Traveler Digital Watch
OOPT Stage 2050

소프트웨어 모델링 및 실습
[19. 05.27]

201411295 이상훈
201711394 민하은
201711395 박성준
201711423 정종화

Index

2042 . Define Reports, UI and Storyboards

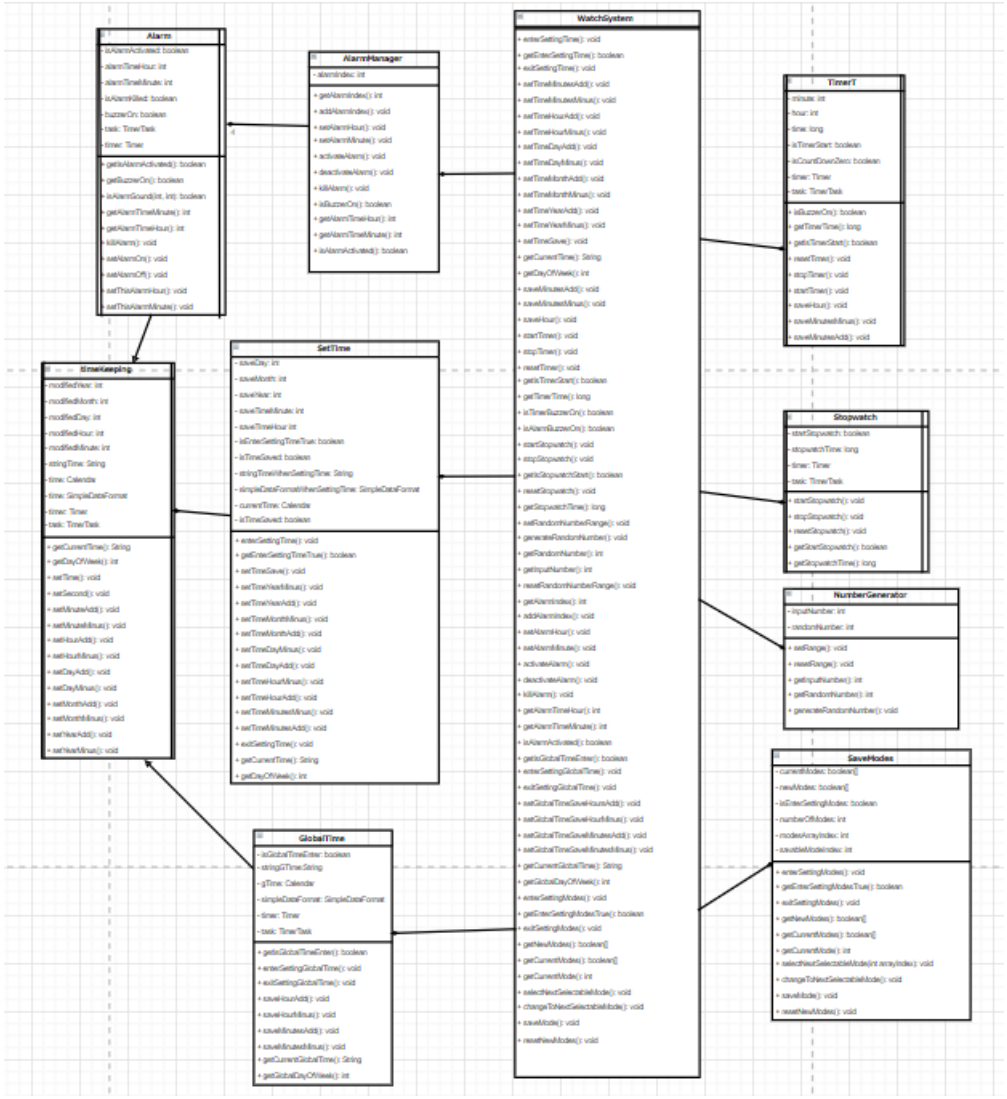
2045. Define Design Class Diagram

2052 . Implement Windows

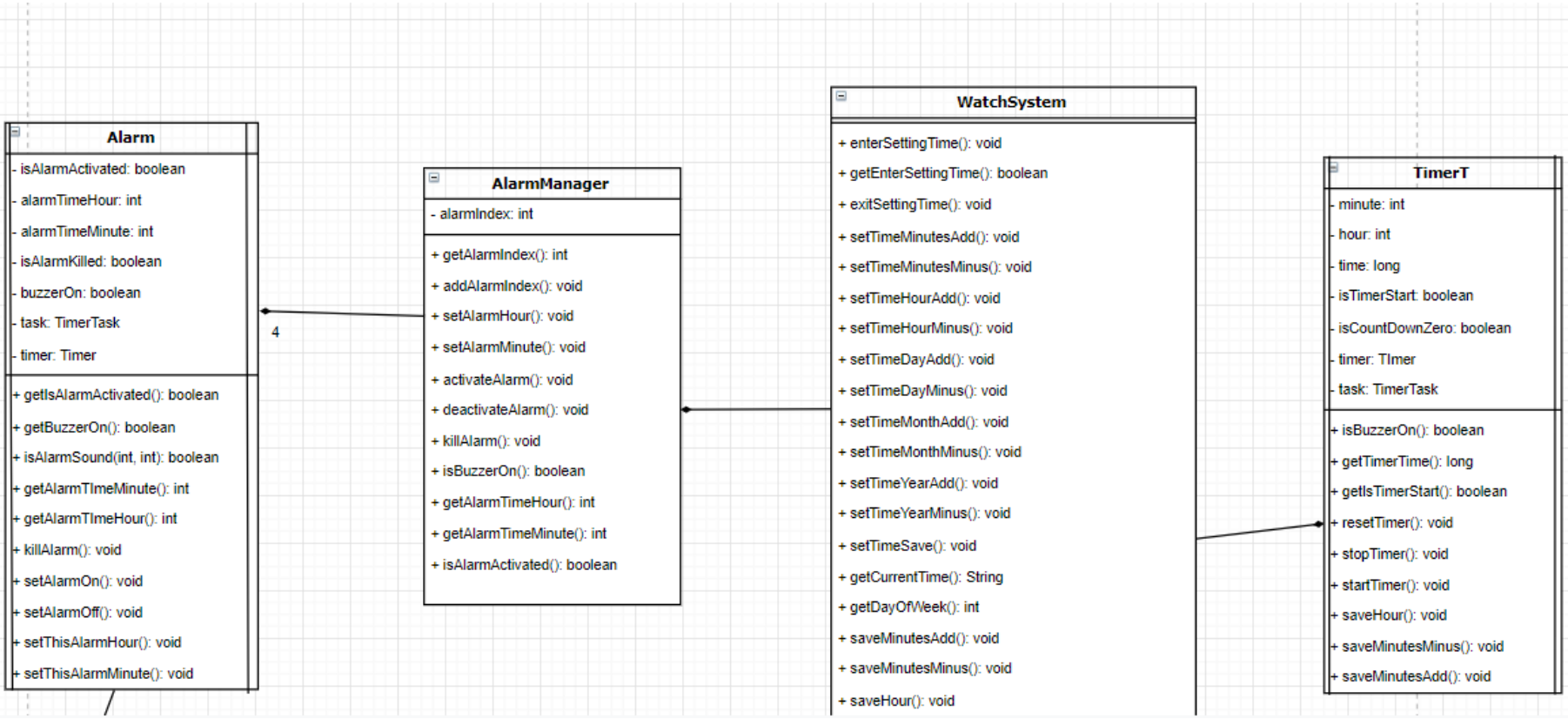
2055. Write Unit Test Code

Define Reports, UI and Storyboards

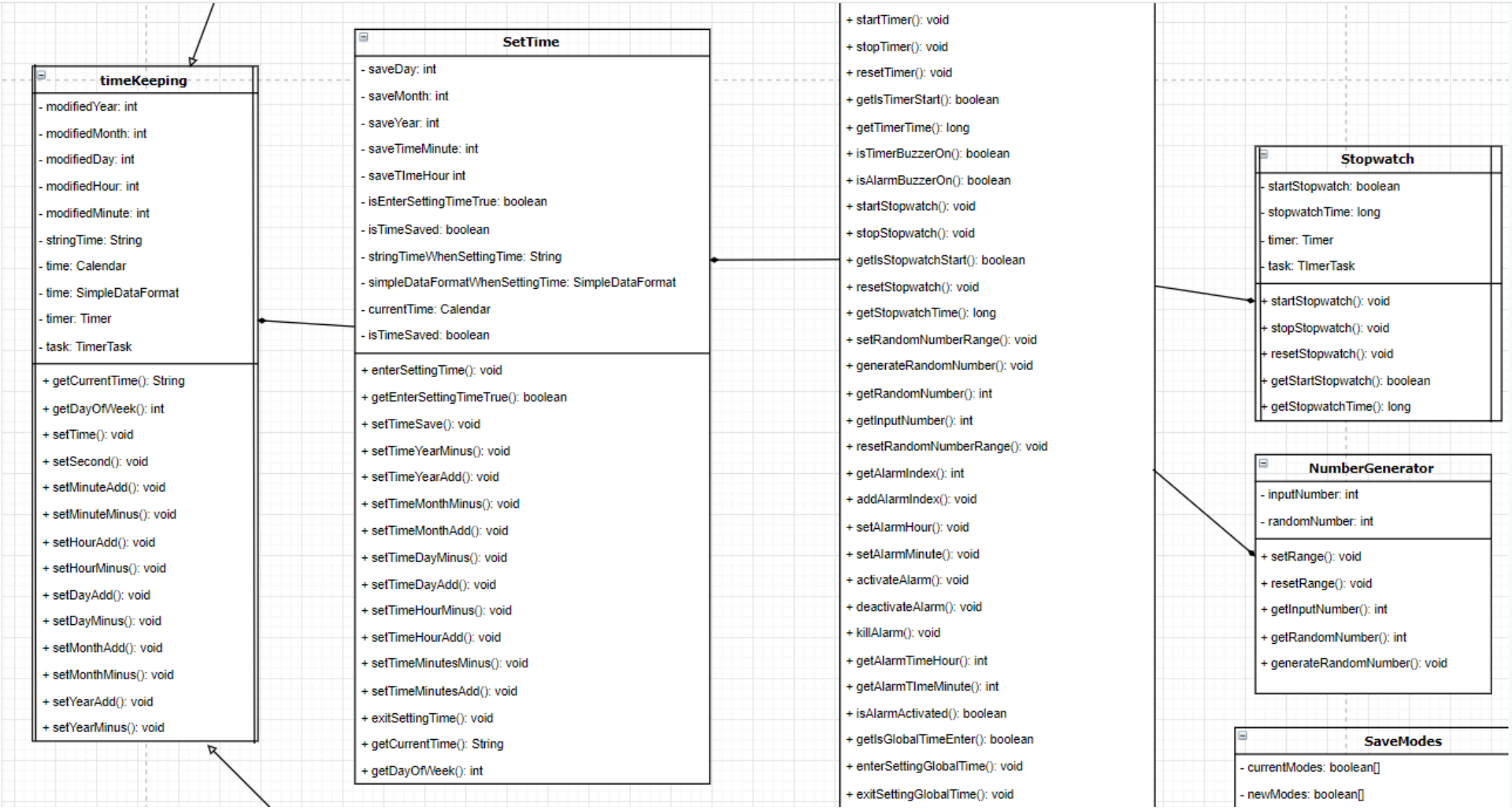
