added in API level 1 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)
Summary: Constants (#constants) | Methods (#pubmethods) | Inherited Methods (#inhmethods) | [Expand All] (#)

LocationManager

```
public class LocationManager
extends Object (https://developer.android.com/reference/java/lang/Object.html)
```

java.lang.Object (https://developer.android.com/reference/java/lang/Object.html)
↳ android.location.LocationManager

This class provides access to the system location services. These services allow applications to obtain periodic updates of the device's geographical location, or to fire an application-specified `Intent` (https://developer.android.com/reference/android/content/Intent.html) when the device enters the proximity of a given geographical location.

> Unless noted, all Location API methods require the `ACCESS_COARSE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_COARSE_LOCATION) or `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permissions. If your application only has the coarse permission then it will not have access to the GPS or passive location providers. Other providers will still return location results, but the update rate will be throttled and the exact location will be obfuscated to a coarse level of accuracy.

Instances of this class must be obtained using `Context.getSystemService(Class)` (https://developer.android.com/reference/android/content/Context.html#getSystemService(java.lang.Class<T>)) with the argument `LocationManager.class` or `Context.getSystemService(String)` (https://developer.android.com/reference/android/content/Context.html#getSystemService(java.lang.String)) with the argument `Context.LOCATION_SERVICE` (https://developer.android.com/reference/android/content/Context.html#LOCATION_SERVICE).

Summary

Constants	
String (https://developer.android.com/reference/java/lang/String.html)	GNSS_HARDWARE_MODEL_NAME_UNKNOWN (https://developer.android.com/reference/android/location/LocationManager.html#GNSS_HARDWARE_MODEL_NAME_UNKNOWN) The value returned by getGnssHardwareModelName() (https://developer.android.com/reference/android/location/LocationManager.html#getGnssHardwareModelName()) when the hardware does not support providing the actual value.
String (https://developer.android.com/reference/java/lang/String.html)	GPS_PROVIDER (https://developer.android.com/reference/android/location/LocationManager.html#GPS_PROVIDER) Name of the GPS location provider.
String (https://developer.android.com/reference/java/lang/String.html)	KEY_LOCATION_CHANGED (https://developer.android.com/reference/android/location/LocationManager.html#KEY_LOCATION_CHANGED) Key used for a Bundle extra holding a Location value when a location change is broadcast using a PendingIntent.
String (https://developer.android.com/reference/java/lang/String.html)	KEY_PROVIDER_ENABLED (https://developer.android.com/reference/android/location/LocationManager.html#KEY_PROVIDER_ENABLED) Key used for a Bundle extra holding an Boolean status value when a provider enabled/disabled event is broadcast using a PendingIntent.
String (https://developer.android.com/reference/java/lang/String.html)	KEY_PROXIMITY_ENTERING (https://developer.android.com/reference/android/location/LocationManager.html#KEY_PROXIMITY_ENTERING) Key used for the Bundle extra holding a boolean indicating whether a proximity alert is entering (true) or exiting (false)..
String (https://developer.android.com/reference/java/lang/String.html)	KEY_STATUS_CHANGED (https://developer.android.com/reference/android/location/LocationManager.html#KEY_STATUS_CHANGED) Key used for a Bundle extra holding an Integer status value when a status change is broadcast using a PendingIntent.
String (https://developer.android.com/reference/java/lang	MODE_CHANGED_ACTION (https://developer.android.com/reference/android/location/LocationManager.html#MODE_CHANGED_ACTION)

	(https://developer.android.com/reference/android/provider/Settings.Secure.html#LOCATION_MODE) changes.
String (https://developer.android.com/reference/java/lang/String.html)	NETWORK_PROVIDER (https://developer.android.com/reference/android/location/LocationManager.html#NETWORK_PROVIDER) Name of the network location provider.
String (https://developer.android.com/reference/java/lang/String.html)	PASSIVE_PROVIDER (https://developer.android.com/reference/android/location/LocationManager.html#PASSIVE_PROVIDER) A special location provider for receiving locations without actually initiating a location fix.
String (https://developer.android.com/reference/java/lang/String.html)	PROVIDERS_CHANGED_ACTION (https://developer.android.com/reference/android/location/LocationManager.html#PROVIDERS_CHANGED_ACTION) Broadcast intent action when the configured location providers change.

Public methods	
boolean	addGpsStatusListener (https://developer.android.com/reference/LocationManager.html#addGpsStatusListener(android.location.GpsSta(https://developer.android.com/reference/android/location/GpsStatu *This method was deprecated in API level 24. use* registerGnssSt(https://developer.android.com/reference/android/location/LocationManager.html#registerGnssStatusCallback(android.location.
boolean	addNmeaListener (https://developer.android.com/reference/andro/LocationManager.html#addNmeaListener(android.location.OnNmeaMessa(OnNmeaMessageListener (https://developer.android.com/referer/OnNmeaMessageListener.html) listener, Handler (https://develo/os/Handler.html) handler) Adds an NMEA listener.
boolean	addNmeaListener (https://developer.android.com/reference/andro/LocationManager.html#addNmeaListener(android.location.OnNmeaMessa(https://developer.android.com/reference/android/location/OnNmeaMe Adds an NMEA listener.
boolean	addNmeaListener (https://developer.android.com/reference/andro/LocationManager.html#addNmeaListener(android.location.GpsStatus.N

	This method was deprecated in API level 24. use addNmeaListene (https://developer.android.com/reference/android/location /LocationManager.html#addNmeaListener(android.location.OnNmeaMessa
void	addProximityAlert (https://developer.android.com/reference/and /LocationManager.html#addProximityAlert(double, double, float, lon latitude, double longitude, float radius, long expira (https://developer.android.com/reference/android/app/PendingIntent Set a proximity alert for the location given by the position (latitude
void	addTestProvider (https://developer.android.com/reference/andro /LocationManager.html#addTestProvider(java.lang.String, boolean, b boolean, int, int))(String (https://developer.android.com/refere requiresNetwork, boolean requiresSatellite, boolean r hasMonetaryCost, boolean supportsAltitude, boolean su supportsBearing, int powerRequirement, int accuracy) Creates a mock location provider and adds it to the set of active p
void	clearTestProviderEnabled (https://developer.android.com/refe /LocationManager.html#clearTestProviderEnabled(java.lang.String))(/reference/java/lang/String.html) provider) Removes any mock enabled value associated with the given provi
void	clearTestProviderLocation (https://developer.android.com/ref /LocationManager.html#clearTestProviderLocation(java.lang.String)) /reference/java/lang/String.html) provider) Removes any mock location associated with the given provider.
void	clearTestProviderStatus (https://developer.android.com/refer /LocationManager.html#clearTestProviderStatus(java.lang.String))(S /reference/java/lang/String.html) provider) Removes any mock status values associated with the given provi
List (https://developer.android.com /reference/java/util /List.html)<String (https://developer.android.com /reference/java/lang /String.html)>	getAllProviders (https://developer.android.com/reference/andro /LocationManager.html#getAllProviders())() Returns a list of the names of all known location providers.
String	getBestProvider (https://developer.android.com/reference/andro

