**Readme (Health History)**

**프로그램 설명**

: 매일 운동 목표를 설정하고 수행한 운동을 기록하는 운동기록관리 프로그램입니다.   
특정 날짜 또는 기간의 운동기록에 대한 통계도 제공합니다.

**프로그램 시작 방법**

1. Login 패키지 내부의 main\_startProgram 클래스 실행
2. 회원가입 후 로그인
3. 날짜 선택 후 운동기록 입력

**데이터 저장구조**

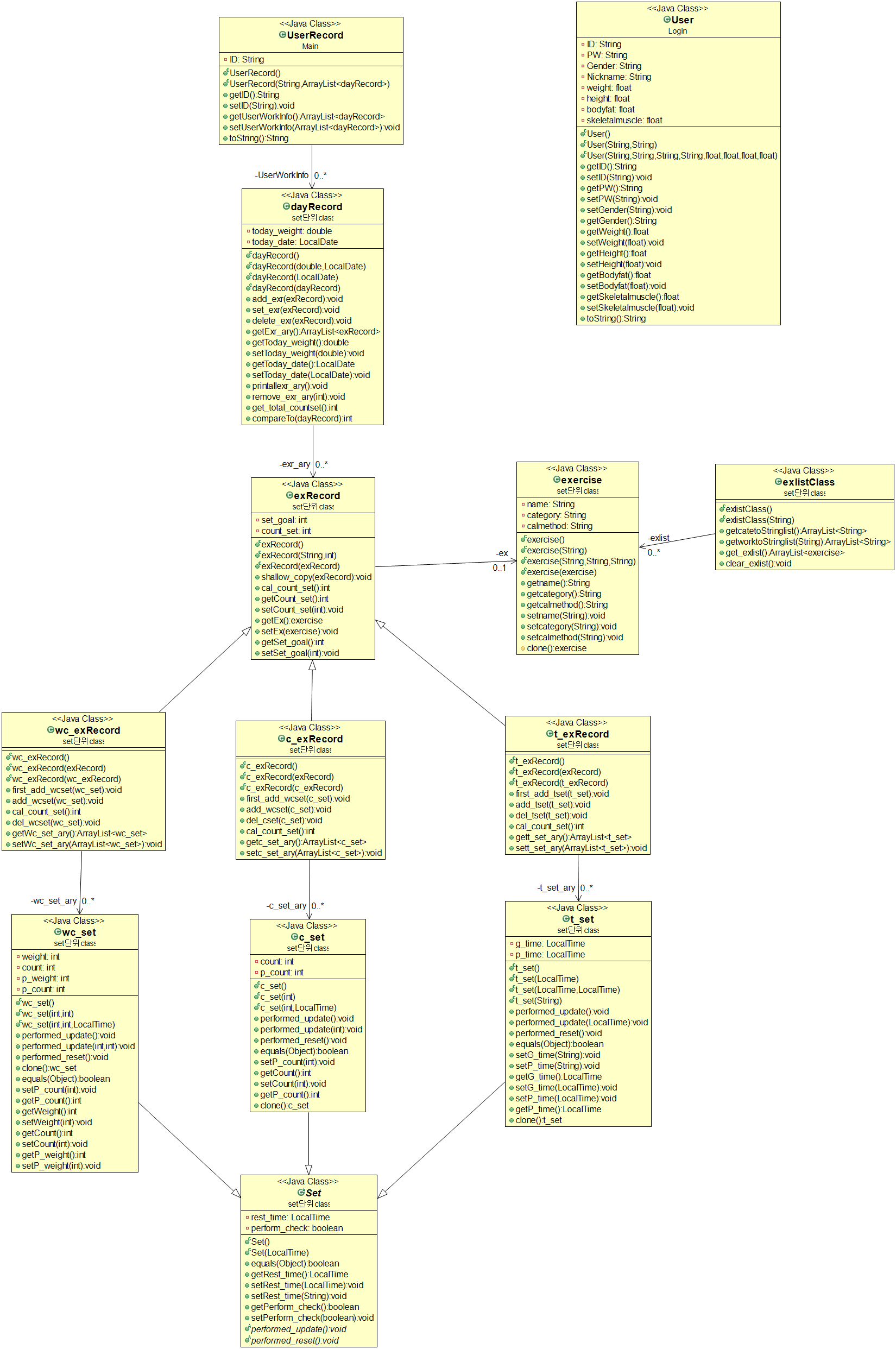
* + User : 회원 개인정보 (ID, PW, 성별, 키, 몸무게, 골근량, 체지방량)
  + UserRecord : 회원 ID, 회원 기록한 모든 운동기록 (Arraylist<dayRecord>)
  + dayRecord : 날짜, 오늘의 몸무게, 그 날짜의 모든 운동기록 (Arraylist<exRecord>)
  + exRecord : 운동정보(exercise), 목표 세트수 (set\_goal), 수행 세트수 (count\_set)
    - 세트의 종류에 따라 3가지로 나뉨
  + set : 휴식시간, 수행완료 여부 (performed check)
    - 측정 방식에 따라 3가지로 나뉘고 각 세트는 목표 측정값과 수행 측정값을 가짐
    - wc\_set (무게 \* 횟수로 측정) ex) 벤치프레스 – 30kg \* 3회
    - c\_set (횟수로 측정) ex) 푸쉬업 – 3회
    - t\_set (시간으로 측정) ex) 매달리기 – 2분
  + exercise : 운동명, 카테고리, 측정 방식

