Android ListView 待办事项实验报告

一、实验目的

1.掌握 ListView 控件的基本使用

2.自定义适配器实现及数据绑定方法

二、实验内容与核心代码实现

1. 数据模型类 TaskItem

成员变量定义

```java
public class TaskItem implements Serializable {
    4 usages
    private String taskName;
    4 usages
    private String dueDate;
    3 usages
    private boolean isCompleted;
    5 usages
    private boolean isFavorite;
    2 usages
    private long id;
    1 usage
    private static long idCounter = 0;
```

构造函数实现

```java
public TaskItem(String taskName, String dueDate) {
    this.id = idCounter++;
    this.taskName = taskName;
    this.dueDate = dueDate;
    this.isCompleted = false;
    this.isFavorite = false;
}
```

Getter/Setter 方法

```java
public String getTaskName() {
    return taskName;
}

no usages
public void setTaskName(String taskName) {
    this.taskName = taskName;
}

1 usage
public String getDueDate() {
    return dueDate;
}

no usages
public void setDueDate(String dueDate) {
    this.dueDate = dueDate;
}

no usages
public boolean isCompleted() {
    return isCompleted;
}
```

2. 主活动 MainActivity

ListView 初始化

```java
etSearch = findViewById(R.id.etSearch);
btnAdd = findViewById(R.id.btnAdd);
listView = findViewById(R.id.listView);

adapter = new TaskAdapter( context: this);
listView.setAdapter(adapter);
```

适配器设置

```java
adapter = new TaskAdapter( context: this);
listView.setAdapter(adapter);
```

示例数据填充

```java
1 usage
public TaskAdapter(Context context) {
    this.context = context;
    this.taskList = new ArrayList<>();
    this.filteredList = new ArrayList<>();

    taskList.add(new TaskItem( taskName: "Complete Android Homework", dueDate: "2025-12-20"));
    taskList.add(new TaskItem( taskName: "Buy groceries", dueDate: "2024-05-01"));
    taskList.add(new TaskItem( taskName: "Walk the dog", dueDate: "2024-04-20"));
    taskList.add(new TaskItem( taskName: "Call John", dueDate: "2024-04-23"));
    filteredList.addAll(taskList);
}
```

3. 自定义适配器 TaskAdapter

getView 方法重写、视图复用机制

```java
@Override
public View getView(int position, View convertView, ViewGroup parent) {
    ViewHolder holder;

    if (convertView == null) {
        convertView = LayoutInflater.from(context).inflate(R.layout.itemtask, parent, attachToRoot: false);
        holder = new ViewHolder();
        holder.ivFavorite = convertView.findViewById(R.id.ivFavorite);
        holder.tvTaskName = convertView.findViewById(R.id.tvTaskName);
        holder.tvDueDate = convertView.findViewById(R.id.tvDueDate);
        holder.cbCompleted = convertView.findViewById(R.id.cbCompleted);
        convertView.setTag(holder);
    } else {
        holder = (ViewHolder) convertView.getTag();
    }

    TaskItem task = filteredList.get(position);
```

数据绑定逻辑

```
TaskItem task = filteredList.get(position);

holder.tvTaskName.setText(task.getTaskName());
holder.tvDueDate.setText(task.getDueDate());

if (task.isCompleted()) {
    holder.tvTaskName.setPaintFlags(holder.tvTaskName.getPaintFlags() | android.graphics.Paint.STRIKE_THRU_TEXT_FLA
    holder.tvTaskName.setTextColor(context.getResources().getColor(android.R.color.darker_gray));
} else {
    holder.tvTaskName.setPaintFlags(holder.tvTaskName.getPaintFlags() & (~android.graphics.Paint.STRIKE_THRU_TEXT_F
    holder.tvTaskName.setTextColor(context.getResources().getColor(android.R.color.black));
}

updateFavoriteIcon(holder.ivFavorite, task.isFavorite());

holder.cbCompleted.setChecked(task.isCompleted());

holder.ivFavorite.setOnClickListener( View v -> {
    task.toggleFavorite();
    updateFavoriteIcon(holder.ivFavorite, task.isFavorite());
    updateOriginalItem(task);
    notifyDataSetChanged();
});
```

4. 布局文件设计

activity_main.xml 布局结构

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="@android:color/white">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:padding="16dp"
        android:background="@android:color/white"
        android:elevation="4dp">

        <EditText
            android:id="@+id/etSearch"
            android:layout_width="0dp"
            android:layout_height="48dp"
            android:layout_weight="1"
            android:hint="请输入关键词"
            android:paddingStart="16dp"
            android:paddingEnd="16dp"
            android:background="@drawable/edittext_background"
            android:inputType="text"
            android:maxLines="1"
            android:imeOptions="actionDone" />
```

```xml
        <Button
            android:id="@+id/btnAdd"
            android:layout_width="48dp"
            android:layout_height="48dp"
            android:layout_marginStart="12dp"
            android:background="@drawable/button_background"
            android:text="+"
            android:textSize="24sp"
            android:textColor="@android:color/white" />
    </LinearLayout>