Define Design Class Diagram



Define Design Class Diagram

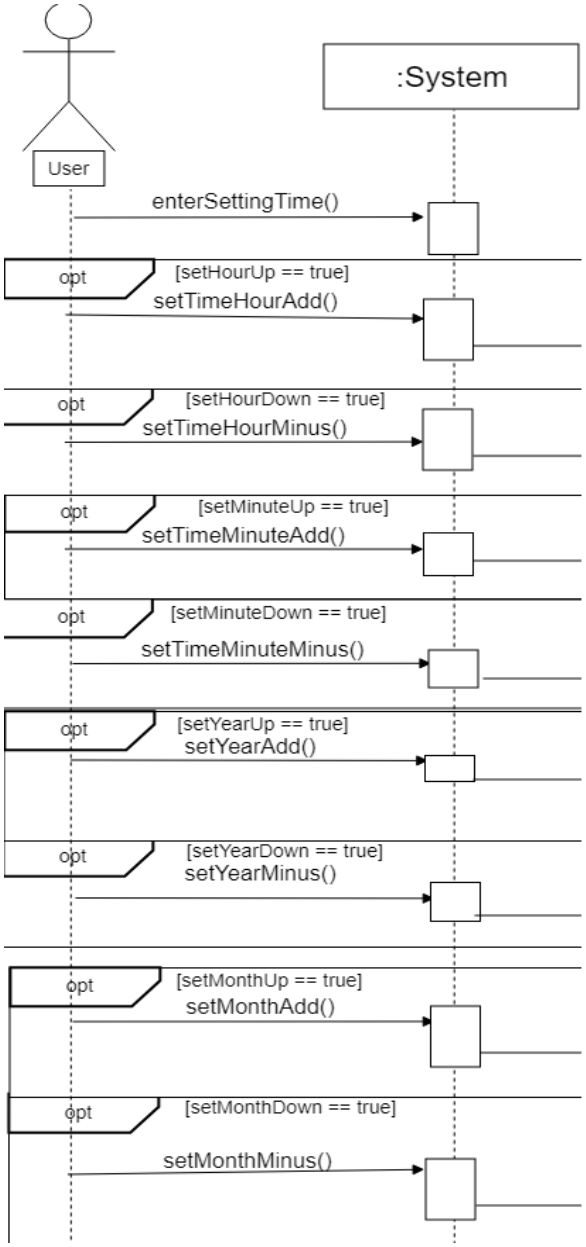


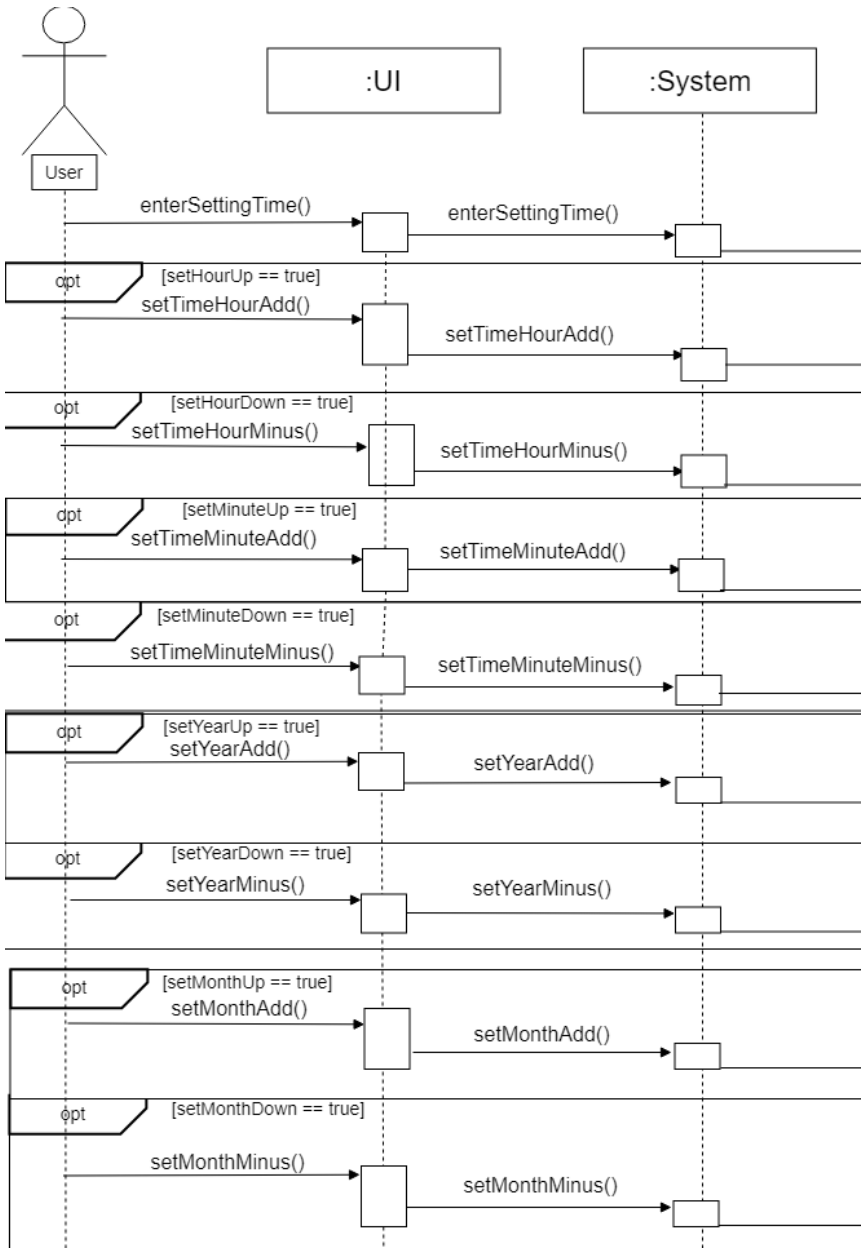
Define Design Class Diagram



Define Design Class Diagram

Implement Windows





Implement Windows

Name	SetTime
Responsibilities	시계의 현재 시간을 조정한다.
Type	UI
Cross References	R.4.3, R.6.0
Note	입력된 버튼에 따라 연, 월, 일, 시, 분을 1씩 조정할 수 있다.
Pre-Conditions	현재 시간을 보는 상태
