# INTEGRATING LOCATION SERVICES CHEATSHEET

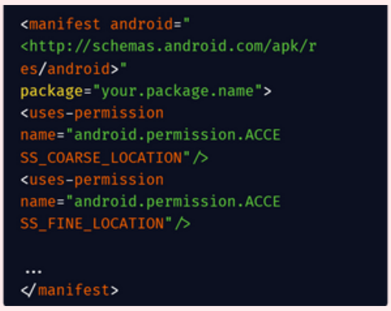
## Add Permissions in Android Manifest

### Permissions:

ACCESS_COARSE_LOCATION : Approximate location.

ACCESS_FINE_LOCATION : Precise location.

### Example:



```
<manifest android="
<http://schemas.android.com/apk/r
es/android>"
package="your.package.name">
<uses-permission
name="android.permission.ACCE
SS_COARSE_LOCATION"/>
<uses-permission
name="android.permission.ACCE
SS_FINE_LOCATION"/>

 ...
</manifest>
```

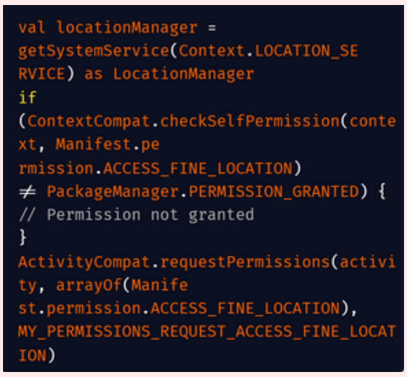
## Understanding Context

### Usage:

Access System Services: LOCATION_SERVICE .

Check Permissions: Use context to verify permissions.

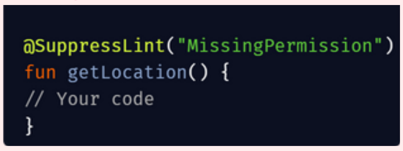
### Examples:



```
val locationManager =
getSystemService(Context.LOCATION_SE
RVICE) as LocationManager
if
(ContextCompat.checkSelfPermission(conte
xt, Manifest.pe
rmission.ACCESS_FINE_LOCATION)
!= PackageManager.PERMISSION_GRANTED) {
// Permission not granted
}
ActivityCompat.requestPermissions(activi
ty, arrayOf(Manife
st.permission.ACCESS_FINE_LOCATION),
MY_PERMISSIONS_REQUEST_ACCESS_FINE_LOCAT
ION)
```

## @SuppressLint

### Usage:

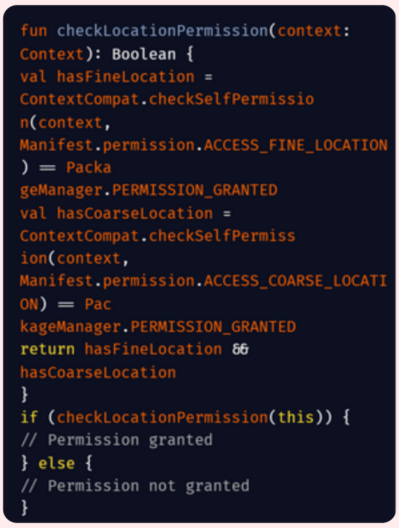


```
@SuppressLint("MissingPermission")
fun getLocation() {
// Your code
}
```

Note: Use cautiously to avoid hiding real issues.

## Check Location Permissions

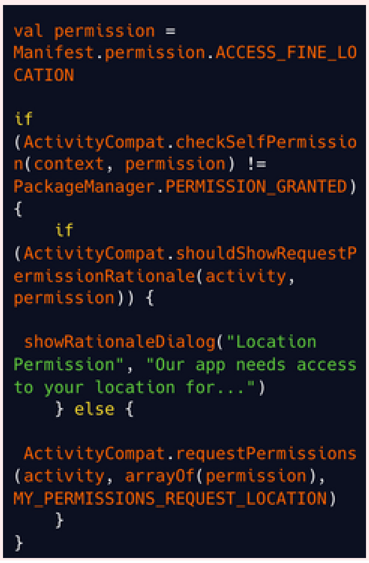
### Function:



```
fun checkLocationPermission(context:
Context): Boolean {
val hasFineLocation =
ContextCompat.checkSelfPermissio
n(context,
Manifest.permission.ACCESS_FINE_LOCATION
) == Packa
geManager.PERMISSION_GRANTED
val hasCoarseLocation =
ContextCompat.checkSelfPermiss
ion(context,
Manifest.permission.ACCESS_COARSE_LOCATI
ON) == Pac
kageManager.PERMISSION_GRANTED
return hasFineLocation &&
hasCoarseLocation
}
if (checkLocationPermission(this)) {
// Permission granted
} else {
// Permission not granted
}
```

