

1st Assignment -To Do List report

NetID:tz164 Name:Tian Zhou

The layout design is based on the **activity_main.xml** file. Set the weight for: To Do List **TextView**,input title **EditText**,input description **EditText**,Task **ListView**,add button **Button** as 1:1:1:5:1. Match the whole page, so the add button is always in the bottom.

Write a customized **MyAdapter** based on **BaseAdapter**. In **getView()** function, we can see each item has two **TextView** type variables "title" and "content" and one checkbox.

In the **MainActivity.java** has two main functions:

btn.setOnClickListener() is a click listener for add button in order to add texts from **EditText** to a new task.

listView.setOnItemLongClickListener() is a long click listener for each item. When holding the item for a long time, the item will be deleted

Part1

1. To do list title: Remove the default title and add a new **TextView** as a general title. Put the text in the center and set the background color blue.
2. Create an **EditText** as the "input task title box" and another **EditText** as "input task description". Set these two text as a hint for each **EditText**. Variable in adapter:
[title: **titleTextView**] [input: **contentTextView**]
3. Add button: Set a onclick listener (**setOnClickListener()** function) for add button. Whenever click the button, run the function that add two new string read from two **EditText** box to the **dataList** and notify the adapter, then set the adapter for **ListView**.
4. Create a function in **setOnClickListener()** function and see whether the input title or input description is empty. If one of them is empty, show "Title or description cannot be empty." on the screen and not add the task. Else, add a new task. This judgement statement make sure that the task should have a title and description.
5. Set the gravity value to "center" in **list_view.XML** file for each **TextView**.
6. In **activity_main.XML** file, A **LinearLayout** has a full size height which contains title,two edit parts,listview and button. Then distribute this layout as weight 1:1:1:5:1. So the **ListView** is always occupy the full range under the form fields to the add button.
7. In **activity_main.XML** file, set **marginTop** and **marginBottom** value as "2dp" for **Button**, so that the add button has a limit distance to adjacent component.

Part2

Add a **setOnItemLongClickListener()** function for **ListView** and listen the long click event. In the function, remove the data in the **dataList** for item oriented by **position**. Then notify the adapter of change and Set the adapter to **ListView**. The item will be removed from the task list.

Additional part

First,write a **CheckBox** in the **list_view.xml** and set the layout gravity to the right. In the adapter **MyAdapter.java** file, declare a private checkbox and a listener function **setOnClickListener()**. When the checkbox is clicked, the item will be removed.