Post-Conditions	조정된 시간을 사용자에게 보여준다.

Name	SetTimer
Responsibilities	타이머의 시간을 조정한다.
Type	UI
Cross References	R.2.1
Note	입력된 버튼에 따라 타이머 시간 1분 혹은 1시간씩 조정한다.
Pre-Conditions	타이머 모드 상태
Post-Conditions	타이머에 조정된 시간이 저장되고, 사용자에게 보여준다.

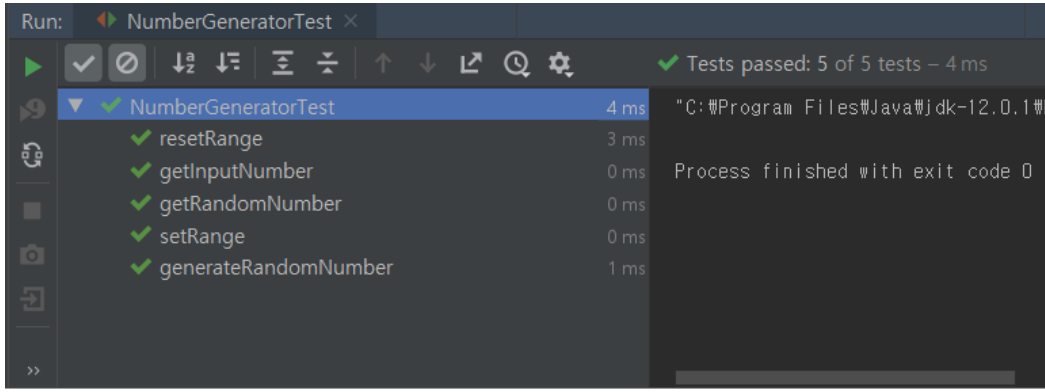
Implement Windows

Name	SetTime
Responsibilities	시계의 현재 시간을 조정한다.
Type	UI
Cross References	R.4.3, R.6.0
Note	입력된 버튼에 따라 연, 월, 일, 시, 분을 1씩 조정할 수 있다.
Pre-Conditions	현재 시간을 보는 상태
Post-Conditions	조정된 시간을 사용자에게 보여준다.

Name	SetTimer
Responsibilities	타이머의 시간을 조정한다.
Type	UI
Cross References	R.2.1
Note	입력된 버튼에 따라 타이머 시간 1분 혹은 1시간씩 조정한다.
Pre-Conditions	타이머 모드 상태
Post-Conditions	타이머에 조정된 시간이 저장되고, 사용자에게 보여준다.

Write Unit Test Code

```java
@Test
public void resetRange() {
    NumberGenerator ng = new NumberGenerator();
    ng.resetRange();
    assertEquals(ng.getInputNumber(),  actual: 0);
}

@Test
public void getInputNumber() {
}

@Test
public void getRandomNumber() {
}

@Test
public void generateRandomNumber() {
    NumberGenerator ng = new NumberGenerator();
    int i=0;
    while(i<11) {
        ng.setRange();
        i++;
    }
    ng.generateRandomNumber();
    testRandomNumber=ng.getRandomNumber();
    assertTrue( condition: testRandomNumber<10);
}
```



