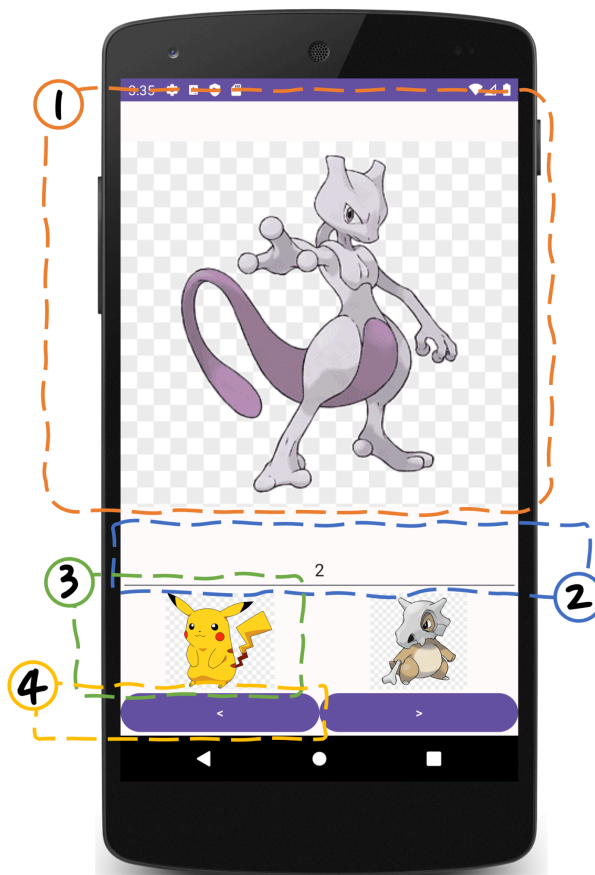


과제#1: 사진앱 만들기

과제 제출 기간 및 방법

- 과제 제출 기간은 제출일로부터 10일 동안임. 제출 마감 일시는 11월 24일 23시 59분임.
- 제출 기간 이후 제출 시 늦은 기간 만큼 감점임(분당 0.1%)
- Andorid 프로젝트를 Zip파일로 압축 후 이ური 11주차 과제에 제출
 - 과제 제출 시 구글 아이디로 로그인 필요함. 구글 아이디가 없으면 가입 필요.
 - Andorid Studio에서 다음 메뉴로 Zip파일을 생성 후 제출: “File > Export > Export to Zip Files”
 - 파일명을 “학번_이름”으로 변경 후 제출. 예: 2023123114_김모프.zip
- 안드로이드 API는 “API 24 (“Nougat”; Android 7.0)을 사용하시오. 그리고 앱 평가 시 사용되는 시뮬레이터의 환경은 “Medium Phone API 28” 입니다.

화면 디자인 설명



<그림 1: 앱 화면 예시>

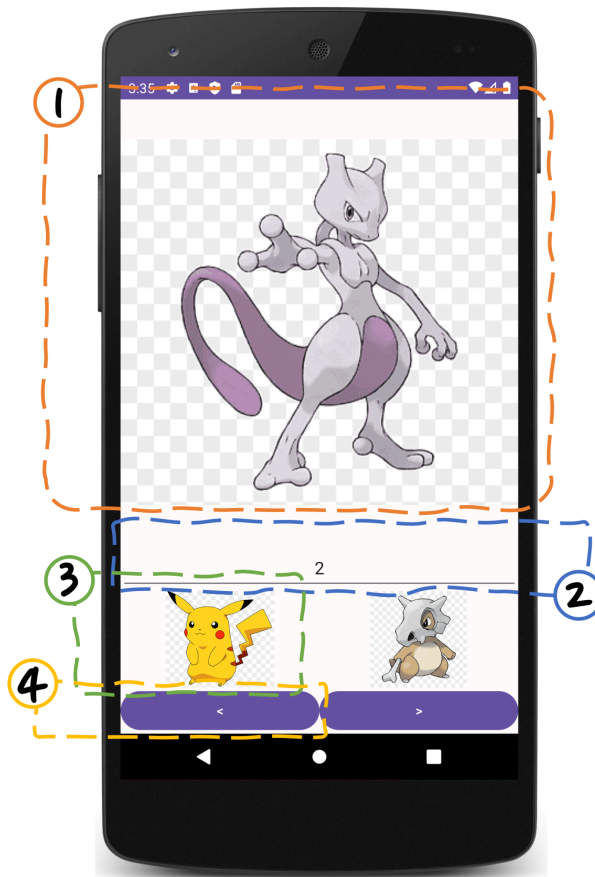
- 앱의 세로 화면은 그림 1처럼 (1) ImageView, (2) EditText, (3) ImageView x 2, (4) Button x 2이 있음. 앱 가로 화면은 세로 화면과 차이가 있음. 자세한 화면 예시는 표1에서 확인 가능함
- [영상](#)에서 앱 작동 영상을 확인 가능함. 다운로드 후 크롬 웹 브라우저에서 재생 가능함. 0:00~1:36 동안은 세로(Portrait) 화면임. 1:36에 세로 화면에서 가로(Landscape) 화면 전환됨. 즉, 1:37부터는 가로 화면임.