/reference/java/lang /String.html)	(https://developer.android.com/reference/android/location/Criteria Returns the name of the provider that best meets the given criteri
String (https://developer.android.com /reference/java/lang /String.html)	getGnssHardwareModelName (https://developer.android.com/refe /LocationManager.html#getGnssHardwareModelName())() Returns the Model Name (including Vendor and Hardware/Softwa
int	getGnssYearOfHardware (https://developer.android.com/referenc /LocationManager.html#getGnssYearOfHardware())() Returns the model year of the GNSS hardware and software build
GpsStatus (https://developer.android.com /reference/android/location /GpsStatus.html)	getGpsStatus (https://developer.android.com/reference/android/l /LocationManager.html#getGpsStatus(android.location.GpsStatus))(Gp /reference/android/location/GpsStatus.html) status) Retrieves information about the current status of the GPS engine.
Location (https://developer.android.com /reference/android/location /Location.html)	getLastKnownLocation (https://developer.android.com/reference /LocationManager.html#getLastKnownLocation(java.lang.String))(Str /java/lang/String.html) provider) Returns a Location indicating the data from the last known locatic
LocationProvider (https://developer.android.com /reference/android/location /LocationProvider.html)	getProvider (https://developer.android.com/reference/android/lo /LocationManager.html#getProvider(java.lang.String))(String (htt /java/lang/String.html) name) Returns the information associated with the location provider of t by that name.
List (https://developer.android.com /reference/java/util /List.html)<String (https://developer.android.com /reference/java/lang /String.html)>	getProviders (https://developer.android.com/reference/android/l /LocationManager.html#getProviders(boolean))(boolean enabledOr Returns a list of the names of location providers.
List (https://developer.android.com /reference/java/util /List.html)<String (https://developer.android.com /reference/java/lang	getProviders (https://developer.android.com/reference/android/l /LocationManager.html#getProviders(android.location.Criteria, bool (https://developer.android.com/reference/android/location/Criteria Returns a list of the names of LocationProviders that satisfy the g

boolean	isLocationEnabled (https://developer.android.com/reference/and /LocationManager.html#isLocationEnabled())() Returns the current enabled/disabled status of location
boolean	isProviderEnabled (https://developer.android.com/reference/and /LocationManager.html#isProviderEnabled(java.lang.String))(String /java/lang/String.html) provider) Returns the current enabled/disabled status of the given provider.
boolean	registerGnssMeasurementsCallback (https://developer.androi /LocationManager.html#registerGnssMeasurementsCallback(android.loc (GnssMeasurementsEvent.Callback (https://developer.android /GnssMeasurementsEvent.Callback.html) callback) Registers a GPS Measurement callback.
boolean	registerGnssMeasurementsCallback (https://developer.androi /LocationManager.html#registerGnssMeasurementsCallback(android.loc android.os.Handler))(GnssMeasurementsEvent.Callback (https /location/GnssMeasurementsEvent.Callback.html) callback, Handl /android/os/Handler.html) handler) Registers a GPS Measurement callback.
boolean	registerGnssNavigationMessageCallback (https://developer. /LocationManager.html#registerGnssNavigationMessageCallback(androi android.os.Handler))(GnssNavigationMessage.Callback (https /location/GnssNavigationMessage.Callback.html) callback, Handl /android/os/Handler.html) handler) Registers a GNSS Navigation Message callback.
boolean	registerGnssNavigationMessageCallback (https://developer. /LocationManager.html#registerGnssNavigationMessageCallback(androi (GnssNavigationMessage.Callback (https://developer.android /GnssNavigationMessage.Callback.html) callback) Registers a GNSS Navigation Message callback.
boolean	registerGnssStatusCallback (https://developer.android.com/re /LocationManager.html#registerGnssStatusCallback(android.location. (GnssStatus.Callback (https://developer.android.com/reference callback) Registers a GNSS status callback.
boolean	registerGnssStatusCallback (https://developer.android.com/re

	(GnssStatus.Callback (https://developer.android.com/reference callback, Handler (https://developer.android.com/reference/and Registers a GNSS status callback.
void	removeGpsStatusListener (https://developer.android.com/refer /LocationManager.html#removeGpsStatusListener(android.location.Gps (https://developer.android.com/reference/android/location/GpsStatu *This method was deprecated in API level 24. use* unregisterGnss (https://developer.android.com/reference/android/location /LocationManager.html#unregisterGnssStatusCallback(android.locatio
void	removeNmeaListener (https://developer.android.com/reference/a /LocationManager.html#removeNmeaListener(android.location.OnNmeaMe (https://developer.android.com/reference/android/location/OnNmeaMe Removes an NMEA listener.
void	removeNmeaListener (https://developer.android.com/reference/a /LocationManager.html#removeNmeaListener(android.location.GpsStatu (https://developer.android.com/reference/android/location/GpsStatu *This method was deprecated in API level 24. use* removeNmeaList (https://developer.android.com/reference/android/location /LocationManager.html#removeNmeaListener(android.location.OnNmeaMe
void	removeProximityAlert (https://developer.android.com/reference /LocationManager.html#removeProximityAlert(android.app.PendingInte (https://developer.android.com/reference/android/app/PendingIntent Removes the proximity alert with the given PendingIntent.
void	removeTestProvider (https://developer.android.com/reference/a /LocationManager.html#removeTestProvider(java.lang.String))(Strir /java/lang/String.html) provider) Removes the mock location provider with the given name.
void	removeUpdates (https://developer.android.com/reference/android/ /LocationManager.html#removeUpdates(android.location.LocationListe (https://developer.android.com/reference/android/location/Location Removes all location updates for the specified LocationListener.
void	removeUpdates (https://developer.android.com/reference/android/ /LocationManager.html#removeUpdates(android.app.PendingIntent))(Pe (https://developer.android.com/reference/android/app/PendingIntent Removes all location updates for the specified pending intent.

void	requestLocationUpdates (https://developer.android.com/referer /LocationManager.html#requestLocationUpdates(java.lang.String, lon (String (https://developer.android.com/reference/java/lang/Strin minDistance, LocationListener (https://developer.android.co /LocationListener.html) listener) Register for location updates using the named provider, and a per
void	requestLocationUpdates (https://developer.android.com/referer /LocationManager.html#requestLocationUpdates(long, float, android. android.location.LocationListener, android.os.Looper))(long minT (https://developer.android.com/reference/android/location/Criteria (https://developer.android.com/reference/android/location/Location (https://developer.android.com/reference/android/os/Looper.html) l Register for location updates using a Criteria, and a callback on th
void	requestLocationUpdates (https://developer.android.com/referer /LocationManager.html#requestLocationUpdates(java.lang.String, lon android.os.Looper))(String (https://developer.android.com/refere minTime, float minDistance, LocationListener (https://d /location/LocationListener.html) listener, Looper (https://dev /os/Looper.html) looper) Register for location updates using the named provider, and a call
void	requestLocationUpdates (https://developer.android.com/referer /LocationManager.html#requestLocationUpdates(long, float, android. android.app.PendingIntent))(long minTime, float minDistanc /reference/android/location/Criteria.html) criteria, PendingI /reference/android/app/PendingIntent.html) intent) Register for location updates using a Criteria and pending intent.
void	requestLocationUpdates (https://developer.android.com/referer /LocationManager.html#requestLocationUpdates(java.lang.String, lon (String (https://developer.android.com/reference/java/lang/Strin minDistance, PendingIntent (https://developer.android.com/re intent) Register for location updates using the named provider, and a per
void	requestSingleUpdate (https://developer.android.com/reference/ /LocationManager.html#requestSingleUpdate(java.lang.String, androi (https://developer.android.com/reference/java/lang/String.html) pI (https://developer.android.com/reference/android/app/PendingIntent

