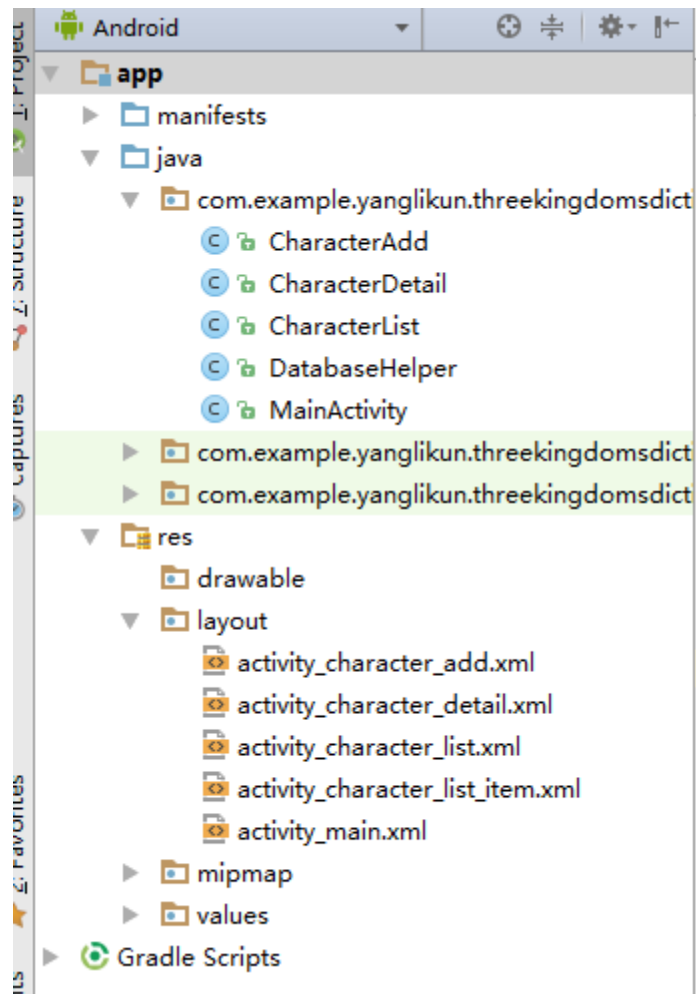


文件结构：



其中：

CharacterAdd.java → activity_character_add.xml
CharacterDetail → activity_character_detail.xml
CharacterList → activity_character_list.xml
MainActivity → activity_main.xml

activity_character_list_item.xml 是人物列表里面 item 要用的布局文件
DataBaseHelper 是数据库的类，这部分我来做吧，你们可以不用管。

我添加的只是上图里面的 5 个 activity 和 5 个 xml 文件（如果有其它文件要添加的和我说）
每个 xml 文件需要的控件和控件的 id 号已经给出，其它属性要自己写（命名格式什么的就不要纠结了）

分工：

杨金辉：实现 MainActivity 和 activity_main.xml
杨立坤：实现 CharacterDetail 和 activity_character_detail.xml
杨志华：实现 CharacterList 和 activity_character_list.xml
张凯：实现 CharacterAdd 和 activity_character_add.xml

细节：

涉及数据库读取操作的地方，你们不用实现，用注释标出来，到时我会自己添加。

布局文件：大概和图片里的差不多就行了，属性自定。（当然也可以添加别的控件）

Activity 要实现的功能：

① MainActivity

- a) 点击“人物”按钮要跳转到人物列表（CharacterList）界面，不需要接收返回信息。
- b) 点击“轶事”或“战争”显示“待实现”对话框。

② CharacterDetail

- a) 点击播放按钮更换按钮图片，并每 5 秒更换一个人物信息，
- b) 点击停止按钮（和播放按钮是同一个）更换按钮图片，停止播放。
- c) 点击 modify 按钮，利用 startActivityForResult 把含有信息（image, name, gender, birthday, birthplace, master）的 intent 传给添加人物界面（CharacterAdd），并接收返回信息，修改数据库文件。
- d) 点击 delete 按钮，显示对话框并提示是否从数据库删除该人物，执行相应操作。
- e) 点击 return 按钮，用 setResult 方法返回，其中包含 bundle（“isDataChanged”，boolean），如果用户执行了 modify 或者 delete 操作，上面的值是 true，否则是 false。

③ CharacterList

- a) 点击返回按钮跳转回主界面（MainActivity）
- b) 点击 add 按钮用 startActivityForResult 跳转到人物添加界面（CharacterAdd），intent 里面不要加任何信息。并接收人物添加界面返回来的 intent，如果 intent 里面的“isCancel”属性是 false，就在数据库添加人物信息，然后更新数据，否则不做任何改变。
- c) 点击 search 按钮弹出对话框，请求用户输入名字。名字正确就直接跳转到人物详情界面（CharacterDetail）。
- d) 点击列表的每一个 item，把含有信息（仅包含 name）的 intent 用 startActivityForResult 方法传给人物详情界面。并接收返回信息，根据“isDataChanged”的值 true 或 false 判断是否需要更新显示的数据（从数据库读取）。

④ CharacterAdd

- a) 根据传过来的 intent 填写 ImageView 和各个 EditText。
- b) 点击 return 按钮用返回带有 bundle（“isCancel”，false）的 intent。
- c) 点击 confirm 按钮，先判断是否有 EditText 为空，有就显示对话框提示信息，没有就把含有人物信息和 bundle（“isCancel”，true）的 intent 通过 setResult 方法返回。