\*세부사항은 아래의 Class Diagram 참고

**데이터 저장방식**

* User는 user 폴더에 한 명씩 binary형태로 저장
* UserRecord는 userworkinfo 폴더에 한 명씩 binary형태로 저장
* exercise는 한 개당 한 줄씩 All\_WORKOUT 파일에 text형태로 저장

**데이터 저장구조에 대한 Class Diagram**



**프로그램 구현**

1. 회원별 데이터 관리 기능 – 로그인 페이지

2. 메인 페이지 (달력 인터페이스)

3. 운동 목표 설정 및 기록 기능

1. 운동 기록 통계 기능
2. **회원별 데이터 관리 기능**

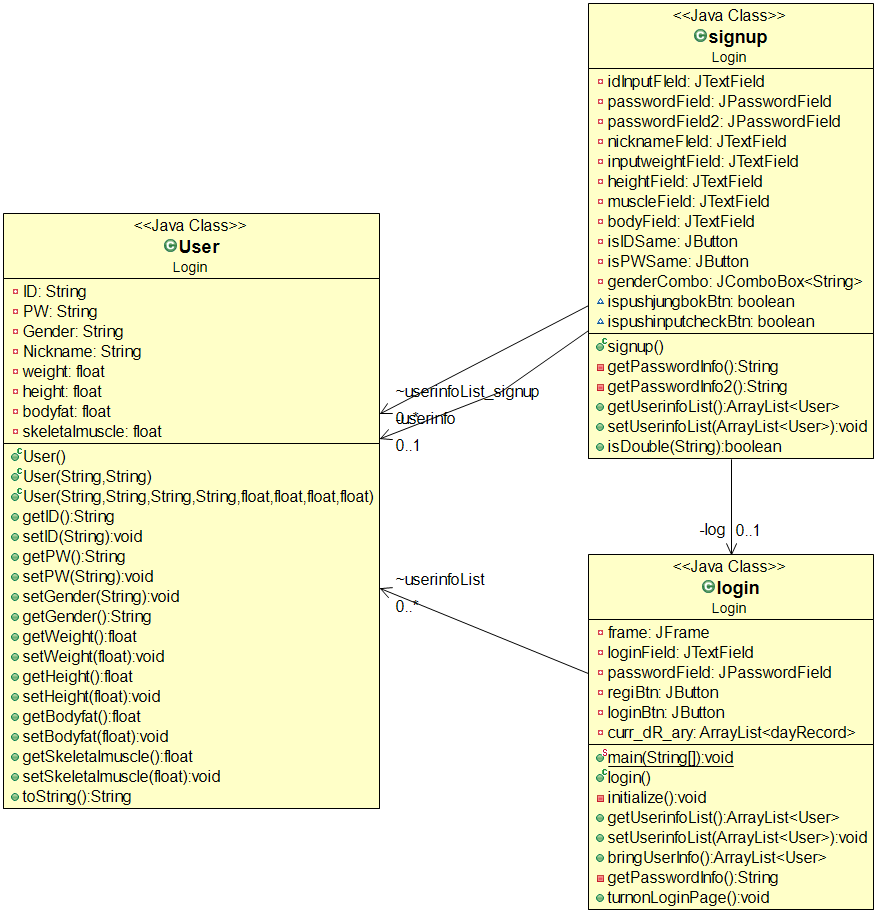
login : 로그인 페이지 (JFrame)

