

You may want to input some data in an EditText field and store that in some variable for a variety of reasons.

Official Documentation: <https://developer.android.com/reference/android/widget/EditText.html>

It could be anything from a name, email, phone number, or just some number that you would be using to calculate something, etc.

In this tutorial I am gonna show you how to store the data entered by user into some variable, on press of a button.

You would need to add an EditText to your activity's layout, and a button to submit the data. And on the click of that button, we will store the data entered by the user in the EditText field in a variable.

Here is the code you need to have in your activity's layout(xml) file:

```
<EditText
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:inputType="number"
    android:ems="10"
    android:id="@+id/input"
    android:layout_centerHorizontal="true"/>

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Submit"
    android:id="@+id/submit"
    android:onClick="store_input"
    android:layout_below="@+id/input"
    android:layout_centerHorizontal="true" />
```

In the above code we have an EditText with id: input and a submit button with id: submit. As you can also notice in the above code I have a parameter inputType set to number. You can change it according to your needs. The available options are: text, phone, textPassword, etc. which are used to specify the keyboard shown.

The line:

```
android:onClick="store_input"
```

means that the function/method called 'store_input' would be called when the button is clicked. This function/method needs to be defined in your activity's java file.

Now go to your activity's java file and create an object of EditText type:

```
EditText inputText = (EditText)findViewById(R.id.input);
```

Now we would define the function:

```
public void store_input() {View view{  
String text = inputText.getText().toString();  
}
```

The above function is called on the click of the submit button and it would get the text entered by the user and store it into a variable called text.

Now you can use this information and operate/process it.

In case you want text to be available to other methods you may want to declare it outside all the methods in the class.

The activity's java file should look somewhat like the following:

Now in the above code we get the text as a string, but you may require a number, integer, or some particular data type only.

In that case you would need to parse that data to the require data type.

The following examples will make it clear:

To convert string to double:

```
double num=Double.parseDouble.text;
```

To convert string to int:

```
int num=Integer.parseInt.text;
```



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/



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"]