void	requestSingleUpdate (https://developer.android.com/reference/a /LocationManager.html#requestSingleUpdate(android.location.Criteri (https://developer.android.com/reference/android/location/Criteria (https://developer.android.com/reference/android/app/PendingIntent Register for a single location update using a Criteria and pending
void	requestSingleUpdate (https://developer.android.com/reference/a /LocationManager.html#requestSingleUpdate(java.lang.String, androi android.os.Looper))(String (https://developer.android.com/refere LocationListener (https://developer.android.com/reference/andr listener, Looper (https://developer.android.com/reference/andr Register for a single location update using the named provider an
void	requestSingleUpdate (https://developer.android.com/reference/a /LocationManager.html#requestSingleUpdate(android.location.Criteri android.os.Looper))(Criteria (https://developer.android.com/refe criteria, LocationListener (https://developer.android.com/re /LocationListener.html) listener, Looper (https://developer.an looper) Register for a single location update using a Criteria and a callbac
boolean	sendExtraCommand (https://developer.android.com/reference/andr /LocationManager.html#sendExtraCommand(java.lang.String, java.lang (https://developer.android.com/reference/java/lang/String.html) pr (https://developer.android.com/reference/java/lang/String.html) co (https://developer.android.com/reference/android/os/Bundle.html) e Sends additional commands to a location provider.
void	setTestProviderEnabled (https://developer.android.com/referer /LocationManager.html#setTestProviderEnabled(java.lang.String, boo (https://developer.android.com/reference/java/lang/String.html) pr Sets a mock enabled value for the given provider.
void	setTestProviderLocation (https://developer.android.com/refer /LocationManager.html#setTestProviderLocation(java.lang.String, an (https://developer.android.com/reference/java/lang/String.html) pr (https://developer.android.com/reference/android/location/Location Sets a mock location for the given provider.
void	setTestProviderStatus (https://developer.android.com/referenc /LocationManager.html#setTestProviderStatus(java.lang.String, int,

	(https://developer.android.com/reference/android/os/Bundle.html) e Sets mock status values for the given provider.
void	unregisterGnssMeasurementsCallback (https://developer.andr /LocationManager.html#unregisterGnssMeasurementsCallback(android.l (GnssMeasurementsEvent.Callback (https://developer.android /GnssMeasurementsEvent.Callback.html) callback) Unregisters a GPS Measurement callback.
void	unregisterGnssNavigationMessageCallback (https://develop /LocationManager.html#unregisterGnssNavigationMessageCallback(andr (GnssNavigationMessage.Callback (https://developer.android /GnssNavigationMessage.Callback.html) callback) Unregisters a GNSS Navigation Message callback.
void	unregisterGnssStatusCallback (https://developer.android.com /LocationManager.html#unregisterGnssStatusCallback(android.locatio (GnssStatus.Callback (https://developer.android.com/reference callback) Removes a GNSS status callback.

Inherited methods

(#)From class java.lang.Object (https://developer.android.com/reference/java/lang/Object.html)

Constants

GNSS_HARDWARE_MODEL_NAME_UNKNOWN

Android P Developer Preview (https://developer.android.com/preview/)

String (https://developer.android.com/reference/java/lang/String.html) GNSS_HARDWARE_MODEL_NAME

The value returned by getGnssHardwareModelName() (https://developer.android.com/reference/android/location/LocationManager.html#getGnssHardwareModelName()) when the hardware does not support providing the actual value.

Constant Value: "Model Name Unknown"

GPS_PROVIDER added in API level 1 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`String (https://developer.android.com/reference/java/lang/String.html) GPS_PROVIDER`

Name of the GPS location provider.

This provider determines location using satellites. Depending on conditions, this provider may take a while to return a location fix. Requires the permission `ACCESS_FINE_LOCATION (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION)`.

The extras Bundle for the GPS location provider can contain the following key/value pairs:

- satellites - the number of satellites used to derive the fix

Constant Value: "gps"

KEY_LOCATION_CHANGED added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`String (https://developer.android.com/reference/java/lang/String.html) KEY_LOCATION_CHANGED`

Key used for a Bundle extra holding a Location value when a location change is broadcast using a PendingIntent.

Constant Value: "location"

KEY_PROVIDER_ENABLED added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`String (https://developer.android.com/reference/java/lang/String.html) KEY_PROVIDER_ENABLED`

Key used for a Bundle extra holding an Boolean status value when a provider enabled/disabled event is broadcast using a PendingIntent.

Constant Value: "providerEnabled"

KEY_PROXIMITY_ENTERING added in API level 1 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`String (https://developer.android.com/reference/java/lang/String.html) KEY_PROXIMITY_ENTERING`

Key used for the Bundle extra holding a boolean indicating whether a proximity alert is entering (true) or exiting (false)..

Constant Value: "entering"

KEY_STATUS_CHANGED

added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`String` (https://developer.android.com/reference/java/lang/String.html) KEY_STATUS_CHANGED

Key used for a Bundle extra holding an Integer status value when a status change is broadcast using a PendingIntent.

Constant Value: "status"

MODE_CHANGED_ACTION

added in API level 19 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`String` (https://developer.android.com/reference/java/lang/String.html) MODE_CHANGED_ACTION

Broadcast intent action when `LOCATION_MODE` (https://developer.android.com/reference/android/provider/Settings.Secure.html#LOCATION_MODE) changes. For use with the `LOCATION_MODE` (https://developer.android.com/reference/android/provider/Settings.Secure.html#LOCATION_MODE) API. If you're interacting with `isProviderEnabled(String)` (https://developer.android.com/reference/android/location/LocationManager.html#isProviderEnabled(java.lang.String)), use `PROVIDERS_CHANGED_ACTION` (https://developer.android.com/reference/android/location/LocationManager.html#PROVIDERS_CHANGED_ACTION) instead. In the future, there may be mode changes that do not result in `PROVIDERS_CHANGED_ACTION` (https://developer.android.com/reference/android/location/LocationManager.html#PROVIDERS_CHANGED_ACTION) broadcasts.

Constant Value: "android.location.MODE_CHANGED"

NETWORK_PROVIDER

added in API level 1 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`String` (https://developer.android.com/reference/java/lang/String.html) NETWORK_PROVIDER

Name of the network location provider.

are retrieved by means of a network lookup.

Constant Value: "network"

PASSIVE_PROVIDER

added in API level 8 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`String` (https://developer.android.com/reference/java/lang/String.html) `PASSIVE_PROVIDER`

A special location provider for receiving locations without actually initiating a location fix.

This provider can be used to passively receive location updates when other applications or services request them without actually requesting the locations yourself. This provider will return locations generated by other providers. You can query the `getProvider()` (https://developer.android.com/reference/android/location/Location.html#getProvider()) method to determine the origin of the location update. Requires the permission `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION), although if the GPS is not enabled this provider might only return coarse fixes.

Constant Value: "passive"

PROVIDERS_CHANGED_ACTION

added in API level 9 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`String` (https://developer.android.com/reference/java/lang/String.html) `PROVIDERS_CHANGED_ACTIO`

Broadcast intent action when the configured location providers change. For use with `isProviderEnabled(String)` (https://developer.android.com/reference/android/location/LocationManager.html#isProviderEnabled(java.lang.String)). If you're interacting with the `LOCATION_MODE` (https://developer.android.com/reference/android/provider/Settings.Secure.html#LOCATION_MODE) API, use `MODE_CHANGED_ACTION` (https://developer.android.com/reference/android/location/LocationManager.html#MODE_CHANGED_ACTION) instead.

Constant Value: "android.location.PROVIDERS_CHANGED"

Public methods

addGpsStatusListener

Added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
boolean addGpsStatusListener (GpsStatus.Listener (https://developer.android.com/reference/
```

> **This method was deprecated in API level 24.**
> use `registerGnssStatusCallback(GnssStatus.Callback)` (https://developer.android.com/reference/android/location/LocationManager.html#registerGnssStatusCallback(android.location.GnssStatus.Callback)) instead.

Adds a GPS status listener.

Requires the `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permission.

Parameters	
`listener`	`GpsStatus.Listener`: GPS status listener object to register

Returns	
`boolean`	true if the listener was successfully added

Throws	
`SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html)	if the ACCESS_FINE_LOCATION permission is not present

addNmeaListener

Added in API level 24 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
boolean addNmeaListener (OnNmeaMessageListener (https://developer.android.com/reference/and
                Handler (https://developer.android.com/reference/android/os/Handler.html) handl
```

Adds an NMEA listener.

