

This is CS50x

OpenCourseWare

David J. Malan (<https://cs.harvard.edu/malan/>)

malan@harvard.edu

[f](https://www.facebook.com/dmalan/) (<https://www.facebook.com/dmalan/>) [G](https://github.com/dmalan) (<https://github.com/dmalan>) [@](https://www.instagram.com/davidjmalan/) (<https://www.instagram.com/davidjmalan/>) [in](https://www.linkedin.com/in/malan/)

(<https://www.linkedin.com/in/malan/>) [Q](https://www.quora.com/profile/David-J-Malan) (<https://www.quora.com/profile/David-J-Malan>) [T](https://twitter.com/davidjmalan) (<https://twitter.com/davidjmalan>)

Fiftygram

Distribution Code

Download this project's [distribution code](https://cdn.cs50.net/2019/fall/tracks/android/fiftygram/fiftygram.zip) (<https://cdn.cs50.net/2019/fall/tracks/android/fiftygram/fiftygram.zip>).

To open the distribution code, extract the ZIP, open Android Studio, select "Import project", and select the folder you extracted from the ZIP.

What To Do

- More Filters
- Saving Photos

More Filters

We've added a few different filters together, but now try experimenting with your own! Add at least one new filter of your choosing to the app. Be creative!

Saving Photos

Our app can apply filters to photos, but it would be nice if we could save those photos so we could post them elsewhere!

First, some bookkeeping. Android has a pretty strict permissions model, so your app will need to request permission to store a photo to the user's device. Different versions of Android handle these permissions differently, so for simplicity's sake, make sure your app has a minimum SDK version of 23. To set the minimum SDK version, open up `build.gradle`, and make sure you have:

```
minSdkVersion 23
```

If you don't, just change the number next to `minSdkVersion`, and then click `Sync now` !

Next, open up `AndroidManifest.xml` and add a line right above `</manifest>` :

```
<uses-permission
    android:name="android.permission.WRITE_EXTERNAL_STORAGE"
    tools:remove="android:maxSdkVersion" />
```

This element tells Android that our app will need permission to write to external storage.

Finally, we need to actually request permission from the app. For this, we'll implement an interface called `ActivityCompat.OnRequestPermissionsResultCallback` like this:

```
public class MainActivity extends AppCompatActivity implements ActivityCompat.OnRequestPermissionsResultCallback {
```

Then, we can request permissions when the app loads by adding the following to `onCreate` :

```
requestPermissions(new String[]{Manifest.permission.WRITE_EXTERNAL_STORAGE}, 1);
```

```
requestPermissions(new String[]{Manifest.permission.WRITE_EXTERNAL_STORAGE}, 1);
```

This should pop-up a dialog that allows the user to allow or deny the permission. You can check the result of that dialog by adding the below method:

```
@Override
public void onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);
}
```

That's it for bookkeeping, so let's implement our save functionality now! Add a new `Button` to the layout, and use `android:onClick` to wire it up to a method in your `MainActivity`. Inside of that method, you'll want to get a `Bitmap` of the modified image, and then use `MediaStore.Images.Media.insertImage` to save the file.

To test, you can open up the `Photos` app in the emulator, and you should see filtered photos saved there.

How to Submit

To submit your code with `submit50`, you may either: (1) upload your code to CS50 IDE and run `submit50` from inside of your IDE, or (2) install `submit50` on your own computer by running `pip3 install submit50` (assuming you have [Python 3](https://www.python.org/downloads/) (<https://www.python.org/downloads/>) installed).

Execute the below, logging in with your GitHub username and password when prompted. For security, you'll see asterisks (`*`) instead of the actual characters in your password.

```
submit50 cs50/problems/2020/x/tracks/android/fiftygram
```