

[android] 解决DatePickerDialog和TimePickerDialog控件取消按钮问题

原创 Eastmount 最后发布于2015-01-07 05:01:38 阅读数 6866 ☆ 收藏

展开



Python+TensorFlow人工智能

该专栏为人工智能入门专栏，采用Python3和TensorFlow实现人工智能相关算法。前期介绍安装流程、基础语法...



Eastmount

¥9.90

去订阅

一. 问题提出

在Android程序中，我们通常需要使用DatePicker来设置日期，TimePicker来设置时间。其基本步骤是：

1.先定义DatePicker和TimePicker布局

2.然后通过Calendar类获得系统时间

3.接着通过init方法将日期传递给DatePicker初始化日期控件

4.在响应点击事件中可以通过DatePicker的getYear()、getDayOfMonth()、getMonth()函数获取具体日期

同时可以在OnDateChangeListener事件中监听日期变化，设置时间函数方法同理。但是使用DatePickerDialog或TimePickerDialog控件时会遇到的一个问题，它就是android版本4.0后没有取消(Cancel)按钮，同时点击界面任何部分都能获取日期或时间，据说它是版本存在的BUG。对比图如下所示：



但是我们期待的效果如下图所示：



我采取的解决方法是通过自定义XML布局，经过DatePicker和TimePicker控件实现，同时在AlertDialog中设置取消按钮和确定按钮，通过函数setNegativeButton()和setPositiveButton()实现。

二. 简单实现日期和时间控件

简单实现方法非常简单，不需要设置日期或时间的XML布局，直接通过new DatePickerDialog或

TimePickerDialog即可实现。代码如下：

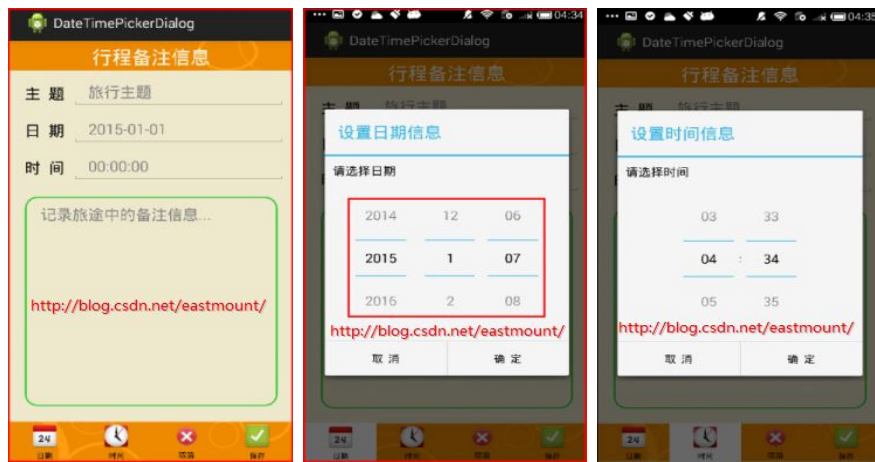
```
// 点击"日期" 按钮布局 设置日期
layoutDate.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View v) {
        new DatePickerDialog(MainActivity.this, new DatePickerDialog.OnDateSetListener() {
            @Override
            public void onDateSet(DatePicker view, int year, int month, int day) {
                // TODO Auto-generated method stub
                mYear = year;
                mMonth = month;
                mDay = day;
                // 更新EditText控件日期 小于10加0
                dateEdit.setText(new StringBuilder().append(mYear).append("-")
                    .append((mMonth + 1) < 10 ? "0" + (mMonth + 1) : (mMonth + 1))
                    .append("-")
                    .append((mDay < 10) ? "0" + mDay : mDay) );
            }
        }, calendar.get(Calendar.YEAR), calendar.get(Calendar.MONTH),
        calendar.get(Calendar.DAY_OF_MONTH) ).show();
    }
});
```

其中mYear、mMonth、mDay是定义变量，同时dateEdit是一个EditText控件用于显示具体日期，当数字小于10时前补0，如2015-01-01。下面是设置时间，显示效果和第三部分的效果相同。

```
// 点击"时间" 按钮布局 设置时间
layoutTime.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View v) {
        new TimePickerDialog(MainActivity.this,
            new TimePickerDialog.OnTimeSetListener() {
                @Override
                public void onTimeSet(TimePicker view, int hour, int minute) {
                    // TODO Auto-generated method stub
                    mHour = hour;
                    mMinute = minute;
                    // 更新EditText控件时间 小于10加0
                    timeEdit.setText(new StringBuilder()
                        .append(mHour < 10 ? "0" + mHour : mHour).append(":")
                        .append(mMinute < 10 ? "0" + mMinute : mMinute).append(":00")
                    );
                }
            }, calendar.get(Calendar.HOUR_OF_DAY),
            calendar.get(Calendar.MINUTE), true).show();
    }
});
```

