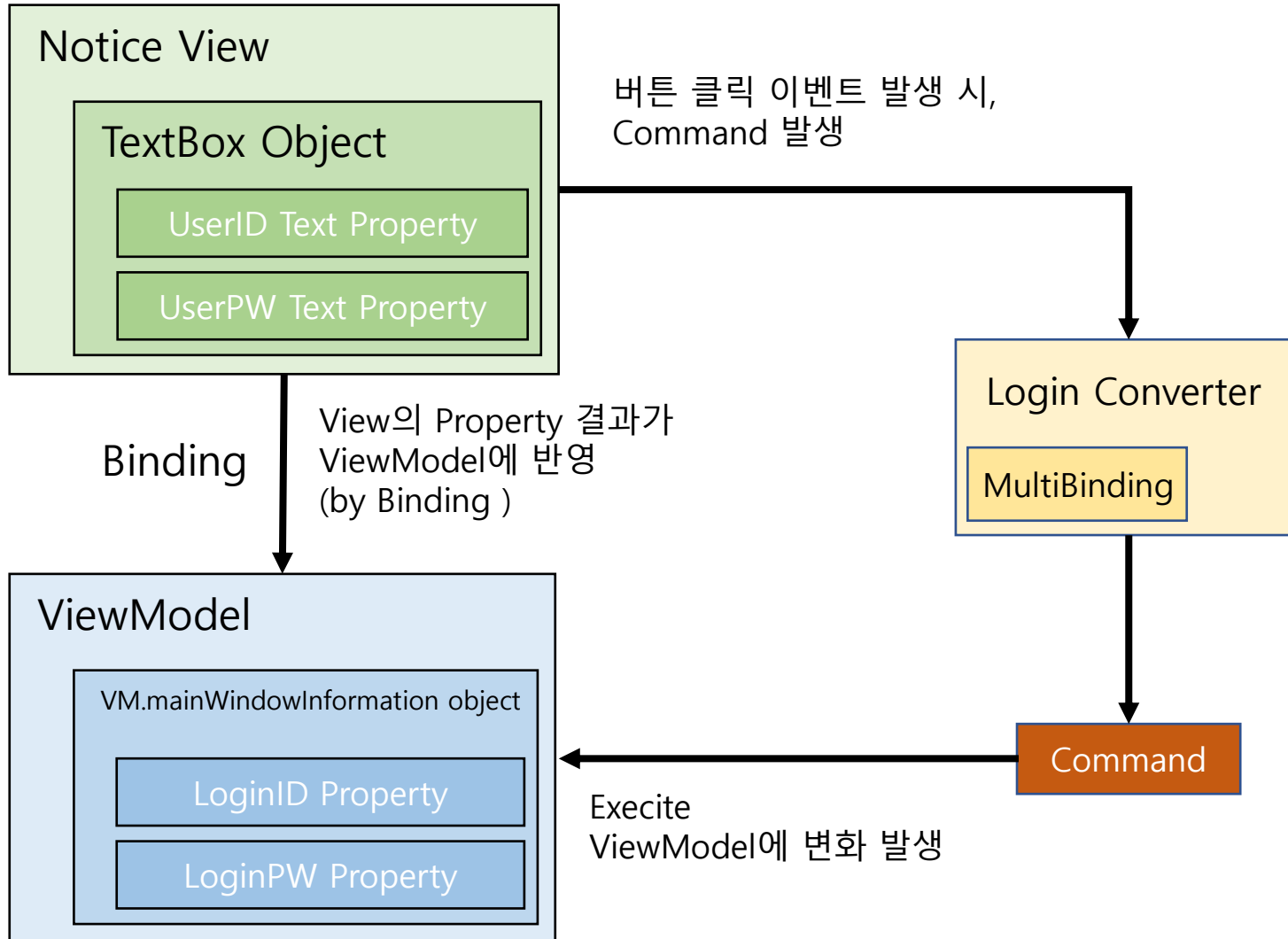




201820946 노준혁/

202021060 김소연/

202011611 구성재/



### 1. 목적

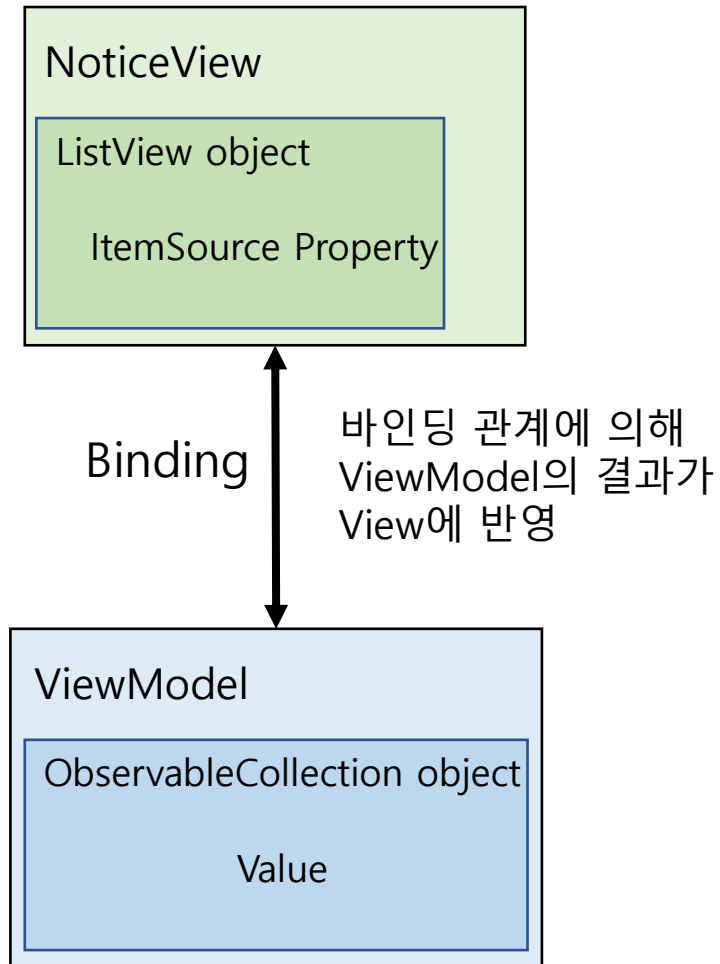
: Notice View의 UserID, UserPw TextBox의 Text Property 값이 변경될 때 ViewModel의 LoginId, LoginProperty 데이터에 반영하기 위해 Binding 사용

### 2. 방법

: Command Interface에 의한 ViewModel 변화 유도  
→ Login Converter의 MultiBinding을 위한 사용자지정 IMultiValueConverter를 사용해 Binding  
→ 두 Text Property 동시에 Command의 parameter의 배열인자로 전달  
→ 데이터 반영

### 3. 이점

: 하나의 바인딩 연결과정으로 여러 데이터를 반영할 수 있다.



