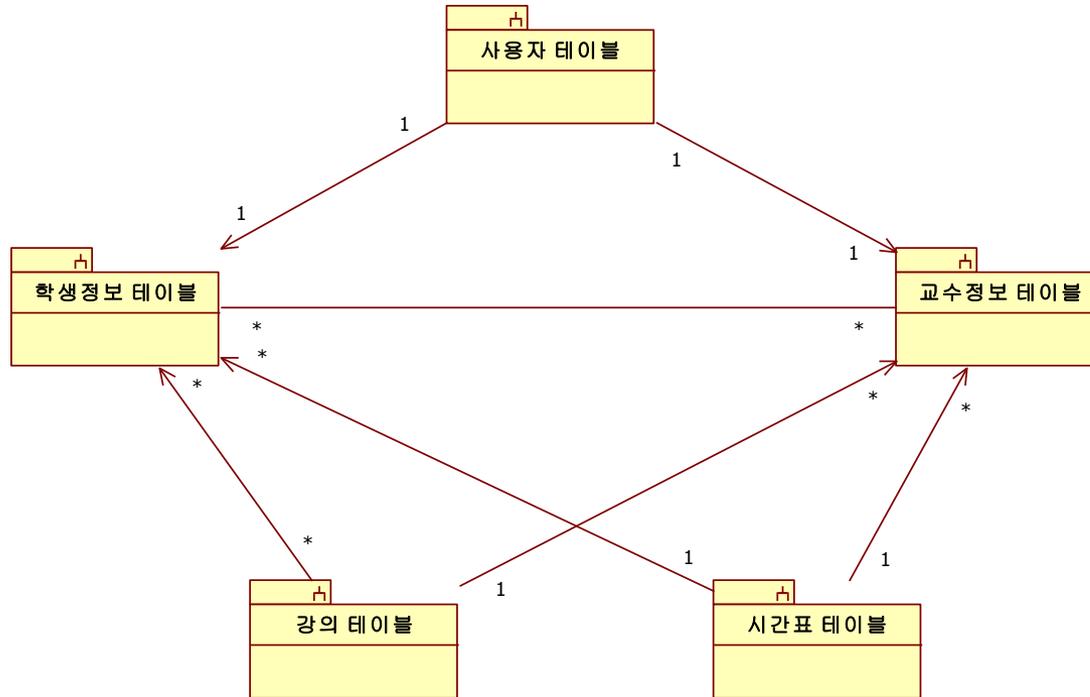
아이디	비밀번호	이름	학번	시간표(list)	신분
exam100	100wjaakswnj	김민하	200911372		2(학생)

- 교수정보 테이블

아이디	비밀번호	신분	강의시간	학생	시험시간표(list)
abcd1234	epqlf	1(교수)	11-13	김민하	



# 2146 Define Database Schema



테이블 관계

Data Base 테이블은 5가지로 구성되는데 사용자 테이블, 학생정보 테이블, 교수정보 테이블, 강의 테이블, 시간표 테이블로 구성되어 있습니다. 각각의 관계를 설명하면 사용자 테이블의 경우 학생정보 테이블과 교수정보 테이블과 각각 일대일 대응 관계로 이루어져 있으며 사용자 하나가 학생 혹은 교수의 한 명하고만 관계를 이룹니다.

학생정보 테이블과 교수정보 테이블과 다(多)대다(多) 관계를 이루고 있으며 학생 하나에 여러 교수, 교수 하나에 여러 학생을 가지고 있습니다. 강의 테이블은 학생정보 테이블에는 강의 하나에 여러 학생, 학생 하나에 여러 강의의 관계를 가지고 있어 다(多)대다(多) 관계가 형성되며, 강의 테이블과 교수정보 테이블은 강의 하나에 교수 하나, 교수하나에 여러 강의를 포함하고 있어 일대다(多) 관계를 이룹니다. 시간표 테이블의 경우에는 교수든 학생이든 시간표 하나가 교수와 학생 모두 여러 명 씩 가지기에 일대다(多) 관계를 이룹니다.