Requires the `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permission.

Parameters	
`listener`	`OnNmeaMessageListener`: a `OnNmeaMessageListener`

	(https://developer.android.com/reference/android/location/OnNmeaMessageListener.html) object to register
handler	Handler: the handler that the listener runs on.

Returns	
boolean	true if the listener was successfully added

Throws	
SecurityException (https://developer.android.com/reference/java/lang/SecurityException.html)	if the ACCESS_FINE_LOCATION permission is not present

addNmeaListener

Added in API level 24 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

boolean addNmeaListener (OnNmeaMessageListener (https://developer.android.com/reference/and

Adds an NMEA listener.

Requires the ACCESS_FINE_LOCATION (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permission.

Parameters	
listener	OnNmeaMessageListener: a OnNmeaMessageListener (https://developer.android.com/reference/android/location/OnNmeaMessageListener.html) object to register

Returns	
boolean	true if the listener was successfully added

Throws	
SecurityException (https://developer.android.com/reference/java/lang/SecurityException.html)	if the ACCESS_FINE_LOCATION permission is not present

addNmeaListener

added in API level 5 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
boolean addNmeaListener (GpsStatus.NmeaListener (https://developer.android.com/reference/a
```

> **This method was deprecated in API level 24.**
> use `addNmeaListener(OnNmeaMessageListener)` (https://developer.android.com/reference/android/location/LocationManager.html#addNmeaListener(android.location.OnNmeaMessageListener)) instead.

Adds an NMEA listener.

Requires the `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permission.

Parameters	
`listener`	`GpsStatus.NmeaListener`: a `GpsStatus.NmeaListener` (https://developer.android.com/reference/android/location/GpsStatus.NmeaListener.html) object to register

Returns	
`boolean`	true if the listener was successfully added

Throws	
`SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html)	if the ACCESS_FINE_LOCATION permission is not present

addProximityAlert

added in API level 1 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void addProximityAlert (double latitude,
                double longitude,
                float radius,
                long expiration,
                PendingIntent (https://developer.android.com/reference/android/app/PendingInten
```

Set a proximity alert for the location given by the position (latitude, longitude) and the given radius.

The fired Intent will have a boolean extra added with key `KEY_PROXIMITY_ENTERING` (https://developer.android.com/reference/android/location/LocationManager.html#KEY_PROXIMITY_ENTERING). If the value is true, the device is entering the proximity region; if false, it is exiting.

Due to the approximate nature of position estimation, if the device passes through the given area briefly, it is possible that no Intent will be fired. Similarly, an Intent could be fired if the device passes very close to the given area but does not actually enter it.

After the number of milliseconds given by the expiration parameter, the location manager will delete this proximity alert and no longer monitor it. A value of -1 indicates that there should be no expiration time.

Internally, this method uses both `NETWORK_PROVIDER` (https://developer.android.com/reference/android/location/LocationManager.html#NETWORK_PROVIDER) and `GPS_PROVIDER` (https://developer.android.com/reference/android/location/LocationManager.html#GPS_PROVIDER).

Before API version 17, this method could be used with `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) or `ACCESS_COARSE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_COARSE_LOCATION). From API version 17 and onwards, this method requires `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permission.

Requires the `ACCESS_COARSE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_COARSE_LOCATION) or `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permissions.

Parameters	
`latitude`	`double`: the latitude of the central point of the alert region
`longitude`	`double`: the longitude of the central point of the alert region
`radius`	`float`: the radius of the central point of the alert region, in meters
`expiration`	`long`: time for this proximity alert, in milliseconds, or -1 to indicate no expiration
`intent`	`PendingIntent`: a PendingIntent that will be used to generate an Intent to fire when entry to or exit from the alert region is detected

Throws

SecurityException (https://developer.android.com/reference/java/lang/SecurityException.html)	if ACCESS_FINE_LOCATION (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permission is not present

addTestProvider added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void addTestProvider (String (https://developer.android.com/reference/java/lang/String.html) na
                boolean requiresNetwork,
                boolean requiresSatellite,
                boolean requiresCell,
                boolean hasMonetaryCost,
                boolean supportsAltitude,
                boolean supportsSpeed,
                boolean supportsBearing,
                int powerRequirement,
                int accuracy)
```

Creates a mock location provider and adds it to the set of active providers.

Parameters	
name	String: the provider name
requiresNetwork	boolean
requiresSatellite	boolean
requiresCell	boolean
hasMonetaryCost	boolean
supportsAltitude	boolean
supportsSpeed	boolean
supportsBearing	boolean
powerRequirement	int
accuracy	int

Throws	
SecurityException	if mock location app op (https://developer.android.com

/reference/java/lang /SecurityException.html)	is not set to allowed (https://developer.android.com/reference /android/app/AppOpsManager.html#MODE_ALLOWED) for your app.
IllegalArgumentException (https://developer.android.com /reference/java/lang /IllegalArgumentException.html)	if a provider with the given name already exists

clearTestProviderEnabled

added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

void clearTestProviderEnabled (String (https://developer.android.com/reference/java/lang/Stri

Removes any mock enabled value associated with the given provider.

Parameters	
provider	String: the provider name

Throws	
SecurityException (https://developer.android.com /reference/java/lang /SecurityException.html)	if mock location app op (https://developer.android.com /reference/android/app/AppOpsManager.html#OPSTR_MOCK_LOCATION) is not set to allowed (https://developer.android.com/reference /android/app/AppOpsManager.html#MODE_ALLOWED) for your app.
IllegalArgumentException (https://developer.android.com /reference/java/lang /IllegalArgumentException.html)	if no provider with the given name exists

clearTestProviderLocation

added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

void clearTestProviderLocation (String (https://developer.android.com/reference/java/lang/Str

Removes any mock location associated with the given provider.

Parameters	

Throws	
`SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html)	if `mock location app op` (https://developer.android.com/reference/android/app/AppOpsManager.html#OPSTR_MOCK_LOCATION) is not set to `allowed` (https://developer.android.com/reference/android/app/AppOpsManager.html#MODE_ALLOWED) for your app.
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if no provider with the given name exists

clearTestProviderStatus

added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`void clearTestProviderStatus (String` (https://developer.android.com/reference/java/lang/Strin

Removes any mock status values associated with the given provider.

Parameters	
`provider`	`String`: the provider name

Throws	
`SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html)	if `mock location app op` (https://developer.android.com/reference/android/app/AppOpsManager.html#OPSTR_MOCK_LOCATION) is not set to `allowed` (https://developer.android.com/reference/android/app/AppOpsManager.html#MODE_ALLOWED) for your app.
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if no provider with the given name exists

getAllProviders

added in API level 1 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`List` (https://developer.android.com/reference/java/util/List.html)`<String` (https://developer.androi

All providers are returned, including ones that are not permitted to be accessed by the calling activity or are currently disabled.

Returns	
`List` (https://developer.android.com/reference/java/util/List.html)`<String` (https://developer.android.com/reference/java/lang/String.html)`>`	list of Strings containing names of the provider

getBestProvider added in API level 1 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
String (https://developer.android.com/reference/java/lang/String.html) getBestProvider (Criter
                boolean enabledOnly)
```

Returns the name of the provider that best meets the given criteria. Only providers that are permitted to be accessed by the calling activity will be returned. If several providers meet the criteria, the one with the best accuracy is returned. If no provider meets the criteria, the criteria are loosened in the following sequence:

- power requirement
- accuracy
- bearing
- speed
- altitude

Note that the requirement on monetary cost is not removed in this process.