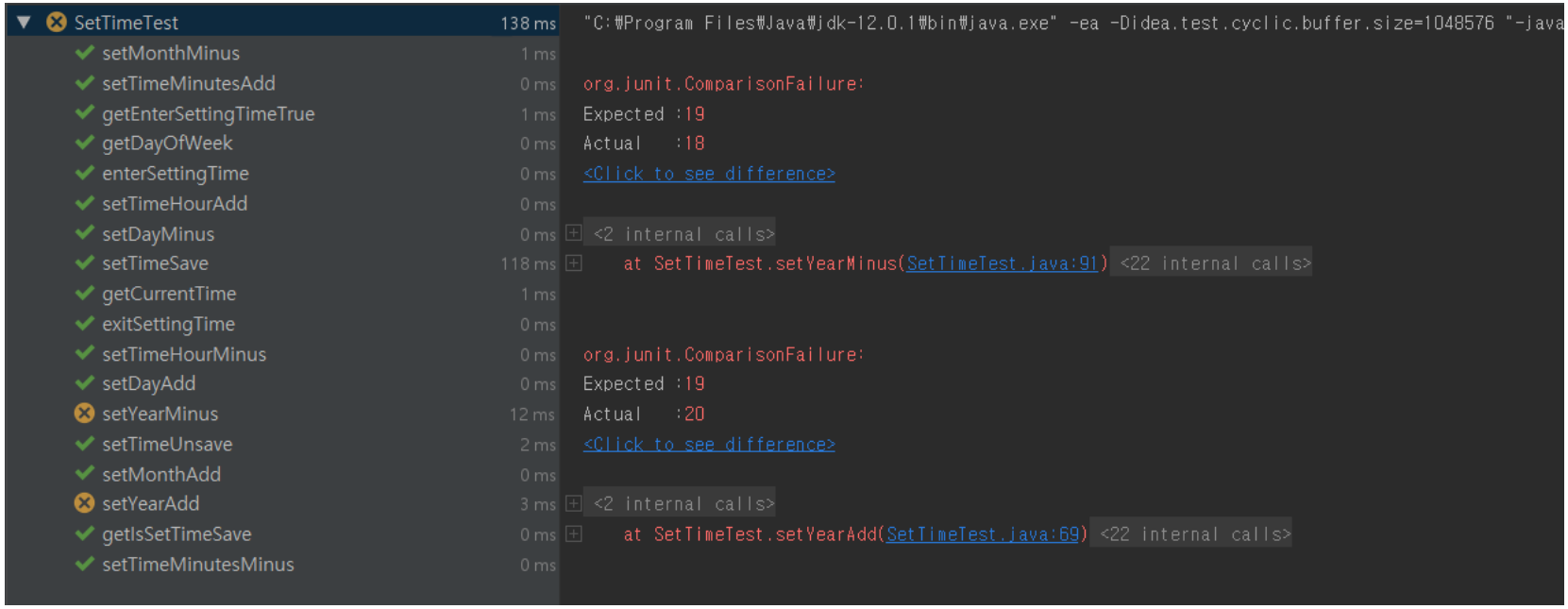
Write Unit Test Code

```java
@Test
public void setTimeSave() {
    SetTime st = new SetTime();
    String testTime;
    StringTokenizer st0, st1;
    Calendar time = Calendar.getInstance();
    SimpleDateFormat simpleDateFormat;

    st.setTimeSave();
    simpleDateFormat = new SimpleDateFormat( pattern: "yy-MM-dd HH:mm:00");
    testTime = simpleDateFormat.format(time.getTime());
    st0= new StringTokenizer(st.getCurrentTime(),  delim: " :-");
    st1 = new StringTokenizer(testTime,  delim: " :-");
    assertEquals(st0.nextToken(),st1.nextToken());
    assertEquals(st0.nextToken(),st1.nextToken());
    assertEquals(st0.nextToken(),st1.nextToken());
    assertEquals(st0.nextToken(),st1.nextToken());
    assertEquals(st0.nextToken(),st1.nextToken());
   // assertEquals(st0.nextToken(),st1.nextToken());
}
```

