

제목	2개의 아날로그 시계 그리기	과제번호	2
학과	독일어과	학번	201602173
이름	윤영택	제출일	20200504

1. 문제 정의

TwoAppleClock Project의 프로그램을 참고하여 2개의 시계를 동시에 보여주며 제시된 3가지 조건에 맞는 프로그램을 FXML을 이용해 작성한다.

2-1. 문제 해결 방법

주어진 문제를 작게 나누면 3가지 문제로 나눌 수 있다.

1) 시간을 어떻게 다르게 설정할 것인가?

AppleClockController클래스의 initClock메소드 내부의 이벤트 핸들러에 의해 시간이 설정된다. 왼쪽 시계가 서울의 시간을 나타내므로 기준이 되기 때문에, 오른쪽 시계의 인스턴스 변수(rightClock.hour)에 접근해서 양 도시의 시차(12)만큼 빼준다.

2) 낮, 밤에 따라 상이한 백그라운드와 포그라운드를 어떻게 구현할 것인가?

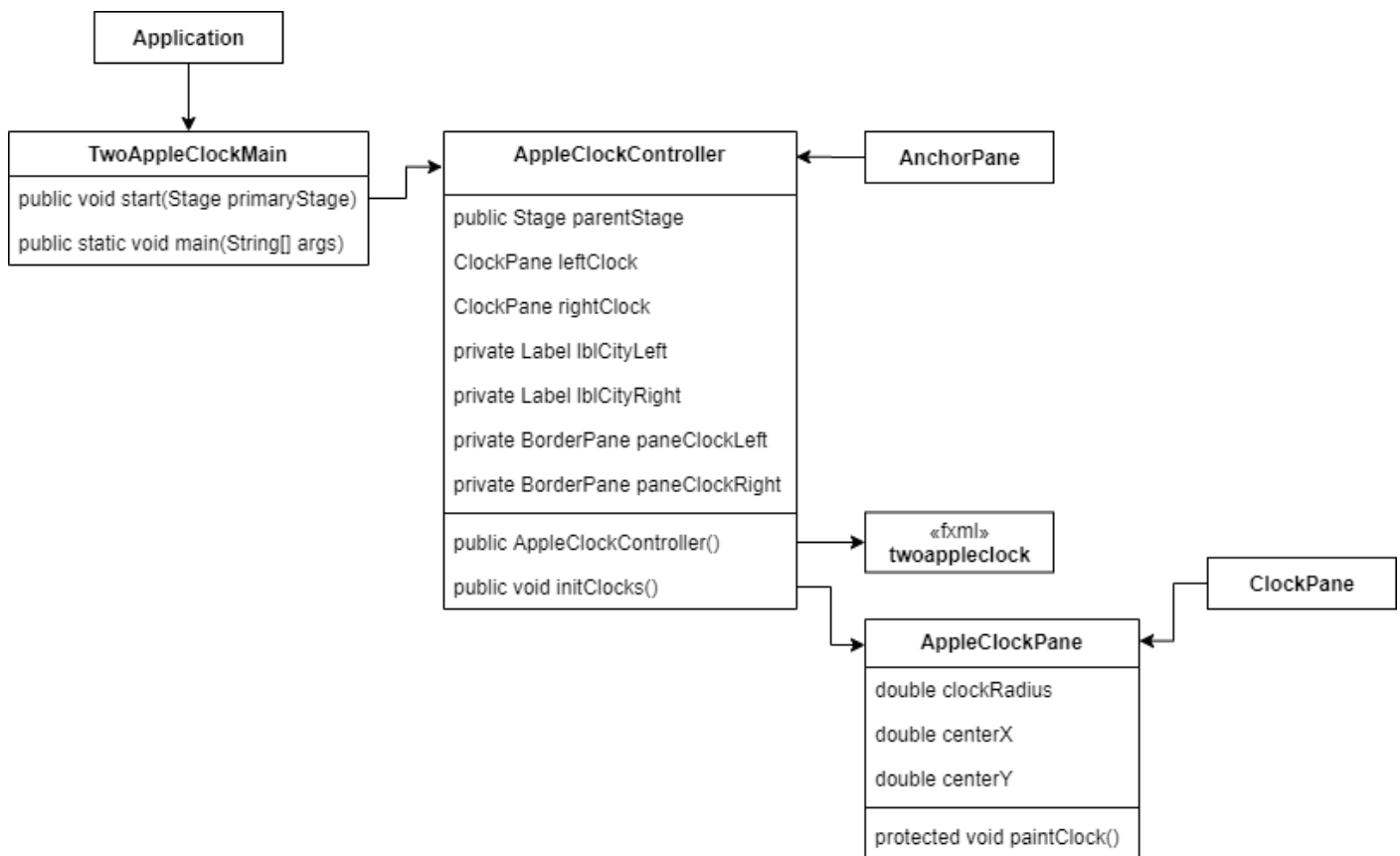
우선 낮 시간을 오전 6시부터 오후 5시 59분까지로 설정했다. 나머지는 모두 밤시간으로 설정한다. 이를 조건문으로 만들어서 AppleClockPane내부에 각 백그라운드와 포그라운드를 설정하는 코드마다 삽입한다. 기존 코드에선 백그라운드의 색이 생성자에서 설정되기 때문에 사용자 정의 생성자를 제거하고 paintClock메소드 내부에 조건문을 삽입한다.