### 1. 목적

: 크롤링으로 가져온 데이터를 ListView에 반영하기 위함

### 2. 방법

: 단일 개체에 대한 Binding이 아닌 데이터 집합에 대한 Binding으로 데이터를 연결하기 위해

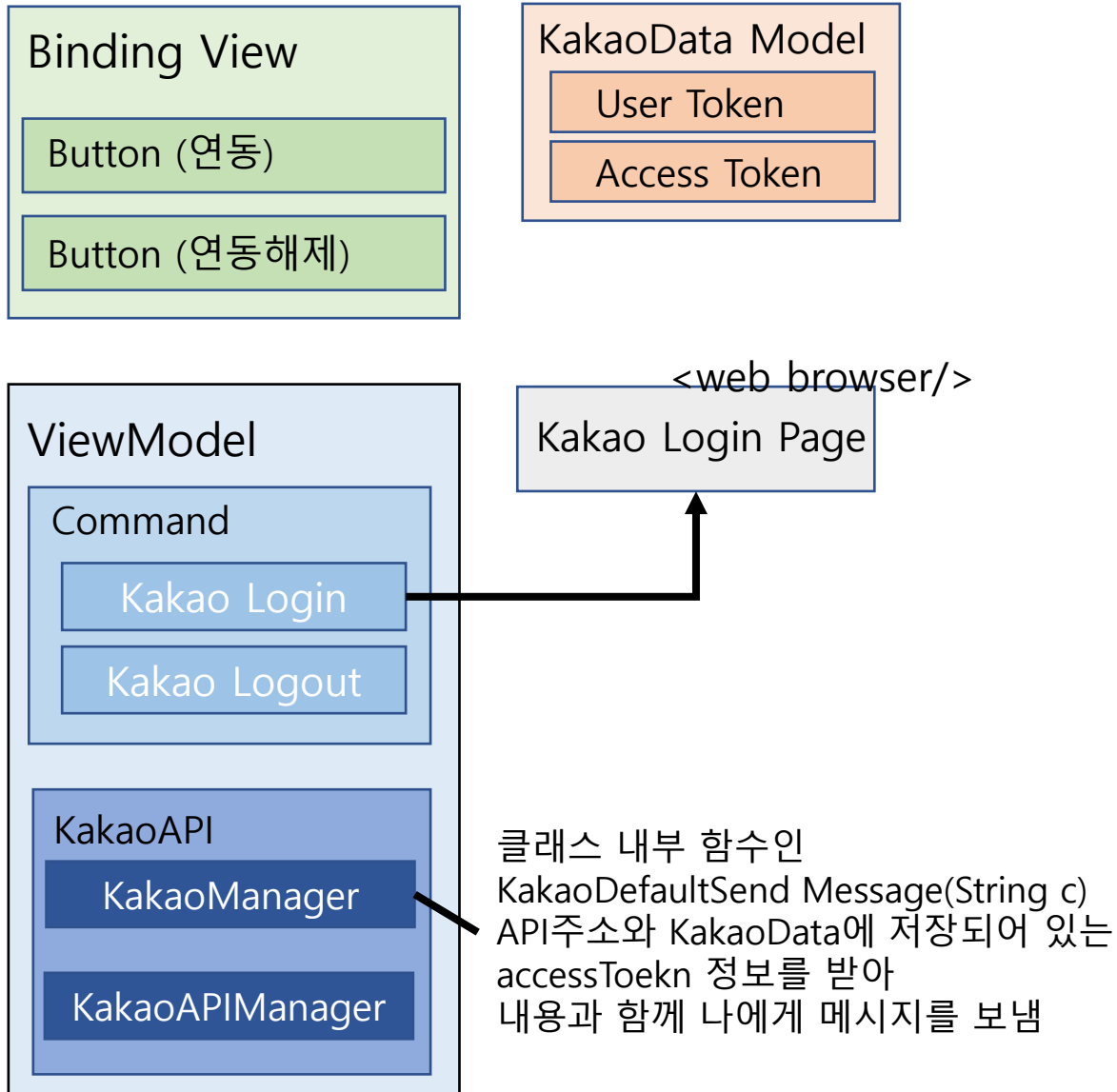
ViewModel의 ObservableCollection 클래스 객체인 Lms\_Data\_Main을 생성

→ ListView object의 ItemsSource Property와 Binding

→ ViewModel의 결과가 ListView에 반영

### 3. 이점

: Collection에서 삽입/삭제가 발생할 때 마다 View가 자동으로 업데이트 된다.