Parameters	
`criteria`	`Criteria`: the criteria that need to be matched
`enabledOnly`	`boolean`: if true then only a provider that is currently enabled is returned

Returns

String (https://developer.android.com/reference/java/lang/String.html)	name of the provider that best matches the requirements

getGnssHardwareModelName Android P Developer Preview (https://developer.android.com/preview/)

String (https://developer.android.com/reference/java/lang/String.html) getGnssHardwareModelName

Returns the Model Name (including Vendor and Hardware/Software Version) of the GNSS hardware driver. Will return GNSS_HARDWARE_MODEL_NAME_UNKNOWN (https://developer.android.com/reference/android/location/LocationManager.html#GNSS_HARDWARE_MODEL_NAME_UNKNOWN) when the GNSS hardware abstraction layer does not support providing this value.

Returns	
String (https://developer.android.com/reference/java/lang/String.html)	This value will never be null.

getGnssYearOfHardware Android P Developer Preview (https://developer.android.com/preview/)

int getGnssYearOfHardware ()

Returns the model year of the GNSS hardware and software build. May return 0 if the model year is less than 2016.

Returns	
int	

getGpsStatus added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

Retrieves information about the current status of the GPS engine. This should only be called from the `onGpsStatusChanged(int)` (https://developer.android.com/reference/android/location/GpsStatus.Listener.html#onGpsStatusChanged(int)) callback to ensure that the data is copied atomically. The caller may either pass in a `GpsStatus` (https://developer.android.com/reference/android/location/GpsStatus.html) object to set with the latest status information, or pass null to create a new `GpsStatus` (https://developer.android.com/reference/android/location/GpsStatus.html) object.

Requires the `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permission.

Parameters	
`status`	`GpsStatus`: object containing GPS status details, or null.

Returns	
`GpsStatus` (https://developer.android.com/reference/android/location/GpsStatus.html)	status object containing updated GPS status.

getLastKnownLocation

added in API level 1 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`Location` (https://developer.android.com/reference/android/location/Location.html) `getLastKnownLo`

Returns a Location indicating the data from the last known location fix obtained from the given provider.

This can be done without starting the provider. Note that this location could be out-of-date, for example if the device was turned off and moved to another location.

If the provider is currently disabled, null is returned.

Requires the `ACCESS_COARSE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_COARSE_LOCATION) or `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permissions.

Parameters

Returns	
`Location` (https://developer.android.com/reference/android/location/Location.html)	the last known location for the provider, or null

Throws	
`SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html)	if no suitable permission is present
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if provider is null or doesn't exist

getProvider

added in API level 1 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`LocationProvider` (https://developer.android.com/reference/android/location/LocationProvider.html)

Returns the information associated with the location provider of the given name, or null if no provider exists by that name.

Parameters	
`name`	`String`: the provider name

Returns	
`LocationProvider` (https://developer.android.com/reference/android/location/LocationProvider.html)	a LocationProvider, or null

Throws	
`IllegalArgumentException`	if name is null or does not exist

/reference/java/lang /IllegalArgumentException.html)	
SecurityException (https://developer.android.com /reference/java/lang /SecurityException.html)	if the caller is not permitted to access the given provider.

getProviders

added in API level 1 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

List (https://developer.android.com/reference/java/util/List.html)<String (https://developer.androi

Returns a list of the names of location providers.

Parameters	
enabledOnly	boolean: if true then only the providers which are currently enabled are returned.

Returns	
List (https://developer.android.com /reference/java/util /List.html)<String (https://developer.android.com /reference/java/lang /String.html)>	list of Strings containing names of the providers

getProviders

added in API level 1 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

List (https://developer.android.com/reference/java/util/List.html)<String (https://developer.androi
boolean enabledOnly)

Returns a list of the names of LocationProviders that satisfy the given criteria, or null if none do. Only providers that are permitted to be accessed by the calling activity will be returned.

`criteria`	`Criteria`: the criteria that the returned providers must match
`enabledOnly`	`boolean`: if true then only the providers which are currently enabled are returned.

Returns	
`List` (https://developer.android.com/reference/java/util/List.html)`<String` (https://developer.android.com/reference/java/lang/String.html)`>`	list of Strings containing names of the providers

isLocationEnabled

Android P Developer Preview (https://developer.android.com/preview/)

`boolean isLocationEnabled ()`

Returns the current enabled/disabled status of location

Returns	
`boolean`	true if location is enabled. false if location is disabled.

isProviderEnabled

added in API level 1 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`boolean isProviderEnabled (String` (https://developer.android.com/reference/java/lang/String.ht

Returns the current enabled/disabled status of the given provider.

If the user has enabled this provider in the Settings menu, true is returned otherwise false is returned

Callers should instead use `isLocationEnabled()` (https://developer.android.com/reference/android/location/LocationManager.html#isLocationEnabled()) unless they depend on provider-specific APIs such as `requestLocationUpdates(String, long, float, LocationListener)` (https://developer.android.com/reference/android/location

Before API version `LOLLIPOP` (https://developer.android.com/reference/android/os/Build.VERSION_CODES.html#LOLLIPOP), this method would throw `SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html) if the location permissions were not sufficient to use the specified provider.

Parameters	
`provider`	`String`: the name of the provider

Returns	
`boolean`	true if the provider exists and is enabled

Throws	
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if provider is null

registerGnssMeasurementsCallback

added in API level 24 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`boolean registerGnssMeasurementsCallback (GnssMeasurementsEvent.Callback (https://de`

Registers a GPS Measurement callback.

Requires the `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permission.

Parameters	
`callback`	`GnssMeasurementsEvent.Callback`: a `GnssMeasurementsEvent.Callback` (https://developer.android.com/reference/android/location/GnssMeasurementsEvent.Callback.html) object to register.

Returns	
`boolean`	`true` if the callback was added successfully, `false` otherwise.

registerGnssMeasurementsCallback

added in API level 24 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
boolean registerGnssMeasurementsCallback (GnssMeasurementsEvent.Callback (https://de
                Handler (https://developer.android.com/reference/android/os/Handler.html) handl
```

Registers a GPS Measurement callback.

Requires the `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permission.

Parameters	
`callback`	`GnssMeasurementsEvent.Callback`: a `GnssMeasurementsEvent.Callback` (https://developer.android.com/reference/android/location/GnssMeasurementsEvent.Callback.html) object to register.
`handler`	`Handler`: the handler that the callback runs on.

Returns	
`boolean`	`true` if the callback was added successfully, `false` otherwise.

registerGnssNavigationMessageCallback

added in API level 24 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
boolean registerGnssNavigationMessageCallback (GnssNavigationMessage.Callback (http
                Handler (https://developer.android.com/reference/android/os/Handler.html) handl
```

Registers a GNSS Navigation Message callback.

Requires the `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permission.

Parameters	
`callback`	`GnssNavigationMessage.Callback`: a `GnssNavigationMessage.Callback` (https://developer.android.com/reference/android/location/GnssNavigationMessage.Callback.html) object to register.
`handler`	`Handler`: the handler that the callback runs on.

Returns	
`boolean`	`true` if the callback was added successfully, `false` otherwise.

registerGnssNavigationMessageCallback

added in API level 24 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
boolean registerGnssNavigationMessageCallback (GnssNavigationMessage.Callback (http
```

Registers a GNSS Navigation Message callback.

Parameters	
`callback`	`GnssNavigationMessage.Callback`: a `GnssNavigationMessage.Callback` (https://developer.android.com/reference/android/location/GnssNavigationMessage.Callback.html) object to register.

Returns	
`boolean`	`true` if the callback was added successfully, `false` otherwise.

registerGnssStatusCallback

added in API level 24 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
boolean registerGnssStatusCallback (GnssStatus.Callback (https://developer.android.com/r
```

Registers a GNSS status callback.

Requires the `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permission.