- Android Studio의 에뮬레이터에서 과제에 첨부된 APK파일(“app-debug.apk”)을 실행시켜 확인도 가능함. 파일을 다운로드 후 에뮬레이터의 스마트폰 화면 위로 드래그 앤 드랍하면 실행됨

<표 1: 앱 화면 예시>

	세로(Portrait) 화면	가로(Landscape) 화면
사진 1 표시 화면		
사진 2 표시 화면		
사진 20 표시 화면		

세로 화면 디자인 설명



<그림 1: 앱 화면 예시>

ImageView (1) 설명:

- ImageView (1)는 화면의 가장 상단에 위치함. 너비는 부모 레이아웃의 너비와 같음
- ImageView는 EditText에 숫자 입력(EditText 설명 참조) 또는 '<' 또는 '>' Button 클릭(Button 설명 참조) 시 적절한 사진을 표시함
- ImageView에 보여줄 사진 20개는 [사진링크](#)에서 다운로드 가능함. EditText에 숫자 입력 또는 '<' 또는 '>' Button 클릭 시 사진 이름에 있는 번호에 따라 화면에 사진을 보여줘야함(EditText 설명 또는 Button 설명 참조)
- 가로화면에서 세로화면 전환 시 또는 반대 경우에 표시 되는 사진이 동일해야함. 예를 들어, 가로화면에서 4번 사진을 보여주고 있었다면 세로화면 전환 후 동일하게 4번 사진을 보여주고 있어야함

EditText (2) 설명:

- EditText (2)는 ImageView (1)와 ImageView (3) 사이에 위치함. EditText (2)의 높이는 48dp임. 너비는 부모 레이아웃의 너비와 같음
- '0'과 양의 실수를 입력할 수 있는 키보드가 표시됨. 사진 번호 입력 시 해당 사진이 ImageView (1)에 표시됨. 예컨대 2 입력 시 [사진링크](#)의 사진 2('2.png')를 ImageView(1)에 표시함.
- 입력된 숫자가 1보다 작거나 20보다 크면 으면 "사진은 1~20까지만 있습니다."라는 Toast 창으로 보여준 후 해당 숫자와 가장 가까운 숫자에 해당하는 사진을 보여줌. 즉 1보다 작으면 사진 1을 표시함. 20보다 크면 사진 20을 표시함
- 입력된 값이 소수점을 가진 실수라면 소수점 첫번째 자리에서 반올림한 숫자와 가장 가까운 번호를 가진 사진을 보여준다. 예컨대 2.4 입력 시 사진 2를 ImageView (1)에 표시함. 2.7 입력 시 사진 3('3.jpg')을 ImageView (1)에 표시함

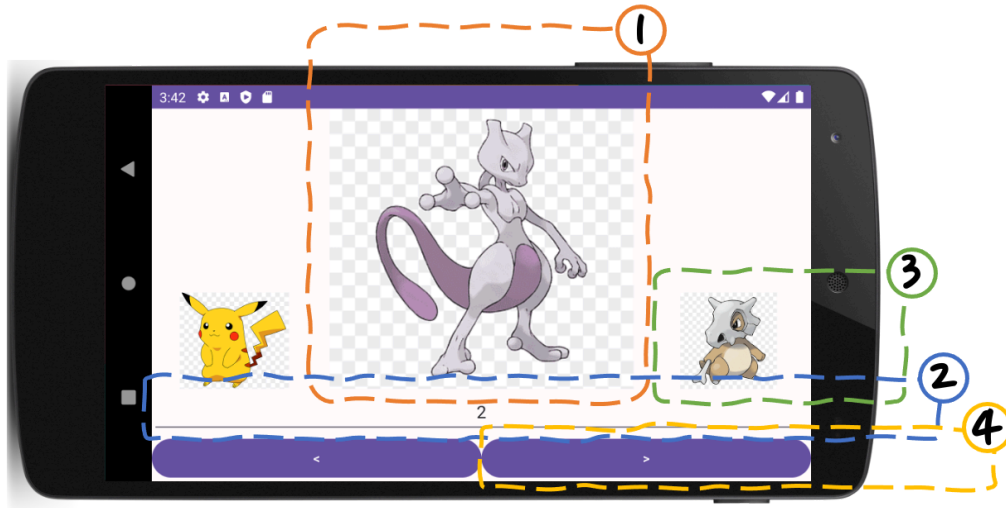
ImageView (3) 설명:

- ImageView (3)는 EditText (2)와 Button (4)의 중간에 위치함. 너비는 부모 레이아웃 너비의 반임(50%)
- 왼쪽 ImageView (3)은 ImageView (1)에 표시된 사진의 이전 사진을 표시함. 예컨대 ImageView (1)에 사진 2가 표시되어 있다면 왼쪽 ImageView (3)에는 사진1이 표시됨. 이전 사진이 없다면 왼쪽 ImageView (3)은 보이지 않음
- 반대로 오른쪽 ImageView (3)은 ImageView (1)에 표시된 사진의 다음 사진을 표시함. 예컨대 ImageView (1)에 사진 2가 표시되어 있다면 오른쪽 ImageView (3)에는 사진3이 표시됨. 다음 사진이 없다면 오른쪽 ImageView (3)은 보이지 않음