### 1. 목적

: 공지사항이 업데이트

→ 기존 내용과 다른 내용이 있다면 카톡으로 알림을 보내기 위함

### 2. 방법

: Binding View부분에서 연동 Button Click event

→ Command execute

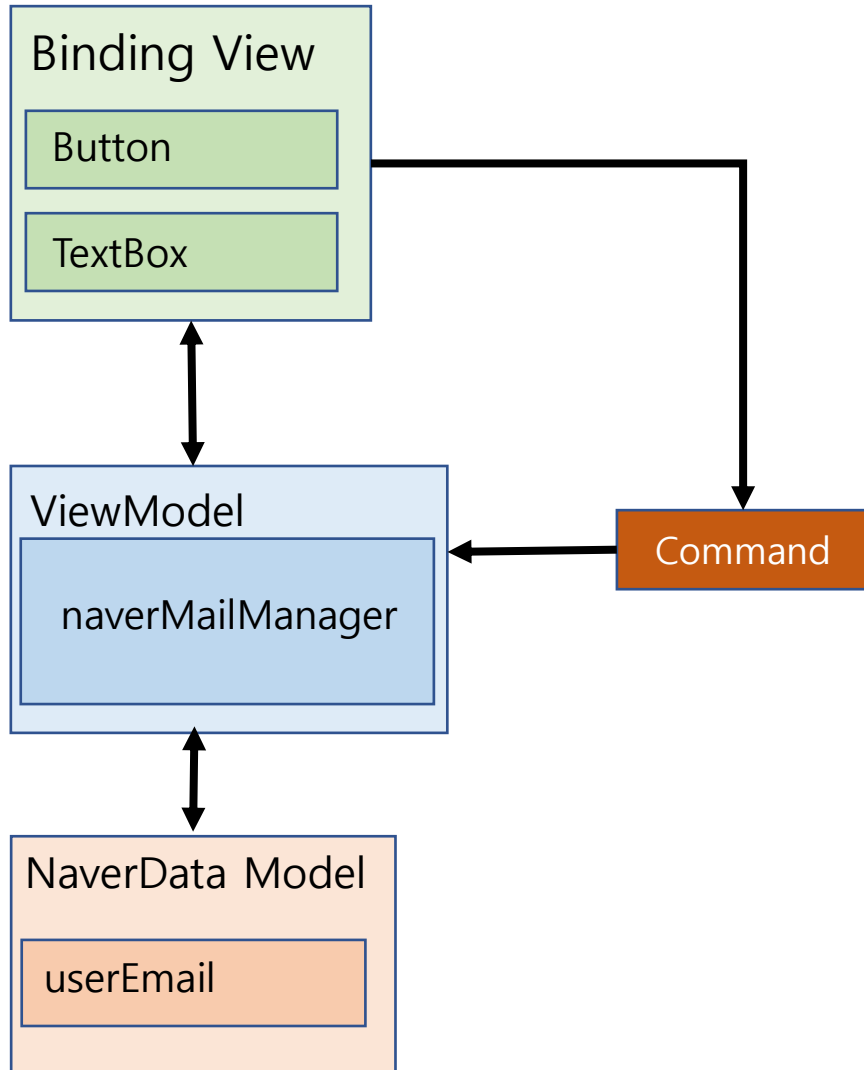
→ 카카오톡 로그인 API 오픈

→ <WebBrowser/>태그를 이용

→ 사용자 token 받음 → KakaoData 저장

### 3. 이점

: 일일이 확인하지 않아도 자동으로 어떤 것이 업데이트 되었는지 빠르게 알 수 있다.



### 1. 목적

: 강의자료와 강의 레포트 업데이트 내용을 메일 보내기 위함

### 2. 방법

: Binding View에 있는 TextBox로 네이버 아이디 값 받음  
→ Button Click Event  
→ ViewModel에 있는 Command와 바인딩  
→ Parameter.ToString() 으로 네이버 메일 체크  
→ NaverData Model의 userEmail에 정보 담기  
→ 엑셀 값비교 → 강의자료, 강의레포트의 변경 사항 여부 확인  
→ SMTP방식으로 미리 지정된 형식으로 메일 보내기

### 3. 이점

: 자동으로 어떤 것이 업데이트 되었는지 확인 가능.