



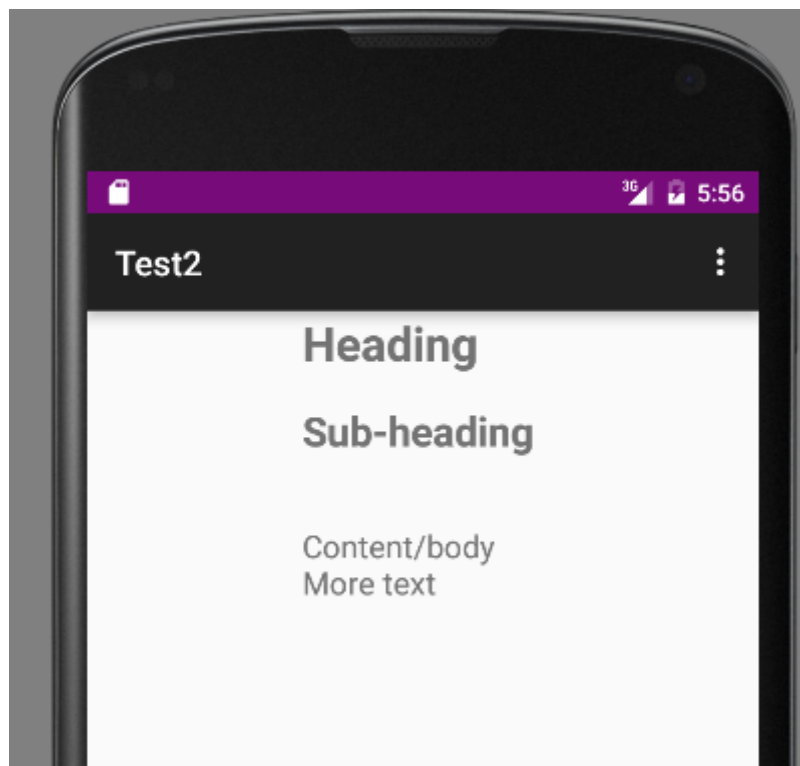
The textView is used to shows some text inside the activity. However, the biggest drawback of textView is that the text inside can't be formatted. If your are creating an app that needs to show a lot of formatted text then textView isn't the best option for you.

However, wouldn't it be nice if you could just embed HTML text inside the textView?

It would solve all our problems isn't it.

You can actually show HTML inside a textView if you follow the steps in the article.

The end result would look something like this:



The procedure is as follows:

1. Add a textView to your activity. Delete the default text generated by Android Studio, that is this line:

```
android:text="Medium Text"
```

Finally your textView should look something like this:

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:id="@+id/textView1"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true" />
```

1. Then go to the strings.xml file to create a string resource that will contain the HTML that you want to show inside the textView. However, there is one thing you need to remember. Make sure that you wrap your HTML within:

```
<![CDATA[          your_HTML_here          ]]>
```

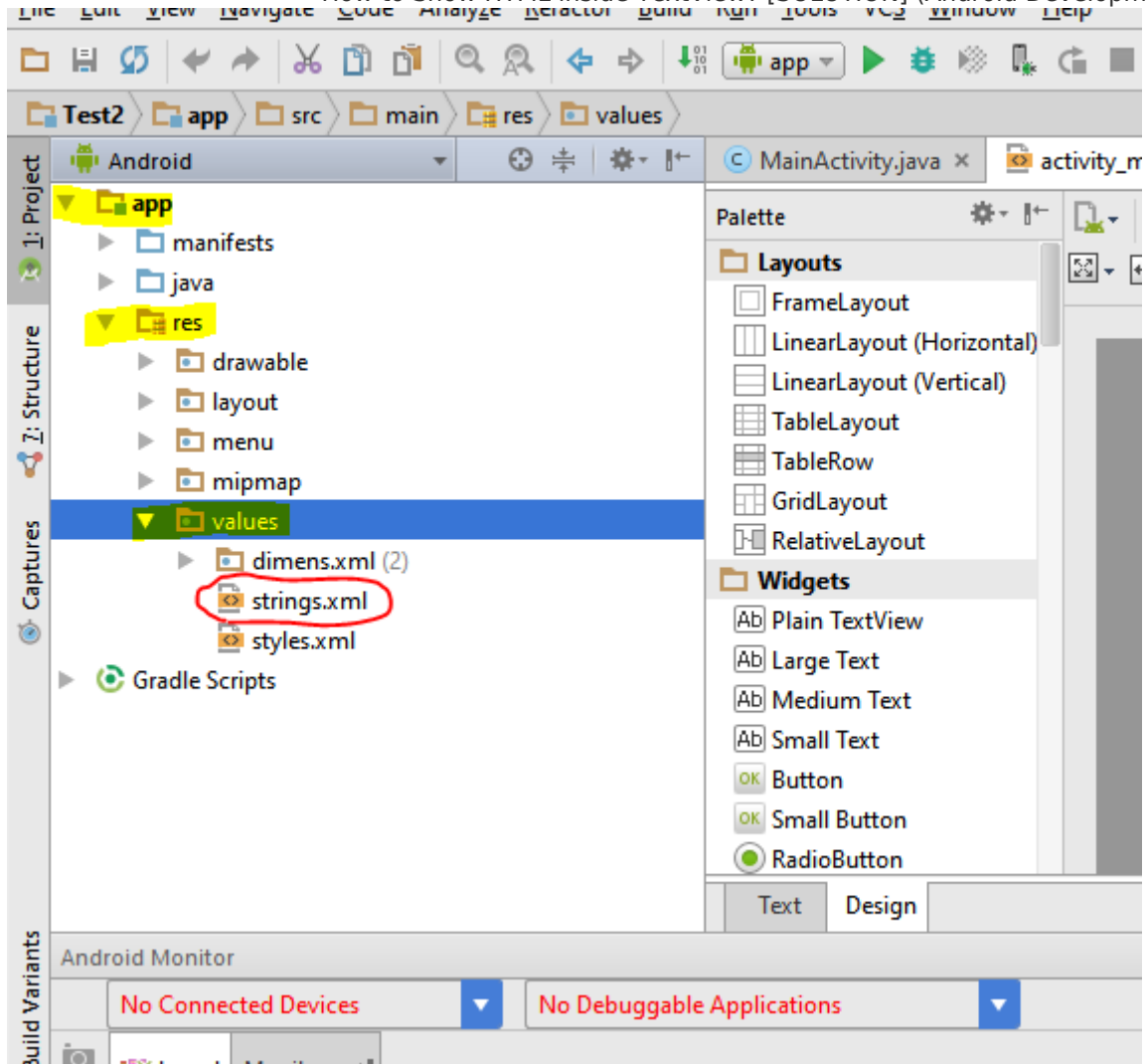
So that the string resource looks like this:

```
<string name="htmltext">
    <![CDATA[
        <h1>Heading</h1>
        <h3>Sub-heading</h3>
        <p></p>
        <br>Content/body</br>
        <br>More text</br>
    ]]>
</string>
```



strings.xml with the required HTML string resource

'strings.xml' can be found in `app>res>values> strings.xml`



location of strings.xml

- Then go to the JAVA file of your Activity and there in the onCreate method copy the following lines of code:

```
TextView newtext = (TextView) findViewById(R.id.textView1);
newtext.setText(Html.fromHtml(getString(R.string.htmltext)));
```

Make sure that the id of the textView matches the id that you assigned to your textView in step #1 and the name of the string resource matches the name of the string resource you created in step#2.

So that your JAVA file looks like this:

```

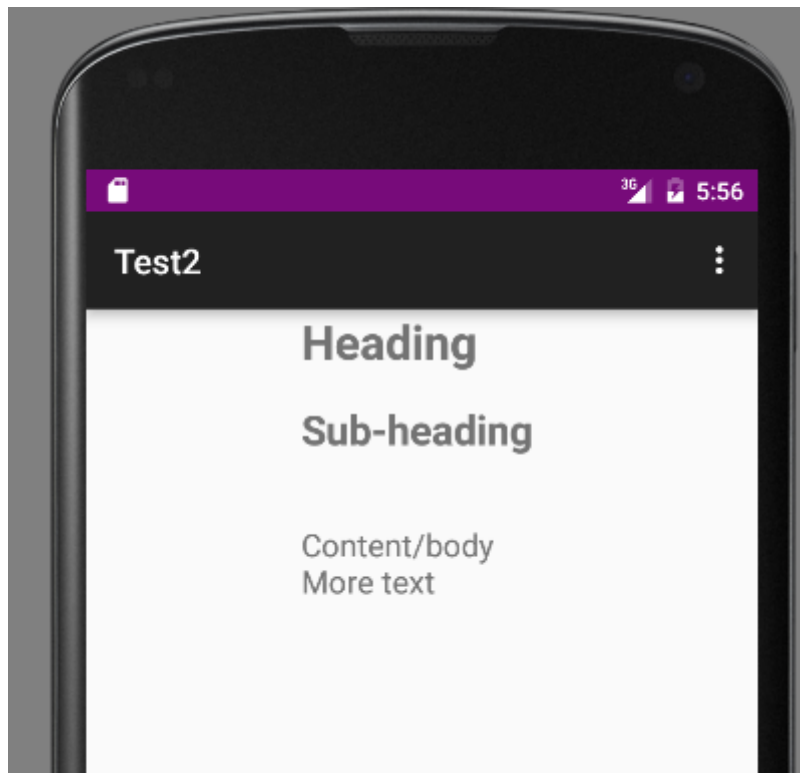
package com.bragitoff.test2;

import ...