Parameters	
`callback`	`GnssStatus.Callback`: GNSS status callback object to register

Returns	
`boolean`	true if the listener was successfully added

Throws	
`SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html)	if the ACCESS_FINE_LOCATION permission is not present

registerGnssStatusCallback

added in API level 24 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
boolean registerGnssStatusCallback (GnssStatus.Callback (https://developer.android.com/r
                Handler (https://developer.android.com/reference/android/os/Handler.html) handl
```

Registers a GNSS status callback.

Requires the `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permission.

Parameters	
`callback`	`GnssStatus.Callback`: GNSS status callback object to register
`handler`	`Handler`: the handler that the callback runs on.

Returns	
`boolean`	true if the listener was successfully added

Throws	
`SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html)	if the ACCESS_FINE_LOCATION permission is not present

removeGpsStatusListener

added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void removeGpsStatusListener (GpsStatus.Listener (https://developer.android.com/reference/
```

> **This method was deprecated in API level 24.**
> use `unregisterGnssStatusCallback(GnssStatus.Callback)` (https://developer.android.com/reference/android/location/LocationManager.html#unregisterGnssStatusCallback(android.location.GnssStatus.Callback)) instead.

Removes a GPS status listener.

Parameters	
`listener`	`GpsStatus.Listener`: GPS status listener object to remove

removeNmeaListener added in API level 24 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`void removeNmeaListener (OnNmeaMessageListener (https://developer.android.com/reference/and`

Removes an NMEA listener.

Parameters	
`listener`	`OnNmeaMessageListener`: a `OnNmeaMessageListener` `(https://developer.android.com/reference/android/location/OnNmeaMessageListener.html)` object to remove

removeNmeaListener added in API level 5 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`void removeNmeaListener (GpsStatus.NmeaListener (https://developer.android.com/reference/a`

> **This method was deprecated in API level 24.**
> use `removeNmeaListener(OnNmeaMessageListener)` `(https://developer.android.com/reference/android/location/LocationManager.html#removeNmeaListener(android.location.OnNmeaMessageListener))` instead.

Removes an NMEA listener.

Parameters	
`listener`	`GpsStatus.NmeaListener`: a `GpsStatus.NmeaListener` `(https://developer.android.com/reference/android/location/GpsStatus.NmeaListener.html)` object to remove

removeProximityAlert added in API level 1 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`void removeProximityAlert (PendingIntent (https://developer.android.com/reference/android/ap`

Removes the proximity alert with the given PendingIntent.

Before API version 17, this method could be used with `ACCESS_FINE_LOCATION` `(https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION)` or `ACCESS_COARSE_LOCATION` `(https://developer.android.com/reference/android`

requires `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permission.

Parameters	
`intent`	`PendingIntent`: the PendingIntent that no longer needs to be notified of proximity alerts

Throws	
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if intent is null
`SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html)	if `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permission is not present

removeTestProvider

added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`void removeTestProvider (String` (https://developer.android.com/reference/java/lang/String.html

Removes the mock location provider with the given name.

Parameters	
`provider`	`String`: the provider name

Throws	
`SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html)	if `mock location app op` (https://developer.android.com/reference/android/app/AppOpsManager.html#OPSTR_MOCK_LOCATION) is not set to `allowed` (https://developer.android.com/reference/android/app/AppOpsManager.html#MODE_ALLOWED) for your app.
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/...)	if no provider with the given name exists

removeUpdates added in API level 1 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void removeUpdates (LocationListener (https://developer.android.com/reference/android/locatio
```

Removes all location updates for the specified LocationListener.

Following this call, updates will no longer occur for this listener.

Parameters	
`listener`	`LocationListener`: listener object that no longer needs location updates

Throws	
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if listener is null

removeUpdates added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void removeUpdates (PendingIntent (https://developer.android.com/reference/android/app/Pending
```

Removes all location updates for the specified pending intent.

Following this call, updates will no longer for this pending intent.

Parameters	
`intent`	`PendingIntent`: pending intent object that no longer needs location updates

Throws	
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if intent is null

requestLocationUpdates

```
void requestLocationUpdates (String (https://developer.android.com/reference/java/lang/String
                long minTime,
                float minDistance,
                LocationListener (https://developer.android.com/reference/android/location/Loc
```

Register for location updates using the named provider, and a pending intent.

See `requestLocationUpdates(long, float, Criteria, PendingIntent)` (https://developer.android.com/reference/android/location/LocationManager.html#requestLocationUpdates(long, float, android.location.Criteria, android.app.PendingIntent)) for more detail on how to use this method.

Requires the `ACCESS_COARSE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_COARSE_LOCATION) or `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permissions.

Parameters	
`provider`	`String`: the name of the provider with which to register
`minTime`	`long`: minimum time interval between location updates, in milliseconds
`minDistance`	`float`: minimum distance between location updates, in meters
`listener`	`LocationListener`: a `LocationListener` (https://developer.android.com/reference/android/location/LocationListener.html) whose `onLocationChanged(Location)` (https://developer.android.com/reference/android/location/LocationListener.html#onLocationChanged(android.location.Location)) method will be called for each location update

Throws	
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if provider is null or doesn't exist on this device
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if listener is null
`RuntimeException`	if the calling thread has no Looper

/reference/java/lang /RuntimeException.html)	
SecurityException (https://developer.android.com /reference/java/lang /SecurityException.html)	if no suitable permission is present

requestLocationUpdates

added in API level 9 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void requestLocationUpdates (long minTime,
                float minDistance,
                Criteria (https://developer.android.com/reference/android/location/Criteria.html
                LocationListener (https://developer.android.com/reference/android/location/Loc
                Looper (https://developer.android.com/reference/android/os/Looper.html) looper)
```

Register for location updates using a Criteria, and a callback on the specified looper thread.

See `requestLocationUpdates(long, float, Criteria, PendingIntent)` (https://developer.android.com/reference/android/location /LocationManager.html#requestLocationUpdates(long, float, android.location.Criteria, android.app.PendingIntent)) for more detail on how to use this method.

Requires the `ACCESS_COARSE_LOCATION` (https://developer.android.com/reference/android /Manifest.permission.html#ACCESS_COARSE_LOCATION) or `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permissions.

Parameters	
minTime	long: minimum time interval between location updates, in milliseconds
minDistance	float: minimum distance between location updates, in meters
criteria	Criteria: contains parameters for the location manager to choose the appropriate provider and parameters to compute the location
listener	LocationListener: a LocationListener (https://developer.android.com /reference/android/location/LocationListener.html) whose onLocationChanged(Location) (https://developer.android.com/reference /android/location /LocationListener.html#onLocationChanged(android.location.Location)) method

	will be called for each location update
looper	Looper: a Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread

Throws	
IllegalArgumentException (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if criteria is null
IllegalArgumentException (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if listener is null
SecurityException (https://developer.android.com/reference/java/lang/SecurityException.html)	if no suitable permission is present

requestLocationUpdates

added in API level 1 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void requestLocationUpdates (String (https://developer.android.com/reference/java/lang/String
                long minTime,
                float minDistance,
                LocationListener (https://developer.android.com/reference/android/location/Loc
                Looper (https://developer.android.com/reference/android/os/Looper.html) looper)
```

Register for location updates using the named provider, and a callback on the specified looper thread.

See requestLocationUpdates(long, float, Criteria, PendingIntent) (https://developer.android.com/reference/android/location/LocationManager.html#requestLocationUpdates(long, float, android.location.Criteria, android.app.PendingIntent)) for more detail on how to use this method.