    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:background="#EEEEEE" />

    <ListView
        android:id="@+id/listView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:divider="#EEEEEE"
        android:dividerHeight="1dp"
        android:padding="0dp" />
</LinearLayout>
```

itemtask.xml 自定义列表项布局

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="16dp"
    android:background="?android:attr/selectableItemBackground">

    <ImageView
        android:id="@+id/ivFavorite"
        android:layout_width="24dp"
        android:layout_height="24dp"
        android:layout_marginEnd="12dp"
        android:src="@drawable/ic_star_border"
        android:contentDescription="收藏按钮" />

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:orientation="vertical">

        <TextView
            android:id="@+id/tvTaskName"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textSize="18sp"
            android:textColor="@android:color/black" />

        <TextView
            android:id="@+id/tvDueDate"
            android:layout_width="match_parent"
```
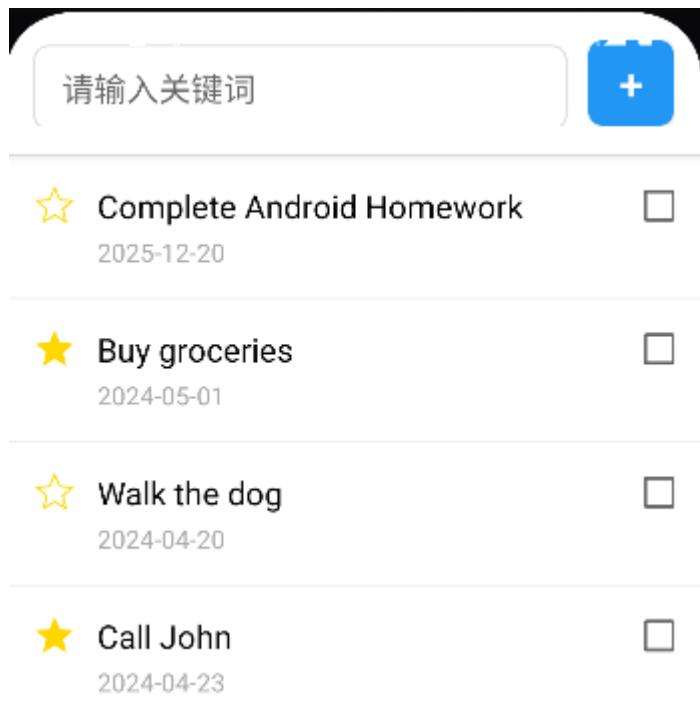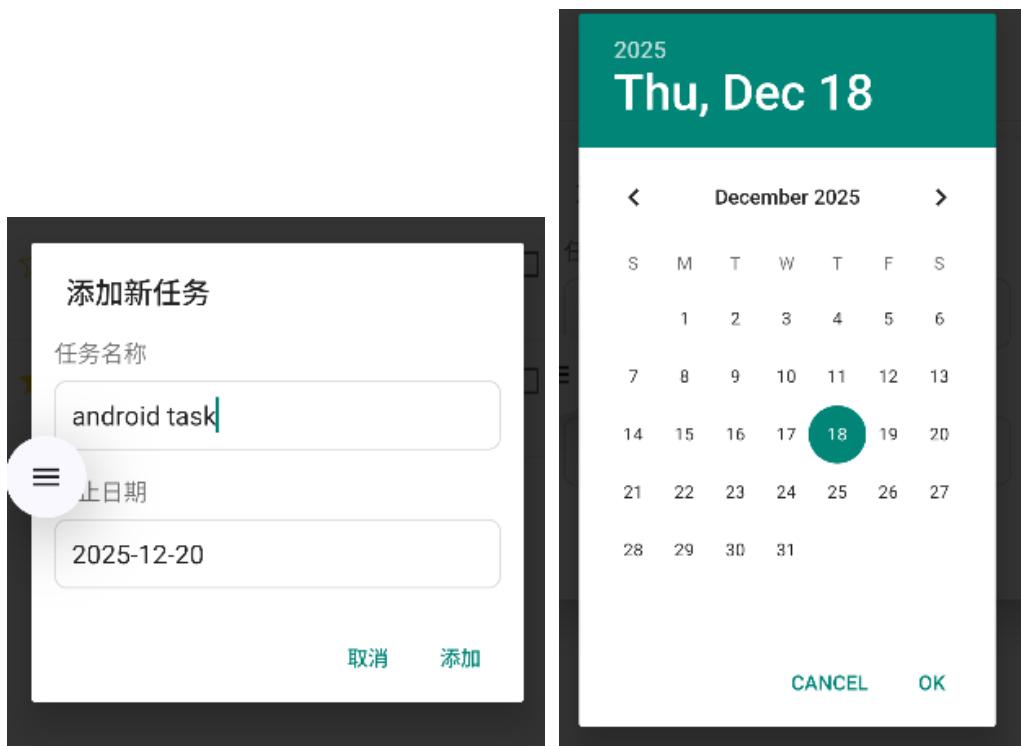
```xml
        <TextView
            android:id="@+id/tvDueDate"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textSize="14sp"
            android:textColor="@android:color/darker_gray"
            android:layout_marginTop="4dp" />
    </LinearLayout>

    <CheckBox
        android:id="@+id/cbCompleted"
        android:layout_width="24dp"
        android:layout_height="24dp"
        android:layout_marginStart="12dp" />
</LinearLayout>
```

四、实验结果展示

1.列表项正确显示任务名称和截止日期可添加收藏或取消收藏

2.添加任务功能正常

添加新任务

任务名称

android task

止日期

2025-12-20

取消　添加

Thu, Dec 18

‹　December 2025　›

S	M	T	W	T	F	S
		1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

CANCEL　OK

请输入关键词　＋

☆ android task
2025-12-20　☐

3.完成任务划掉正常

请输入关键词　＋

☆ ~~android task~~
2025-12-20　☑