三. 自定义布局实现

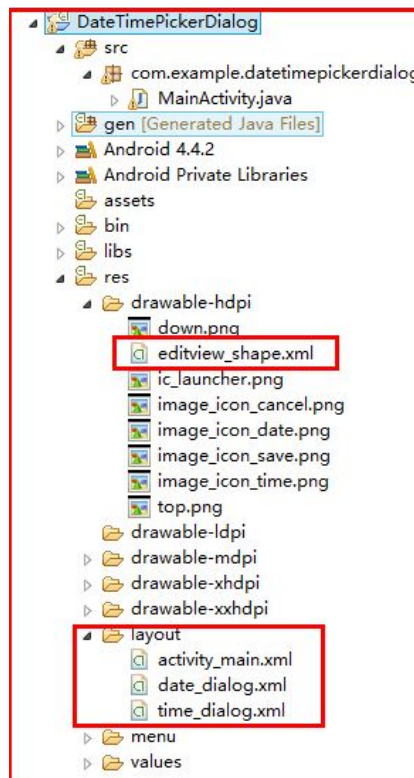
下面是自定义布局实现，而且我自认为该界面布局非常好看，这也是我“随手拍”项目自己最后总结出来的一个比较欣赏的界面吧！希望你也喜欢，效果如下：



如图所示，界面中日期、时间EditText不可编辑，需要通过点击底部布局来设置。同时备注信息的EditText通过自定义背景实现，设置日期、时间中有取消按钮。

源代码：

1.项目结构如下图所示



2.activity_main.xml布局文件

```
<?xml version="1.0" encoding="UTF-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >

    <!-- 顶部路径 -->
    <RelativeLayout
        android:id="@+id/MyLayout_top"
        android:orientation="horizontal"
        android:layout_width="fill_parent"
        android:layout_height="40dp"
        android:background="@drawable/top"
        >
```

```

        android:layout_alignParentTop="true"
        android:gravity="center">
<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="25sp"
    android:textColor="#ffffff"
    android:layout_marginTop="5dp"
    android:layout_marginBottom="5dp"
    android:text="行程备注信息" />
</RelativeLayout>

<!-- 底部按钮 -->
<RelativeLayout
    android:id="@+id/MyLayout_bottom"
    android:orientation="horizontal"
    android:layout_width="fill_parent"
    android:layout_height="55dp"
    android:layout_alignParentBottom="true"
    android:background="@drawable/down"
    android:gravity="center">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="horizontal"
        android:layout_alignParentBottom="true" >
        <LinearLayout
            android:id="@+id/layout_date"
            android:layout_width="wrap_content"
            android:layout_height="match_parent"
            android:layout_weight="1"
            android:gravity="center"
            android:orientation="vertical" >
            <ImageView
                android:id="@+id/image_date"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:padding="3dp"
                android:src="@drawable/image_icon_date" />
            <TextView
                android:id="@+id/textview_date"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="日期"
                android:textColor="#000000"
                android:textSize="10sp" />
        </LinearLayout>
        <LinearLayout
            android:id="@+id/layout_time"
            android:layout_width="wrap_content"
            android:layout_height="match_parent"
            android:layout_weight="1"
            android:gravity="center"
            android:orientation="vertical" >
            <ImageView
                android:id="@+id/image_time"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:padding="3dp"
                android:src="@drawable/image_icon_time" />
            <TextView

```

```

        android:id="@+id/textview_time"

        android:layout_height="wrap_content"
        android:text="时间"
        android:textColor="#000000"
        android:textSize="10sp" />
</LinearLayout>
<LinearLayout
    android:id="@+id/layout_cancel"
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:gravity="center"
    android:orientation="vertical" >
    <ImageView
        android:id="@+id/image_cancel"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="3dp"
        android:src="@drawable/image_icon_cancel" />
    <TextView
        android:id="@+id/textview_cancel"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="取消"
        android:textColor="#000000"
        android:textSize="10sp" />
</LinearLayout>
<LinearLayout
    android:id="@+id/layout_save"
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:gravity="center"
    android:orientation="vertical" >
    <ImageView
        android:id="@+id/image_save"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="3dp"
        android:src="@drawable/image_icon_save" />
    <TextView
        android:id="@+id/textview_save"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="保存"
        android:textColor="#000000"
        android:textSize="10sp" />
</LinearLayout>
</LinearLayout>
</RelativeLayout>

<!-- 显示图片 -->
<RelativeLayout
    android:id="@+id/Content_Layout"
    android:orientation="horizontal"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:layout_above="@id/MyLayout_bottom"
    android:layout_below="@id/MyLayout_top"
    android:background="#eee8cc"