* 로그인 버튼
  + 입력된 ID와 PW를 user 폴더에 있는 모든 유저 정보와 비교
  + 일치하는 값이 있으면 로그인 성공, 없으면 로그인 실패
* 회원가입 버튼
  + 버튼 클릭 시 signup 페이지를 연다

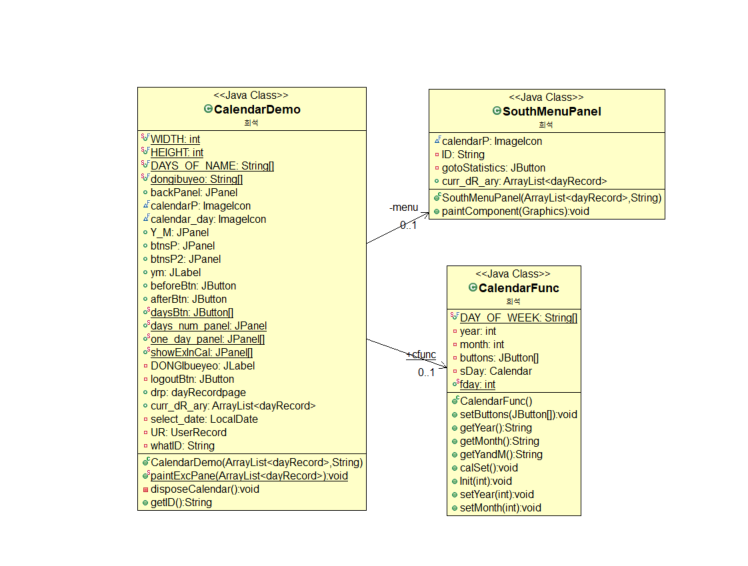
Signup : 회원가입 페이지 (JFrame)

* User 클래스 형태로 저장함
* 중복확인 버튼 (isIDSame)
  + 이때까지 입력된 모든 회원의 ID와 중복되는지 확인
* 입력확인 버튼 (isPWSame)
  + 먼저 입력한 PW와 같은지 확인
* 회원가입 버튼 (registerBtn)
  + 각 입력 값들이 정상 입력 되었는지 확인
  + 정상 입력 되었다면 가입 처리와 동시에 유저 이름으로 binary파일을 만들어  
    user폴더에 저장

**회원별 데이터 관리 기능 Class Diagram**



**메인 페이지 Class Diagram**

****

1. **메인 페이지**

CalendarDemo : 달력 인터페이스 구현

* userworkinfo 폴더에 미리 기록된 정보가 있는지 확인 후 있다면 불러옴
* 달력 조작
  + Before 버튼 (beforeBtn) : 이전 월로 이동
  + After 버튼 (afteBtn) : 다음 월로 이동
* 동기부여 글귀
  + 글귀 5개 중 랜덤하게 생성
* 날짜 panel (one\_day\_panel)
  + 날짜 버튼(daysBtn)과 운동 정보 패널(showExInCal)로 나뉨
  + 날짜 버튼
    - 날짜 한 개를 선택했을 때 dayRecordpage가 열림
    - 해당 날짜에 운동기록이 있을 경우 날짜 버튼이 노란색으로 변경됨
  + 운동 정보 패널
    - paintExcPane 메소드로 구현
    - 운동기록이 있는 날짜에는 카테고리별 세트 수 top3를 보여줌
    - 수행 세트 수 / 목표 세트 수 형식으로 세트 수 보여줌
* SouthMenuPanel : 달력 인터페이스 하단 menu panel
  + 통계 버튼
    - 버튼 클릭 시 SelectStatisticsWayDemo 페이지가 열림

CalendatFunc : 캘린더 버튼의 숫자를 설정해주는 기능 구현

* calSet() 메소드를 이용해 버튼에 실제 달력과 동일한 위치에 날짜를 할당함

1. **운동목표 설정 및 기록 기능** (아래의 Class Diagram 참고)