3) 도시 이름의 폰트와 사이즈를 어떻게 바꿀 것인가?

Gluon에서 제공하는 JavaFX Scene Builder를 이용해서 fxml코드를 수정한다.

2-2. 설계 결과(프로그램 구성도, 필요 함수, 클래스 설명)

1) 프로그램 구성도



2) 필요함수, 클래스 설명

1) AppleClockController

메인프로그램의 루트 컨테이너 역할을 하는 클래스이다.

2) public AppleClockController()

AppleClockController의 생성자로서, fxml파일을 불러온 후 해당 클래스에 적용하는 역할을 한다. 그 후에 initClocks() 메소드를 호출한다.

3) public void initClocks()

시계 객체를 초기화하고 화면상에 보여주는 역할을 한다. 이벤트를 처리하는 과정에서 시계의 시간이 초기화된다.

4) AppleClockPane

시계의 세부적인 디자인을 결정하는 클래스이다. 상위 클래스인 ClockPane클래스의 paintClock()메소드를 오버라이딩하여 사용한다. 이때, 특정 인스턴스 변수(this.hour)에 따라 조건적으로 디자인이 바뀐다.

3. 결론 및 소감

3.1 과제 해결 중 생긴 문제와 해결 과정

문제 1: Scene Builder를 사용할 수 없었다.

-> JavaFX 플러그인을 설치한 뒤 Scene Builder를 사용하려고 했으나, 자바를 처음 설치할 때 경로를 바꾼 탓인지 Scene Builder의 실행파일을 찾지 못해서 이클립스에 연결해주지 못했다. 결국 부득이하게 이클립스 외부에서 Gluon을 설치했다. 설치한 폴더 내부에 있는 Scene Builder.exe를 이클립스에 연결해주고 나서야 이 문제를 해결할 수 있었다.

문제 2: 백그라운드 설정이 제대로 이루어지지 않았다.

-> 기존 코드엔 AppleClockPane의 생성자에서 기본 백그라운드 설정이 이루어졌다. 그래서 원래 생성자 코드를 내버려두고 paintClock()메소드에 조건문을 삽입해서 낮, 밤에 따른 상이한 백그라운드를 나타내려 했다. 처음엔 아무 문제없이 잘 작동되는 것처럼 보였지만, 오후 6시가 지나니 양 시계 모두에서 같은 백그라운드가 나타나는 문제가 발생했다. 생각해보니 생성자에 매개변수가 없기 때문에, 굳이 생성자를 쓸 필요가 없었다. 결국 사용자 정의 생성자를 제거하고 백그라운드 설정이 모두 paintClock() 내부에서 이루어지도록 코드를 수정했다.

3.2 배운 점 & 소감

지난 번 과제를 할 때는 레이아웃을 설정하는 데 많은 시간이 필요했다. 이번 과제의 경우 기본 코드에서 레이아웃이 잘 잡혀 있기도 했고, FXML을 이용하니 코드를 작성하기가 편리했다. 지난 번 일일이 코드를 작성할 때 보다 반절 이상 시간이 단축됐다. FXML의 중요성을 깨닫게 됐다. 그리고 디자인을 변경하던 도중 FXML과 CSS를 같이 쓸 수 있다는 사실을 알게 되었다. 이를 이용해서 다음 과제물의 디자인을 한층 발전시키도록 하겠다.