Requires the ACCESS_COARSE_LOCATION (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_COARSE_LOCATION) or ACCESS_FINE_LOCATION (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION)

Parameters	
`provider`	`String`: the name of the provider with which to register
`minTime`	`long`: minimum time interval between location updates, in milliseconds
`minDistance`	`float`: minimum distance between location updates, in meters
`listener`	`LocationListener` (https://developer.android.com/reference/android/location/LocationListener.html) whose `onLocationChanged(Location)` (https://developer.android.com/reference/android/location/LocationListener.html#onLocationChanged(android.location.Location)) method will be called for each location update
`looper`	`Looper`: a Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread

Throws	
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if provider is null or doesn't exist
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if listener is null
`SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html)	if no suitable permission is present

requestLocationUpdates

added in API level 9 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void requestLocationUpdates (long minTime,
                float minDistance,
                Criteria (https://developer.android.com/reference/android/location/Criteria.html
                PendingIntent (https://developer.android.com/reference/android/app/PendingInten
```

The `requestLocationUpdates()` and `requestSingleUpdate()` register the current activity to be updated periodically by the named provider, or by the provider matching the specified `Criteria` (https://developer.android.com/reference/android/location/Criteria.html), with location and status updates.

It may take a while to receive the first location update. If an immediate location is required, applications may use the `getLastKnownLocation(String)` (https://developer.android.com/reference/android/location/LocationManager.html#getLastKnownLocation(java.lang.String)) method.

Location updates are received either by `LocationListener` (https://developer.android.com/reference/android/location/LocationListener.html) callbacks, or by broadcast intents to a supplied `PendingIntent` (https://developer.android.com/reference/android/app/PendingIntent.html).

If the caller supplied a pending intent, then location updates are sent with a key of `KEY_LOCATION_CHANGED` (https://developer.android.com/reference/android/location/LocationManager.html#KEY_LOCATION_CHANGED) and a `Location` (https://developer.android.com/reference/android/location/Location.html) value.

The location update interval can be controlled using the minTime parameter. The elapsed time between location updates will never be less than minTime, although it can be more depending on the Location Provider implementation and the update interval requested by other applications.

Choosing a sensible value for minTime is important to conserve battery life. Each location update requires power from GPS, WIFI, Cell and other radios. Select a minTime value as high as possible while still providing a reasonable user experience. If your application is not in the foreground and showing location to the user then your application should avoid using an active provider (such as `NETWORK_PROVIDER` (https://developer.android.com/reference/android/location/LocationManager.html#NETWORK_PROVIDER) or `GPS_PROVIDER` (https://developer.android.com/reference/android/location/LocationManager.html#GPS_PROVIDER)), but if you insist then select a minTime of 5 * 60 * 1000 (5 minutes) or greater. If your application is in the foreground and showing location to the user then it is appropriate to select a faster update interval.

The minDistance parameter can also be used to control the frequency of location updates. If it is greater than 0 then the location provider will only send your application an update when the location has changed by at least minDistance meters, AND at least minTime milliseconds have passed. However it is more difficult for location providers to save power using the minDistance parameter, so minTime should be the primary tool to conserving battery life.

If your application wants to passively observe location updates triggered by other applications, but not consume any additional power otherwise, then use the `PASSIVE_PROVIDER` (https://developer.android.com/reference/android/location/LocationManager.html#PASSIVE_PROVIDER) This provider does not actively turn on or modify active location providers, so you do not need to be as

location update (such as network activity) then you should select non-zero values for minTime and/or minDistance to rate-limit your update frequency in the case another application enables a location provider with extremely fast updates.

In case the provider is disabled by the user, updates will stop, and a provider availability update will be sent. As soon as the provider is enabled again, location updates will immediately resume and a provider availability update sent. Providers can also send status updates, at any time, with extra's specific to the provider. If a callback was supplied then status and availability updates are via `onProviderDisabled(String)` (https://developer.android.com/reference/android/location/LocationListener.html#onProviderDisabled(java.lang.String)), `onProviderEnabled(String)` (https://developer.android.com/reference/android/location/LocationListener.html#onProviderEnabled(java.lang.String)) or `onStatusChanged(String, int, Bundle)` (https://developer.android.com/reference/android/location/LocationListener.html#onStatusChanged(java.lang.String, int, android.os.Bundle)). Alternately, if a pending intent was supplied then status and availability updates are broadcast intents with extra keys of `KEY_PROVIDER_ENABLED` (https://developer.android.com/reference/android/location/LocationManager.html#KEY_PROVIDER_ENABLED) or `KEY_STATUS_CHANGED` (https://developer.android.com/reference/android/location/LocationManager.html#KEY_STATUS_CHANGED).

If a `LocationListener` (https://developer.android.com/reference/android/location/LocationListener.html) is used but with no Looper specified then the calling thread must already be a `Looper` (https://developer.android.com/reference/android/os/Looper.html) thread such as the main thread of the calling Activity. If a Looper is specified with a `LocationListener` (https://developer.android.com/reference/android/location/LocationListener.html) then callbacks are made on the supplied Looper thread.

> Prior to Jellybean, the minTime parameter was only a hint, and some location provider implementations ignored it. From Jellybean and onwards it is mandatory for Android compatible devices to observe both the minTime and minDistance parameters.

Requires the `ACCESS_COARSE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_COARSE_LOCATION) or `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permissions.

Parameters	
`minTime`	`long`: minimum time interval between location updates, in milliseconds
`minDistance`	`float`: minimum distance between location updates, in meters
`criteria`	`Criteria`: contains parameters for the location manager to choose the

intent	PendingIntent: a PendingIntent (https://developer.android.com/reference/android/app/PendingIntent.html) to be sent for each location update

Throws	
IllegalArgumentException (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if criteria is null
IllegalArgumentException (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if intent is null
SecurityException (https://developer.android.com/reference/java/lang/SecurityException.html)	if no suitable permission is present

requestLocationUpdates

added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void requestLocationUpdates (String (https://developer.android.com/reference/java/lang/String
                long minTime,
                float minDistance,
                PendingIntent (https://developer.android.com/reference/android/app/PendingInten
```

Register for location updates using the named provider, and a pending intent.

See requestLocationUpdates(long, float, Criteria, PendingIntent) (https://developer.android.com/reference/android/location/LocationManager.html#requestLocationUpdates(long, float, android.location.Criteria, android.app.PendingIntent)) for more detail on how to use this method.

Requires the ACCESS_COARSE_LOCATION (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_COARSE_LOCATION) or ACCESS_FINE_LOCATION (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permissions.

Parameters

minTime	long: minimum time interval between location updates, in milliseconds
minDistance	float: minimum distance between location updates, in meters
intent	PendingIntent: a PendingIntent (https://developer.android.com/reference/android/app/PendingIntent.html) to be sent for each location update

Throws	
IllegalArgumentException (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if provider is null or doesn't exist on this device
IllegalArgumentException (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if intent is null
SecurityException (https://developer.android.com/reference/java/lang/SecurityException.html)	if no suitable permission is present

requestSingleUpdate

added in API level 9 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void requestSingleUpdate (String (https://developer.android.com/reference/java/lang/String.htm
                PendingIntent (https://developer.android.com/reference/android/app/PendingInten
```

Register for a single location update using a named provider and pending intent.

See requestLocationUpdates(long, float, Criteria, PendingIntent) (https://developer.android.com/reference/android/location/LocationManager.html#requestLocationUpdates(long, float, android.location.Criteria, android.app.PendingIntent)) for more detail on how to use this method.

Requires the ACCESS_COARSE_LOCATION (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_COARSE_LOCATION) or ACCESS_FINE_LOCATION (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permissions.

provider	String: the name of the provider with which to register
intent	PendingIntent: a PendingIntent (https://developer.android.com/reference/android/app/PendingIntent.html) to be sent for the location update

Throws	
IllegalArgumentException (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if provider is null or doesn't exist
IllegalArgumentException (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if intent is null
SecurityException (https://developer.android.com/reference/java/lang/SecurityException.html)	if no suitable permission is present

requestSingleUpdate

added in API level 9 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void requestSingleUpdate (Criteria (https://developer.android.com/reference/android/location/C
                PendingIntent (https://developer.android.com/reference/android/app/PendingInten
```

Register for a single location update using a Criteria and pending intent.