```

```

android:gravity="center">
    <LinearLayout
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:orientation="vertical" >

        <LinearLayout
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:layout_alignParentBottom="true" >
            <TextView
                android:layout_width="wrap_content"
                android:layout_height="30dp"
                android:textColor="#000000"
                android:text="主 题"
                android:layout_marginLeft="20dp"
                android:layout_marginTop="10dp"
                android:textSize="20dp" />
            <EditText
                android:id="@+id/showtitle"
                android:layout_width="fill_parent"
                android:layout_height="40dp"
                android:layout_marginTop="10dp"
                android:layout_marginLeft="10dp"
                android:layout_marginRight="10dp"
                android:hint="旅行主题"
                android:layout_weight="1"
                android:textSize="20dp"
                android:textColor="#000000"
                android:paddingLeft="20dp" />
        </LinearLayout>
        <LinearLayout
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:layout_alignParentBottom="true" >
            <TextView
                android:layout_width="wrap_content"
                android:layout_height="30dp"
                android:textColor="#000000"
                android:text="日 期"
                android:layout_marginLeft="20dp"
                android:layout_marginTop="10dp"
                android:textSize="20dp" />
            <EditText
                android:id="@+id/showdate"
                android:layout_width="fill_parent"
                android:layout_height="40dp"
                android:layout_marginTop="10dp"
                android:layout_marginLeft="10dp"
                android:layout_marginRight="10dp"
                android:hint="2015-01-01"
                android:layout_weight="1"
                android:textSize="20dp"
                android:textColor="#000000"
                android:editable="false"
                android:paddingLeft="20dp" />
        </LinearLayout>
    </LinearLayout>

```

```

        android:orientation="horizontal"
        android:layout_alignParentBottom="true" >

            android:layout_width="wrap_content"
            android:layout_height="30dp"
            android:textColor="#000000"
            android:text="时 间"
            android:layout_marginLeft="20dp"
            android:layout_marginTop="10dp"
            android:textSize="20dp" />
        <EditText
            android:id="@+id/showtime"
            android:layout_width="fill_parent"
                android:layout_height="40dp"
                android:layout_marginTop="10dp"
                android:layout_marginLeft="10dp"
                android:layout_marginRight="10dp"
                android:hint="00:00:00"
                android:layout_weight="1"
                android:textSize="20dp"
            android:textColor="#000000"
            android:editable="false"
            android:paddingLeft="20dp" />
    </LinearLayout>
    <LinearLayout
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:orientation="horizontal"
        android:layout_alignParentBottom="true" >
        <!-- 设置运行多行 设置圆角图形 黑色字体
            android:lineSpacingExtra 设置行间距
            android:typeface 设置字体
            android:paddingLeft 设置距离
        -->
        <EditText
            android:id="@+id/editText1"
            android:layout_height="fill_parent"
            android:layout_width="fill_parent"
            android:textColor="#000000"
            android:layout_margin="20dp"
            android:textSize="20sp"
            android:hint="记录旅途中的备注信息..."
            android:maxLength="500"
            android:lineSpacingExtra="5dip"
            android:typeface="sans"
            android:paddingLeft="20dp"
            android:paddingRight="20dp"
            android:paddingTop="10dp"
            android:singleLine="false"
            android:background="@drawable/editview_shape"
            android:gravity="top" />
    </LinearLayout>
</LinearLayout>
</RelativeLayout>

</RelativeLayout>

```

3.date_dialog.xml文件: 日期控件布局

```

<?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:padding="10dip" >

        <TextView
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:singleLine="true"
            android:text="请选择日期"
            android:textColor="#000000"
            android:textSize="16sp" />

        <DatePicker
            android:id="@+id/date_picker"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="5dip"
            android:layout_gravity="center" >

        </DatePicker>

    </LinearLayout>

```

4.time_dialog.xml文件: 时间控件布局

```

<?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:padding="10dip" >

    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:singleLine="true"
        android:text="请选择日期"
        android:textColor="#000000"
        android:textSize="16sp" />

    <DatePicker
        android:id="@+id/date_picker"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="5dip"
        android:layout_gravity="center" >

    </DatePicker>

</LinearLayout>