DayRecordpage : 하루 운동기록을 보여주는 page

* 오늘의 날짜가 상단에 표시됨
* 해당 날짜에 기록된 exRecord들을 보여줌. 여러 개의 expanel로 구성
* 운동 추가 버튼 클릭 시 addexRecordpage가 열림
* addexRecordpage에서 받아온 정보로 expanel을 만듦
* 오늘의 몸무게를 입력할 수 있음

addexRecordpage : 운동1개를 추가하는 page. (목표만 설정함)

* 카테고리, 운동명을 선택하고 목표 세트수를 입력함

expanel : 운동 1개에 대한 조작을 하는 Jpanel

* 운동명과 (현재 수행 세트수/목표 세트수)를 나타냄
* 수정 버튼 클릭 시 해당 운동에 대한 exRecordpage가 열림
* 삭제 버튼 클릭 시 해당 운동을 삭제함

exRecordpage : 운동 1개에 대한 기록을 보여주는 page

* 해당 운동의 측정 방식에 따라 다르게 보여짐
* 해당 운동에 기록된 set들을 보여줌. 여러 개의 ?\_setpanel로 구성
* 운동 추가 버튼 클릭 시 운동 측정 방식에 따라 다른 add\_?\_setpage가 열림
* add\_?\_setpage로부터 받아온 정보로 ?\_setpanel을 만듦

\* 참고 : ?에는 wc (무게\*횟수), c (횟수), t(시간) 가 들어간다.

단, 하나의 운동에 대해서는 한가지 방식으로만 저장되고 나타내짐.

Ex) 푸쉬업은 횟수로 측정함으로 c\_set으로만 저장됨. wc\_set이나 t\_set이 들어가지 않음.

또한 add\_csetpage가 열리고 c\_setpanel로만 구성됨

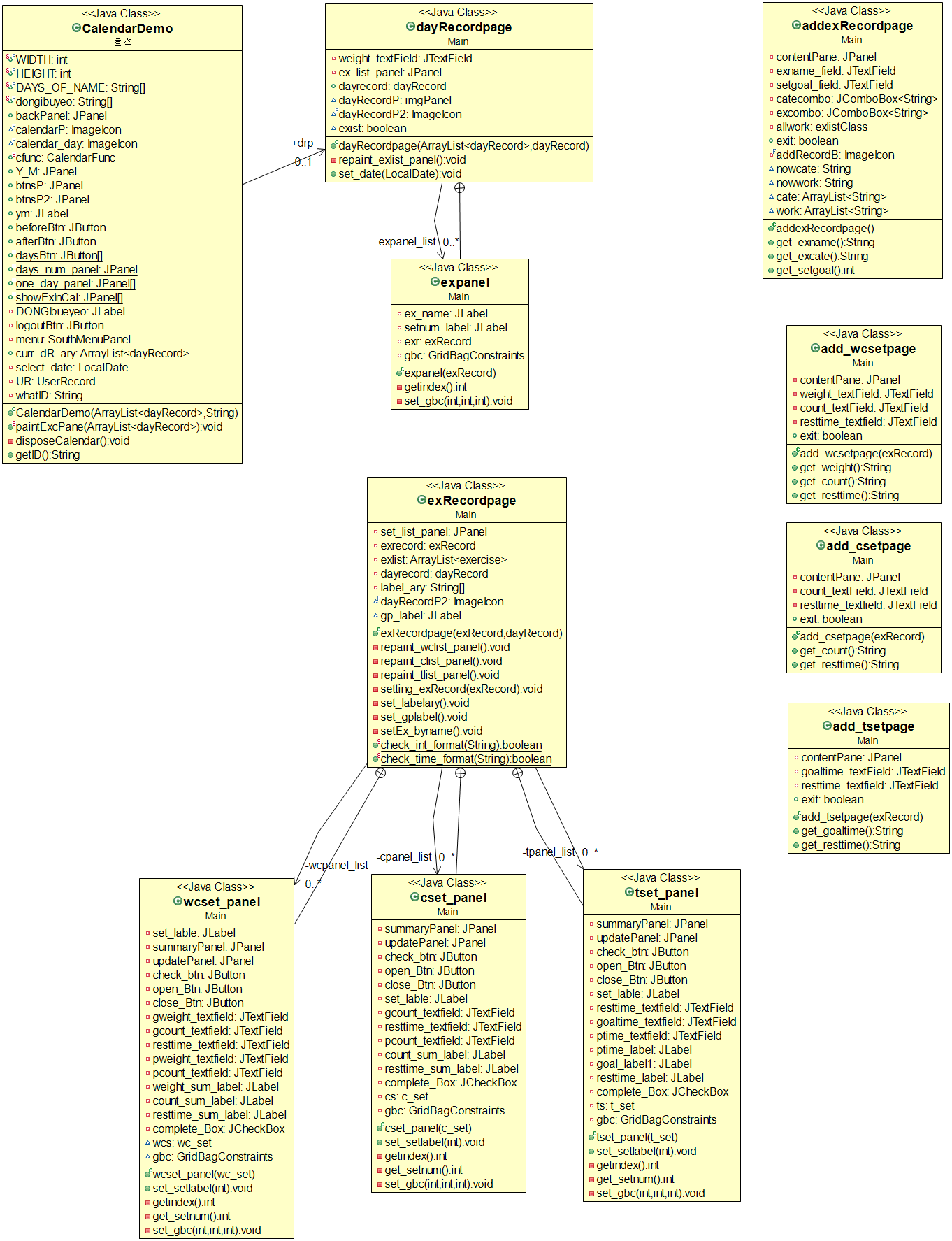
add\_?\_setpage : 세트 1개를 추가하는 page (목표만 설정함)

