

显示富文本：

- 支持Html格式文本
- 支持图片点击事件
- 链接自动回调
- 方式一:依靠Html类里面的`fromHtml`解析html标签,转换成Spanned对象
- 主要的方法：**Html.fromHtml**(接收一个HTML标签的字符串)； 将这种字符串解析为相应的内容

```
mEt = (EditText) findViewById(R.id.et);
mTv = (TextView) findViewById(R.id.tv);
mStr1 = "<font color=red>我是红色文本</font>"+fuText2("我是普通文本+我是红色文本")
      +fuText3("我是caise文本");
Spanned spanned = Html.fromHtml(mStr1);
mEt.setText(spanned);
mTv.setText(spanned);
}
```

改变文本的局部颜色
通过Html.fromHtml()方法

```
//通过占位符改变文本显示的颜色
private String fuText3(String con) {
    String str = String.format("<font color='red'>%s</font>",con);
    return str;
}
private String fuText2(String C) {
    String S = String.format("<font color='green'>%s</font>",C);
    return S;
}
```

```
mEt = (EditText) findViewById(R.id.et);
mTv = (TextView) findViewById(R.id.tv);
// <a 标签标识可以将 网址 转为 链接输出展示
mStr2 = "\t我已经在活动“<a href='\"http://www.oschina.net/question/574485_242756\"' +
        \"\ttarget='\"_blank\"'>OSC长沙开源行 技术交流峰会</a>”签到! \n";
Spanned spanned = Html.fromHtml(mStr2);
mEt.setText(spanned);
mTv.setText(spanned);
}
```

将带有<a /> 标签的字符串解析为链接显示

- **方法二：通过Spannable里面的setSpan方法 完美支持EditText**
- SpannableString msp = new SpannableString(接收字符串)
- 设置字体
- msp.setSpan(new TypefaceSpan("monospace"), 0, 2, Spanned.SPAN_EXCLUSIVE_EXCLUSIVE);
- //设置字体大小(绝对值,单位:像素)
- msp.setSpan(new AbsoluteSizeSpan(20), 4, 6, Spanned.SPAN_EXCLUSIVE_EXCLUSIVE);
- //设置图片
- Drawable drawable = getResources().getDrawable(R.mipmap.ic_launcher);
drawable.setBounds(0, 0, drawable.getIntrinsicWidth(), drawable.getIntrinsicHeight());
msp.setSpan(new ImageSpan(drawable), 53, 57, Spanned.SPAN_EXCLUSIVE_EXCLUSIVE);
- mTv.setMovementMethod(LinkMovementMethod.getInstance()); //点击事件设置
- mTv.setText(msp);

setSpan第四个参数的含义

1. SPAN_EXCLUSIVE_EXCLUSIVE: 包括内容不会扩大,只会包含起点和终点,长度绝对不能为0,如果被删除的所有它们涵盖
2. SPAN_EXCLUSIVE_INCLUSIVE: 可以扩大的跨度,非0,包括文字插入他们的终点,而不是在他们的出发点,如果为0的时候
3. SPAN_INCLUSIVE_EXCLUSIVE: 可以扩大的跨度,非0,包括文字插入他们的出发点,而不是在他们的终点,如果为0的时候
4. SPAN_INCLUSIVE_INCLUSIVE: 跨度类型的扩大包括文本插入他们的起点或终点。

设置显示表情图

方式一:使用fromHtml重载方法,配合标签

表情占位符 'asasa' 表示资源的名称

```
mTv = (TextView) findViewById(R.id.tv);
String html = "天气好热。。<img src='asasa' />没关系，我有空调！";
Spanned spanned = Html.fromHtml(html, new Html.ImageGetter() {
    @Override
    public Drawable getDrawable(String source) {
        int resourceId = getResourceId(source);
        Drawable drawable = getResources().getDrawable(resourceId);
        //对图片进行压缩（此处我采用原图）
        drawable.setBounds(0, 0, drawable.getIntrinsicWidth(), drawable.getIntrinsicHeight());
        return drawable;
    }
}, null);
mTv.setText(spanned);
}

public int getResourceId(String name) {
    try {
        //利用Java反射机制获取R资源Id
        Field field = R.drawable.class.getField(name);
        return Integer.parseInt(field.get(null).toString());
    } catch (Exception e) {
        e.printStackTrace();
    }
    return 0;
}
```

方法二：使用spannable里面的setSpan

```
String html = "[pig]我<img src='pig' />已经在活动 “<a href=\\\"http://www.oschina.net/question/574485_242756\\\" target=\\\"_blank\\\">OSC长沙";
SpannableString spannableString = new SpannableString(html);
//创建ImageSpan
Bitmap bitmap = BitmapFactory.decodeResource(getResources(), R.drawable.pig);
ImageSpan imgSpan = new ImageSpan(this, bitmap);
spannableString.setSpan(imgSpan, 0, 5, Spanned.SPAN_EXCLUSIVE_EXCLUSIVE);
//设置文本
mTextView.setText(spannableString);
```