See requestLocationUpdates(long, float, Criteria, PendingIntent) (https://developer.android.com/reference/android/location/LocationManager.html#requestLocationUpdates(long, float, android.location.Criteria, android.app.PendingIntent)) for more detail on how to use this method.

Requires the ACCESS_COARSE_LOCATION (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_COARSE_LOCATION) or ACCESS_FINE_LOCATION (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permissions.

Parameters

`criteria`	`Criteria`: contains parameters for the location manager to choose the appropriate provider and parameters to compute the location
`intent`	`PendingIntent`: a `PendingIntent` (https://developer.android.com/reference/android/app/PendingIntent.html) to be sent for the location update

Throws	
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if provider is null or doesn't exist
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if intent is null
`SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html)	if no suitable permission is present

requestSingleUpdate

added in API level 9 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void requestSingleUpdate (String (https://developer.android.com/reference/java/lang/String.htm
                LocationListener (https://developer.android.com/reference/android/location/Loc
                Looper (https://developer.android.com/reference/android/os/Looper.html) looper)
```

Register for a single location update using the named provider and a callback.

See `requestLocationUpdates(long, float, Criteria, PendingIntent)` (https://developer.android.com/reference/android/location/LocationManager.html#requestLocationUpdates(long, float, android.location.Criteria, android.app.PendingIntent)) for more detail on how to use this method.

Requires the `ACCESS_COARSE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_COARSE_LOCATION) or `ACCESS_FINE_LOCATION` (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permissions.

Parameters	
`provider`	`String`: the name of the provider with which to register
`listener`	`LocationListener`: a `LocationListener` (https://developer.android.com/reference/android/location/LocationListener.html) whose `onLocationChanged(Location)` (https://developer.android.com/reference/android/location/LocationListener.html#onLocationChanged(android.location.Location)) method will be called when the location update is available
`looper`	`Looper`: a Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread

Throws	
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if provider is null or doesn't exist
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if listener is null
`SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html)	if no suitable permission is present

requestSingleUpdate

added in API level 9 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void requestSingleUpdate (Criteria (https://developer.android.com/reference/android/location/C
                LocationListener (https://developer.android.com/reference/android/location/Loc
                Looper (https://developer.android.com/reference/android/os/Looper.html) looper)
```

Register for a single location update using a Criteria and a callback.

See `requestLocationUpdates(long, float, Criteria, PendingIntent)` (https://developer.android.com/reference/android/location/LocationManager.html#requestLocationUpdates(long, float, android.location.Criteria,

Requires the ACCESS_COARSE_LOCATION (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_COARSE_LOCATION) or ACCESS_FINE_LOCATION (https://developer.android.com/reference/android/Manifest.permission.html#ACCESS_FINE_LOCATION) permissions.

Parameters	
criteria	Criteria: contains parameters for the location manager to choose the appropriate provider and parameters to compute the location
listener	LocationListener: a LocationListener (https://developer.android.com/reference/android/location/LocationListener.html) whose onLocationChanged(Location) (https://developer.android.com/reference/android/location/LocationListener.html#onLocationChanged(android.location.Location)) method will be called when the location update is available
looper	Looper: a Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread

Throws	
IllegalArgumentException (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if criteria is null
IllegalArgumentException (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if listener is null
SecurityException (https://developer.android.com/reference/java/lang/SecurityException.html)	if no suitable permission is present

sendExtraCommand

Added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
boolean sendExtraCommand (String (https://developer.android.com/reference/java/lang/String.htm
                String (https://developer.android.com/reference/java/lang/String.html) command,
```

Sends additional commands to a location provider. Can be used to support provider specific extensions to the Location Manager API

Parameters	
`provider`	`String`: name of the location provider.
`command`	`String`: name of the command to send to the provider.
`extras`	`Bundle`: optional arguments for the command (or null). The provider may optionally fill the extras Bundle with results from the command.

Returns	
`boolean`	true if the command succeeds.

setTestProviderEnabled

added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void setTestProviderEnabled (String (https://developer.android.com/reference/java/lang/String
                boolean enabled)
```

Sets a mock enabled value for the given provider. This value will be used in place of any actual value from the provider.

Parameters	
`provider`	`String`: the provider name
`enabled`	`boolean`: the mock enabled value

Throws	
`SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html)	if `mock location app op` (https://developer.android.com/reference/android/app/AppOpsManager.html#OPSTR_MOCK_LOCATION) is not set to `allowed` (https://developer.android.com/reference/android/app/AppOpsManager.html#MODE_ALLOWED) for your app.
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if no provider with the given name exists

setTestProviderLocation

added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void setTestProviderLocation (String (https://developer.android.com/reference/java/lang/Strin
                Location (https://developer.android.com/reference/android/location/Location.html
```

Sets a mock location for the given provider.

This location will be used in place of any actual location from the provider. The location object must have a minimum number of fields set to be considered a valid LocationProvider Location, as per documentation on `Location` (https://developer.android.com/reference/android/location/Location.html) class.

Parameters	
`provider`	`String`: the provider name
`loc`	`Location`: the mock location

Throws	
`SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html)	if `mock location app op` (https://developer.android.com/reference/android/app/AppOpsManager.html#OPSTR_MOCK_LOCATION) is not set to `allowed` (https://developer.android.com/reference/android/app/AppOpsManager.html#MODE_ALLOWED) for your app.
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if no provider with the given name exists
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if the location is incomplete

setTestProviderStatus

added in API level 3 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void setTestProviderStatus (String (https://developer.android.com/reference/java/lang/String.h
                int status,
                Bundle (https://developer.android.com/reference/android/os/Bundle.html) extras,
                long updateTime)
```

values from the provider.

Parameters	
`provider`	`String`: the provider name
`status`	`int`: the mock status
`extras`	`Bundle`: a Bundle containing mock extras
`updateTime`	`long`: the mock update time

Throws	
`SecurityException` (https://developer.android.com/reference/java/lang/SecurityException.html)	if `mock location app op` (https://developer.android.com/reference/android/app/AppOpsManager.html#OPSTR_MOCK_LOCATION) is not set to `allowed` (https://developer.android.com/reference/android/app/AppOpsManager.html#MODE_ALLOWED) for your app.
`IllegalArgumentException` (https://developer.android.com/reference/java/lang/IllegalArgumentException.html)	if no provider with the given name exists

unregisterGnssMeasurementsCallback

added in API level 24 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`void unregisterGnssMeasurementsCallback (GnssMeasurementsEvent.Callback (https://dev`

Unregisters a GPS Measurement callback.

Parameters	
`callback`	`GnssMeasurementsEvent.Callback`: a `GnssMeasurementsEvent.Callback` (https://developer.android.com/reference/android/location/GnssMeasurementsEvent.Callback.html) object to remove.

unregisterGnssNavigationMessageCallback

added in API level 24 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

`void unregisterGnssNavigationMessageCallback (GnssNavigationMessage.Callback (https`

Parameters	
`callback`	`GnssNavigationMessage.Callback`: a `GnssNavigationMessage.Callback` (https://developer.android.com/reference/android/location/GnssNavigationMessage.Callback.html) object to remove.

unregisterGnssStatusCallback

added in API level 24 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

```
void unregisterGnssStatusCallback (GnssStatus.Callback (https://developer.android.com/ref
```

Removes a GNSS status callback.

Parameters	
`callback`	`GnssStatus.Callback`: GNSS status callback object to remove