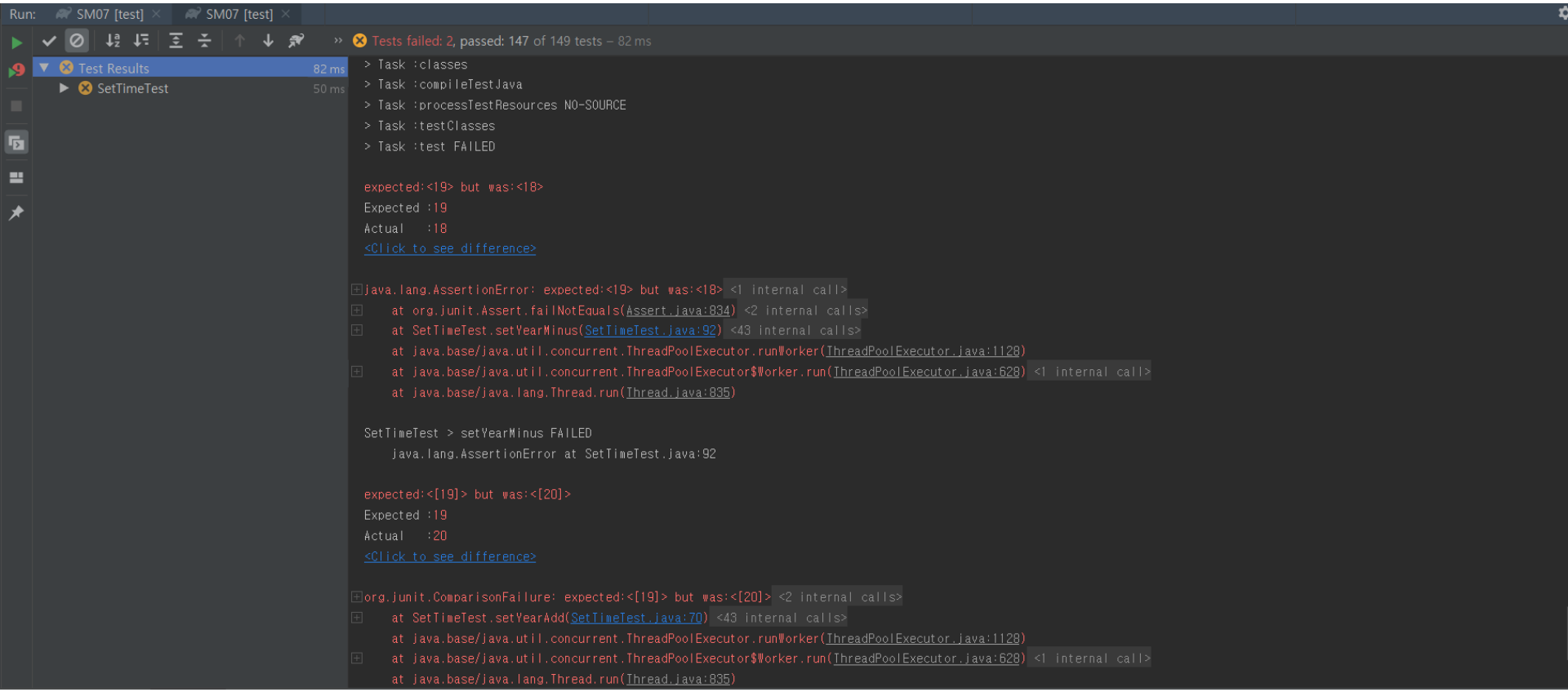
Write Unit Test Code

```java
@Test
public void setYearAdd() {
    SetTime st = new SetTime();
    String testTime;
    StringTokenizer st0, st1;
    Calendar time = Calendar.getInstance();
    SimpleDateFormat simpleDateFormat;

    st.setYearAdd();
    time.add(Calendar.YEAR, amount: 1);
    simpleDateFormat = new SimpleDateFormat( pattern: "yy-MM-dd HH:mm:ss");
    testTime = simpleDateFormat.format(time.getTime());
    st0= new StringTokenizer(st.getCurrentTime(), delim: " :-");
    st1 = new StringTokenizer(testTime, delim: " :-");
    assertEquals(st0.nextToken(),st1.nextToken());
    assertEquals(st0.nextToken(),st1.nextToken());
    assertEquals(st0.nextToken(),st1.nextToken());
    assertEquals(st0.nextToken(),st1.nextToken());
    assertEquals(st0.nextToken(),st1.nextToken());
    //assertEquals(st0.nextToken(),st1.nextToken());
}
```

Write Unit Test Code



Write Unit Test Code



감사합니다

Q & A