**Manifest File ( AndroidManifest.xml )**

* Manifest file defines the structure and metadata of our application, its components, and its requirements.
* the manifest file is required to declare the components of the app, permissions, The hardware and software features the app requires.

## **1. manifest**

* + The main component of the AndroidManifest.xml file is known as manifest.
  + It must contain an <application> element with the **xmlns:android** and **package** attribute specified.

<?xml version="1.0" encoding="utf-8"?>

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

package="com.example.geeksforgeeks">

<!-- manifest nodes -->

<application> </application>

</manifest>

## **2. uses-sdk**

* + It is used to define a minimum and maximum SDK version by means of an API Level integer that must be available on a device so that our application functions properly.
  + It is contained within the <manifest> element.

<uses-sdk

android:minSdkVersion="18"

android:targetSdkVersion="27" />

## **3. uses-permission**

Android apps must request permission to access sensitive user data (such as contacts and SMS) or certain system features (such as the camera and internet access).

<uses-permission

android:name="android.permission.CAMERA"

android:name="android.permission.SEND\_SMS" />

## **4. application**

* + A manifest can contain only one application node.
  + The application node also acts as a container for the Activity, Service, Content Provider, and Broadcast Receiver nodes that specify the application components.

The commonly used attributes are of this element are **title,** **icon,** **label,** **theme** etc.

**android:icon** represents the icon for all the android application components.

**android:label** works as the default label for all the application components.

**android:theme** represents a common theme for all the android activities.

<application

android:name=".GeeksForGeeks"

android:icon="@drawable/gfgIcon"

android:label="@string/app\_name"

android:roundIcon="@mipmap/ic\_launcher\_round"

android:theme="@android:style/Theme.Light"

tools:targetApi="31"

.

.

> </application>

## **6. activity**

* In the manifest file, all elements must be represented by <activity>. Any activity that is not declared there won’t run and won’t be visible to the system.
* It is contained within the <application> element.
* It has various characteristics, like label, name, theme, launchMode, and others.

<activity

android:name=".MainActivity"

android:exported="true">

</activity>

can contain:

[<intent-filter>](https://developer.android.com/guide/topics/manifest/intent-filter-element)  
[<meta-data>](https://developer.android.com/guide/topics/manifest/meta-data-element)  
[<layout>](https://developer.android.com/guide/topics/ui/multi-window#layout)

**7. BroadcastReciever**

<receiver android:name=".SampleBroadcastReceiver"

android:[**enabled**](https://developer.android.com/guide/topics/manifest/receiver-element#enabled)=["true" | "false"]  
          android:[**exported**](https://developer.android.com/guide/topics/manifest/receiver-element#exported)=["true" | "false"]

    . . .  
></receiver>

android:enabled

* Whether or not the broadcast receiver can be instantiated by the system — "true" if it can be, and "false" if not. The default value is "true".

android:exported

* "false" **:** the only messages the broadcast receiver can receive are those sent by

the system, components of the same application, or applications with

the same user ID.

* "true"  **:** the default value depends on whether the broadcast receiver contains

intent filters.

If the receiver contains at least one intent filter, then the default value

is "true". Otherwise, the default value is "false".

can contain:

[<intent-filter>](https://developer.android.com/guide/topics/manifest/intent-filter-element)  
[<meta-data>](https://developer.android.com/guide/topics/manifest/meta-data-element)

## **8. intent-filter**

* + It is the sub-element of activity that specifies the type of intent to which the activity, service, or broadcast receiver can send a response.
  + The intent filter must contain at least one **<action>** element.
  + If there are no **<action>** elements in an intent filter, the filter doesn't accept any [Intent](https://developer.android.com/reference/android/content/Intent) objects

<intent-filter>

<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

## **9. action**

It adds an action for the intent-filter. It is contained within the <intent-filter> element.

<action android:name="android.intent.action.MAIN" />

## **10. category**

It adds a category name to an intent-filter. It is contained within the <intent-filter> element.

<category android:name="android.intent.category.LAUNCHER" />