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        TextView newText = (TextView) findViewById(R.id.textView1);
        newText.setText(Html.fromHtml(getString(R.string.htmltext)));
    }
}
    
```

Well that's all you need to do. Now if you run your app in the emulator then you can see that HTML displays correctly. Something like this:



HTML in a TextView (Example)

Hope you found it useful.

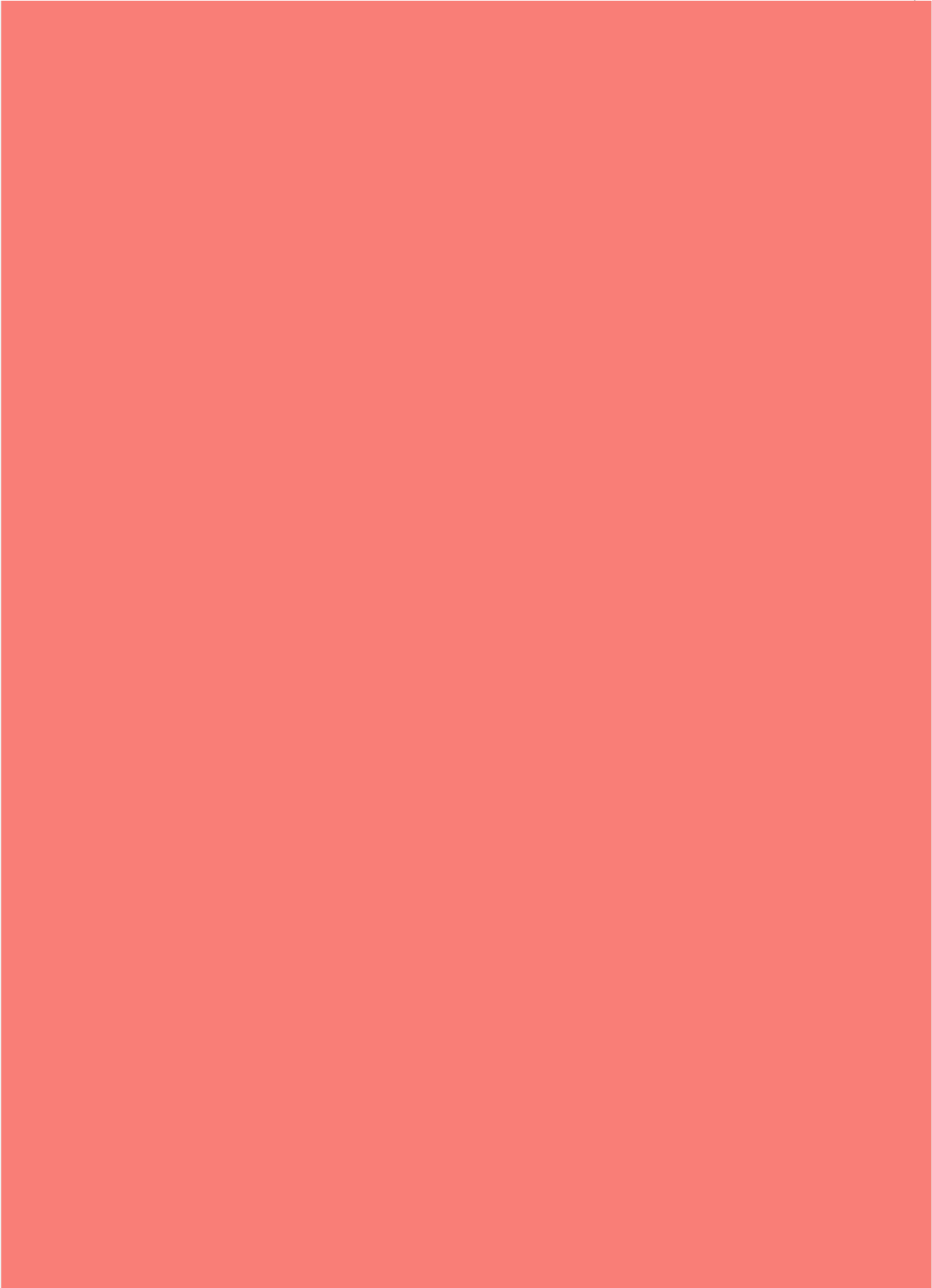
Leave your doubts/questions in the comments section below.

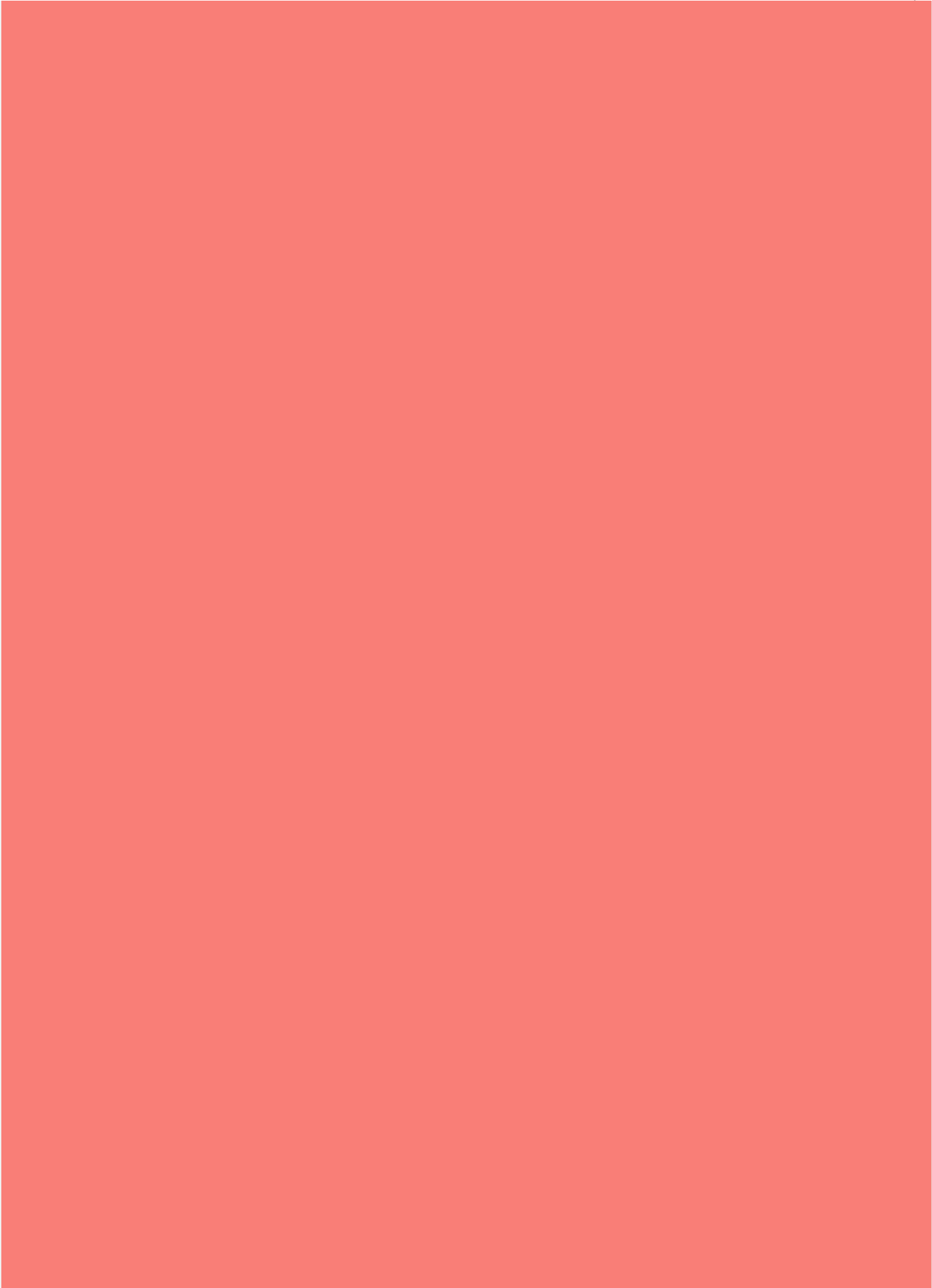


Manas Sharma

I'm a physicist specializing in computational material science with a PhD in Physics from Friedrich-Schiller University Jena, Germany. I write efficient codes for simulating light-matter interactions at atomic scales. I like to develop Physics, DFT, and Machine Learning related apps and software from time to time. Can code in most of the popular languages. I like to share my knowledge in Physics and applications using this Blog and a YouTube channel.

[manas.bragitoff.com/](https://manas.bragitoff.com/)







**Share this:**

[Click to share on Facebook \(Opens in new window\)](#)

[Click to share on Twitter \(Opens in new window\)](#)

[Click to share on WhatsApp \(Opens in new window\)](#)

[Click to share on Pinterest \(Opens in new window\)](#)

[Click to share on Reddit \(Opens in new window\)](#)

[Click to share on LinkedIn \(Opens in new window\)](#)

[Click to email a link to a friend \(Opens in new window\)](#)

[Click to print \(Opens in new window\)](#)

[Click to share on Tumblr \(Opens in new window\)](#)

[Click to share on Pocket \(Opens in new window\)](#)

[Click to share on Telegram \(Opens in new window\)](#)

[wpedon id="7041" align="center"]