```

5.drawables-hdpi中editview_shape.xml文件 实现自定义EditText控件格式

```

<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="rectangle"
    android:padding="10dp">
    <!-- 填充editView的颜色 -->
    <solid android:color="#ffffff"/>
    <!-- 设置圆角的弧度, radius半径越大, editView的边角越圆 -->
    <corners

```



```

        android:radius="15dp"
                                android:bottomRightRadius="15dp"
        android:bottomLeftRadius="15dp"
        android:topLeftRadius="15dp"
        android:topRightRadius="15dp"/>
<stroke
        android:color="#32CD32"
        android:width="4px" />
</shape>

```

6.MainActivity.java源码

```

public class MainActivity extends Activity {

    // 自定义变量
    private EditText titleEdit;
    private EditText dateEdit;
    private EditText timeEdit;
    private EditText contentEdit;
    // 底部四个布局按钮
    private LinearLayout layoutDate;
    private LinearLayout layoutTime;
    private LinearLayout layoutCancel;
    private LinearLayout layoutSave;
    // 定义显示时间控件
    private Calendar calendar; // 通过Calendar获取系统时间
    private int mYear;
    private int mMonth;
    private int mDay;
    private int mHour;
    private int mMinute;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        // 锁定屏幕
        setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);
        setContentView(R.layout.activity_main);
        // 获取对象
        titleEdit = (EditText) findViewById(R.id.showtitle);
        dateEdit = (EditText) findViewById(R.id.showdate);
        timeEdit = (EditText) findViewById(R.id.showtime);
        contentEdit = (EditText) findViewById(R.id.editText1);
        layoutDate = (LinearLayout) findViewById(R.id.layout_date);
        layoutTime = (LinearLayout) findViewById(R.id.layout_time);
        layoutCancel = (LinearLayout) findViewById(R.id.layout_cancel);
        layoutSave = (LinearLayout) findViewById(R.id.layout_save);

        calendar = Calendar.getInstance();

        // 点击"日期"按钮布局 设置日期
        layoutDate.setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {

                // 通过自定义控件AlertDialog实现
                AlertDialog.Builder builder = new AlertDialog.Builder(MainActivity.this);
                View view = (LinearLayout) getLayoutInflater().inflate(R.layout.date_dialog,
                    null);

                final DatePicker datePicker = (DatePicker)

```

```

view.findViewById(R.id.date_picker);
// 设置日期简略显示 否则详细显示 包括: 星期\周
datePicker.setCalendarViewShown(false);
// 初始化当前日期
calendar.setTimeInMillis(System.currentTimeMillis());
datePicker.init(calendar.get(Calendar.YEAR), calendar.get(Calendar.MONTH),
                calendar.get(Calendar.DAY_OF_MONTH), null);
// 设置date布局
builder.setView(view);
builder.setTitle("设置日期信息");
builder.setPositiveButton("确 定", new DialogInterface.OnClickListener() {
@Override
public void onClick(DialogInterface dialog, int which) {
// 日期格式
StringBuilder sb = new StringBuffer();
sb.append(String.format("%d-%02d-%02d",
                        datePicker.getYear(),
                        datePicker.getMonth() + 1,
                        datePicker.getDayOfMonth()));
dateEdit.setText(sb);
mYear = datePicker.getYear();
mMonth = datePicker.getMonth();
mDay = datePicker.getDayOfMonth();
dialog.cancel();
});
builder.setNegativeButton("取 消", new DialogInterface.OnClickListener() {
@Override
public void onClick(DialogInterface dialog, int which) {
dialog.cancel();
}
});
builder.create().show();
});
layoutDate.setOnTouchListener(new OnTouchListener() { // 设置布局背景
@Override
public boolean onTouch(View v, MotionEvent event) {
if(event.getAction() == MotionEvent.ACTION_DOWN) {
layoutDate.setBackgroundColor(Color.WHITE);
layoutTime.setBackgroundColor(Color.TRANSPARENT);
layoutCancel.setBackgroundColor(Color.TRANSPARENT);
layoutSave.setBackgroundColor(Color.TRANSPARENT);
}
return false;
}
});

// 点击"时间" 按钮布局 设置时间
layoutTime.setOnClickListener(new OnClickListener() {
@Override
public void onClick(View v) {
// 自定义控件
AlertDialog.Builder builder = new AlertDialog.Builder(MainActivity.this);
View view = (LinearLayout) getLayoutInflater().inflate(R.layout.time_dialog,
null);

final TimePicker timePicker = (TimePicker)
view.findViewById(R.id.time_picker);
// 初始化时间
calendar.setTimeInMillis(System.currentTimeMillis());
timePicker.setIs24HourView(true);
timePicker.setCurrentHour(calendar.get(Calendar.HOUR_OF_DAY));
timePicker.setCurrentMinute(Calendar.MINUTE);