## Explaining Permissions

### Example:



```
val permission =
Manifest.permission.ACCESS_FINE_LO
CATION

if
(ActivityCompat.checkSelfPermissio
n(context, permission) !=
PackageManager.PERMISSION_GRANTED)
{
    if
(ActivityCompat.shouldShowRequestP
ermissionRationale(activity,
permission)) {

 showRationaleDialog("Location
Permission", "Our app needs access
to your location for...")
    } else {

 ActivityCompat.requestPermissions
(activity, arrayOf(permission),
MY_PERMISSIONS_REQUEST_LOCATION)
    }
}
```

## Location Tracking Components

### Components:

LocalContext.current : Access current context.

FusedLocationProviderClient : Efficient location data provider.

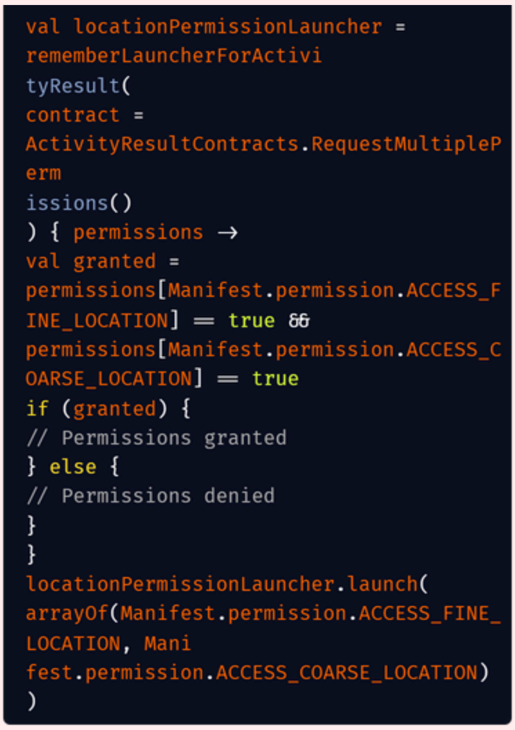
LocationCallback : Receive location updates.

LocationRequest.Builder : Define location update criteria.

Looper : Handle location updates sequentially.

## Handling Permission Requests

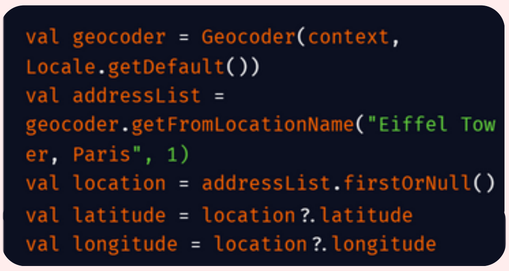
### Example:



```
val locationPermissionLauncher =
rememberLauncherForActivi
tyResult(
contract =
ActivityResultContracts.RequestMultipleP
erm
issions()
) { permissions ->
val granted =
permissions[Manifest.permission.ACCESS_F
INE_LOCATION] == true &&
permissions[Manifest.permission.ACCESS_C
OARSE_LOCATION] == true
if (granted) {
// Permissions granted
} else {
// Permissions denied
}
}
locationPermissionLauncher.launch(
arrayOf(Manifest.permission.ACCESS_FINE_
LOCATION, Mani
fest.permission.ACCESS_COARSE_LOCATION)
)
```

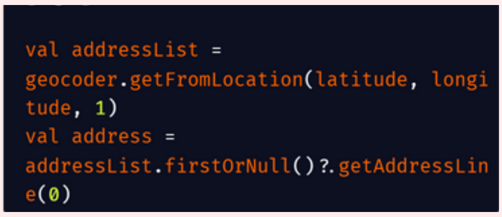
## Geocoder

### Geocoding:



```
val geocoder = Geocoder(context,
Locale.getDefault())
val addressList =
geocoder.getFromLocationName("Eiffel Tow
er, Paris", 1)
val location = addressList.firstOrNull()
val latitude = location?.latitude
val longitude = location?.longitude
```

### Reverse Geocoding:



```
val addressList =
geocoder.getFromLocation(latitude, longi
tude, 1)
val address =
addressList.firstOrNull()?.getAddressLin
e(0)
```