* 운동 측정 방식에 따라 다름
* add\_wcsetpage : 무게, 횟수, 휴식시간 입력
* add\_csetpage : 횟수, 휴식시간 입력
* add\_tsetpage : 목표시간, 휴식시간 입력

?\_setpanel : 세트 1개에 대한 기록 및 조작을 하는 Jpanel

* 내부적으로 summary panel과 update panel로 구성
* summary panel
  + 체크박스 클릭 시 운동을 목표 세트와 같게 설정하고 체크표시
  + 체크표시 되어있는 체크박스 클릭 시 수행 측정값 초기화 및 체크해제
  + 삭제 버튼을 눌러 해당 세트 삭제가능
  + 열기 버튼 클릭 시 update panel이 열리고 닫기 버튼으로 바뀜
* update panel
  + Load 버튼을 눌러 목표 측정값을 그대로 수행 측정값으로 가져옴
  + Reset 버튼을 눌러 수행 측정값을 초기화
  + 목표 측정값, 수행 측정값을 직접 입력하고 저장 버튼을 눌러 수정 가능
    - wc\_setpanel과 c\_setpanel에서는 수행 횟수 > 목표 횟수일 때 체크표시
    - t\_setpanel에서는 수행 시간 > 목표시간 일 때 체크표시

**운동 목표 설정 및 기록 Class Diagram**



1. **통계 기능**

SelectStatisticsWayDemo : 통계 종류 선택 page

* 하루 통계 체크박스 선택 (dayChkBox)
  + 날짜를 입력하고 통계 버튼을 누르면 DaystatisticsDemo 페이지 열림
* 기간 통계 체크박스 선택 (periodChkBox)
  + 시작일, 종료일을 선택하고 통계 버튼을 누르면 PeriodStatisticsDemo 페이지 열림

DaystatisticsDemo : 하루 통계 page

* DayStatisticsFunc을 통해 해당 날짜의 dayRecord 내부 정보를 가져옴
* 정보 출력
  + 총 세트 수 : 모든 운동의 세트 수 합계
  + 총 무게 : 무게 \* 횟수로 측정한 운동에 대해서만 계산

PeriodStatisticsDemo : 기간 통계 page

* PeriodStatisticsFunc을 통해 해당 기간의 dayRecord 정보를 Hashmap의 형태로 가져옴
* 정보 출력
  + 총 세트 수 : 모든 운동의 세트 수 합계
  + 총 무게 : 무게 \* 횟수로 측정한 운동에 대해서만 계산
  + 카테고리별 운동 수, 세트 수 합계
  + 세트 비중 : 해당 카테고리의 세트 수 / 전체 세트수
  + 세트 성공율 : 해당 카테고리의 수행 세트 수 / 목표 세트 수
* 가져온 정보를 바탕으로 xchart 라이브러리를 사용해 차트 구현
  + 총 세트 수 LineChart :기간 내 운동기록이 있는 날짜에 대해서만 나타냄
  + 몸무게 변동 LineChart : 무게를 입력하지 않은 날짜는 가져오지 않음
  + 카테고리별 세트 비율 PieChart

**통계 기능 Class Diagram**