```

```

        // 设置time布局
        builder.setView(view);
        builder.setTitle("设置时间信息");
        builder.setPositiveButton("确 定", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                mHour = timePicker.getCurrentHour();
                mMinute = timePicker.getCurrentMinute();
                // 时间小于10的数字 前面补0 如01:12:00
                timeEdit.setText(new StringBuilder().append(mHour < 10 ? "0" + mHour :
mHour).append(":")
                    .append(mMinute < 10 ? "0" + mMinute : mMinute).append(":00") );
                dialog.cancel();
            }
        });
        builder.setNegativeButton("取 消", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                dialog.cancel();
            }
        });
        builder.create().show();
    });
    layoutTime.setOnTouchListener(new OnTouchListener() { // 设置布局背景
        @Override
        public boolean onTouch(View v, MotionEvent event) {
            if(event.getAction() == MotionEvent.ACTION_DOWN) {
                layoutDate.setBackgroundColor(Color.TRANSPARENT);
                layoutTime.setBackgroundColor(Color.WHITE);
                layoutCancel.setBackgroundColor(Color.TRANSPARENT);
                layoutSave.setBackgroundColor(Color.TRANSPARENT);
            }
            return false;
        }
    });

    // 点击"取消"按钮
    layoutCancel.setOnClickListener(new OnClickListener() {
        @Override
        public void onClick(View v) {
            dateEdit.setText("");
            dateEdit.setHint("2015-01-01");
            timeEdit.setText("");
            timeEdit.setHint("00:00:00");
            contentEdit.setText("");
            contentEdit.setHint("记录旅途中的备注信息...");
        }
    });

    layoutCancel.setOnTouchListener(new OnTouchListener() { // 设置布局背景
        @Override
        public boolean onTouch(View v, MotionEvent event) {
            if(event.getAction() == MotionEvent.ACTION_DOWN) {
                layoutDate.setBackgroundColor(Color.TRANSPARENT);
                layoutTime.setBackgroundColor(Color.TRANSPARENT);
                layoutCancel.setBackgroundColor(Color.WHITE);
                layoutSave.setBackgroundColor(Color.TRANSPARENT);
            }
            return false;
        }
    });

    // 点击"保存"按钮
    layoutSave.setOnClickListener(new OnClickListener() {
        @Override

```

```

        public void onClick(View v) {

            // 确认保存按钮
            new AlertDialog.Builder(MainActivity.this).setTitle("确认保存吗? ")
                .setIcon(android.R.drawable.ic_dialog_info)
                .setPositiveButton("确 定", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {

                        /**
                         * 数据库插入操作
                         */

                    }

                })
                .setNegativeButton("返 回", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {

                    }
                }).show();
        }

    });
    layoutSave.setOnTouchListener(new OnTouchListener() { // 设置布局背景
        @Override
        public boolean onTouch(View v, MotionEvent event) {
            if(event.getAction() == MotionEvent.ACTION_DOWN) {
                layoutDate.setBackgroundColor(Color.TRANSPARENT);
                layoutTime.setBackgroundColor(Color.TRANSPARENT);
                layoutCancel.setBackgroundColor(Color.TRANSPARENT);
                layoutSave.setBackgroundColor(Color.WHITE);
            }
            return false;
        }
    });

} // End onCreate

}

```

最后希望文章对大家有所帮助！如果你知道这篇文章想要阐述的内容是什么？为什么要写这篇文章？或许它会对你有帮助，一方面是布局可能对你有启发；另一方面就是刚好遇到那个问题的同学。

在点击“保存”按钮时，也可把数据存储至数据库中调用MySQLiteOpenHelper，这里就不再介绍。写着写着就到了凌晨5点了，程序猿生活还是要改下啊~自己保重自己的身体吧！期待改正自己的作息，难难难~

(By:Eastmount 2015-1-7 凌晨5点 <http://blog.csdn.net/eastmount/>)

👍 点赞 4 ☆ 收藏 ➦ 分享 ...



Eastmount 博客专家

发布了445 篇原创文章 · 获赞 5981 · 访问量 487万+

他的留言板

关注