Button (4) 설명:

- “<”, “>”이라고 표시된 Button 두개가 가장 하단에 위치함. 버튼 너비는 부모 레이아웃 너비의 반임(50%)
- “<” Button 클릭 시 이전 사진을 보여줌. 이전 사진이 없다면 화면에 버튼이 표시되지 않음
- “>” Button 클릭 시 다음 사진을 보여줌. 다음 사진이 없다면 화면에 버튼이 표시되지 않음

세로 화면 디자인 설명



<그림 1: 앱 화면 예시>

ImageView (1) 설명:

- ImageView (1)는 가로 화면의 중간, EditText (2) 위에 위치함. 너비는 부모 레이아웃 너비의 반임(50%)
- ImageView는 EditText에 숫자 입력(EditText 설명 참조) 또는 '<' 또는 '>' Button 클릭(Button 설명 참조) 시 적절한 사진을 표시함
- EditText에 숫자 입력 또는 '<' 또는 '>' Button 클릭 시 사진 이름에 있는 번호에 따라 화면에 사진을 보여줘야함(EditText 설명 또는 Button 설명 참조)

EditText (2) 설명:

- EditText (2)는 ImageView (1)와 Button (4) 사이에 위치함. EditText (2)의 높이는 48dp임. 너비는 부모 레이아웃의 너비와 같음
- '0'과 양의 실수를 입력할 수 있는 키보드가 표시됨. 사진 번호 입력 시 해당 사진이 ImageView (1)에 표시됨. 예컨대 2 입력 시 사진 2('2.png')를 ImageView(1)에 표시함.
- 입력된 숫자가 1보다 작거나 20보다 크면 으면 "사진은 1~20까지만 있습니다."라는 Toast 창으로 보여준 후 해당 숫자와 가장 가까운 숫자에 해당하는 사진을 보여줌. 즉 1보다 작으면 사진 1을 표시함. 20보다 크면 사진 20을 표시함
- 입력된 값이 소수점을 가진 실수라면 소수점 첫번째 자리에서 반올림한 숫자와 가장 가까운 번호를 가진 사진을 보여준다. 예컨대 2.4 입력 시 사진 2를 ImageView (1)에 표시함. 2.7 입력 시 사진 3('3.jpg')을 ImageView (1)에 표시함

ImageView (3) 설명:

- ImageView (3)는 부모 레이아웃의 왼쪽과 오른쪽에 위치함. 그리고 EditText (2)의 위에 위치함. 너비는 부모 레이아웃 너비의 1/4임(25%)
- 왼쪽 ImageView (3)은 ImageView (1)에 표시된 사진의 이전 사진을 표시함. 예컨대 ImageView (1)에 사진 2가 표시되어 있다면 왼쪽 ImageView (3)에는 사진1이 표시됨. 이전 사진이 없다면 왼쪽 ImageView (3)은 보이지 않음

- 반대로 오른쪽 **ImageView (3)**은 **ImageView (1)**에 표시된 사진의 다음 사진을 표시함. 예컨대 **ImageView (1)**에 사진 2가 표시되어 있다면 오른쪽 **ImageView (3)**에는 사진3이 표시됨. 다음 사진이 없다면 오른쪽 **ImageView (3)**은 보이지 않음

Button (4) 설명:

- “<”, “>”이라고 표시된 **Button** 두개가 가장 하단에 위치함. 버튼 너비는 부모 레이아웃 너비의 반임(50%)
- “<” **Button** 클릭 시 이전 사진을 보여줌. 이전 사진이 없다면 화면에 버튼이 표시되지 않음
- “>” **Button** 클릭 시 다음 사진을 보여줌. 다음 사진이 없다면 화면에 버튼이 표시되지 않음

힌트

- 자바 코드에서 **EditText** 이벤트 처리 방법: **EditText**의 입력 완료 이벤트는 **TextView.OnEditorActionListener** 이벤트 리스너로 처리 가능함. **EditText**에서 입력 완료 시 **TextView.OnEditorActionListener** 인터페이스에 정의된 **onEditorAction(TextView v, int actionId, KeyEvent event)** 메서드가 불러짐.
- 자바 코드에서 **drawable** 폴더 내 이미지의 아이디값을 얻는 방법: **res** 폴더에 있는 특정 리소스를 아이디 값(**identifier**)을 찾을 때는 **Resources** 클래스의 **getIdentifier (String name, String defType, String defPackage)** 메서드를 사용하면 됨. 예컨대 아래 코드는 **res/drawable** 폴더에 있는 **"imageOne"** 파일의 아이디값을 반환함:

```
getResources().getIdentifier("imageOne","drawable", getPackageName());
```