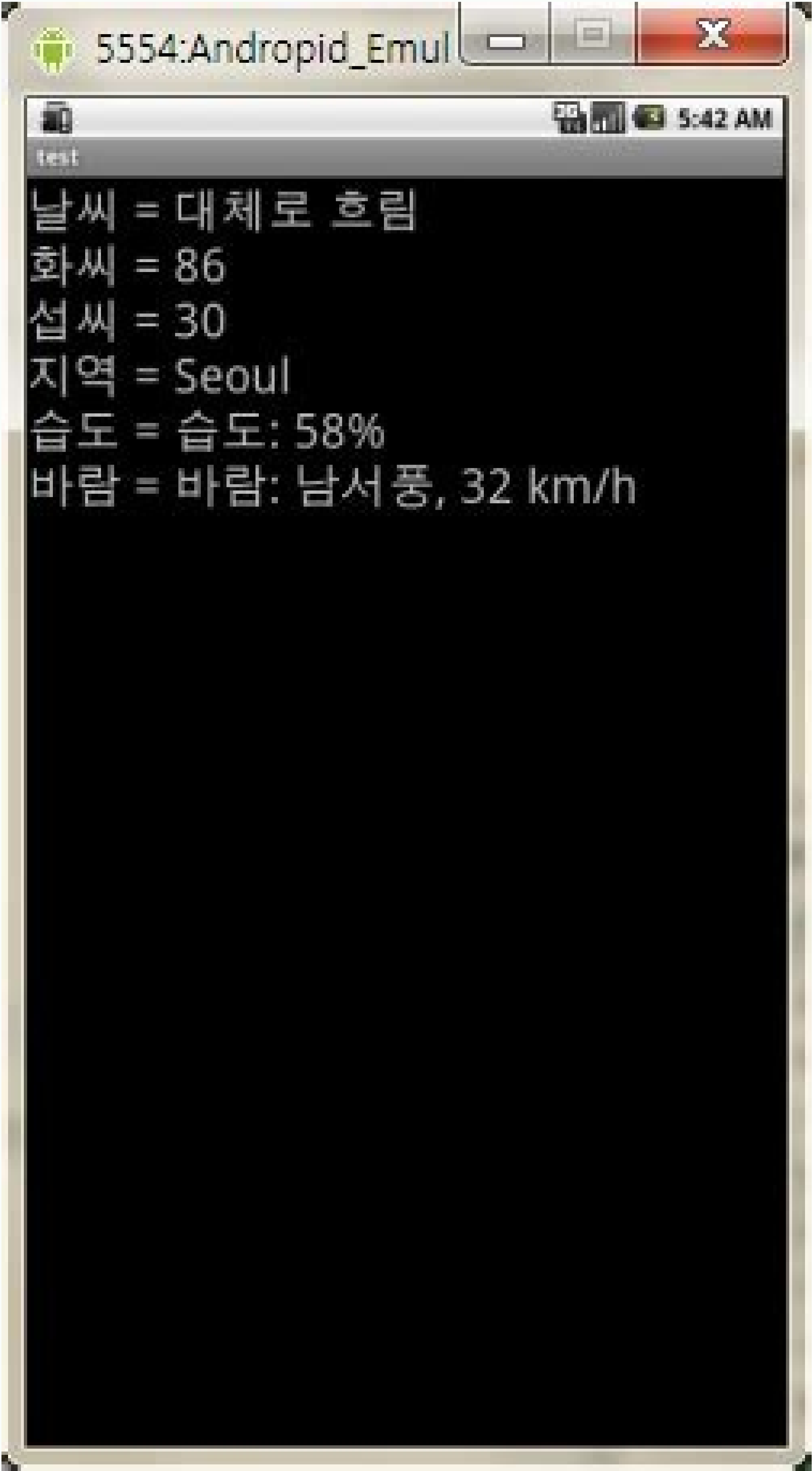


Continue



Emulating GPS Location

Use DDMS > Emulator Control



Android geocoder getfromlocation not working. Android geocoder get address. Android geocoder getfromlocation deprecated. Google maps geocoder example. Google geocoder example. Android geocoder.getfromlocation returns null. Grpc failed at android.location.geocoder.getfromlocation. Android geocoder getfromlocationname.

In the previous Android Google Map and Android Google Map Show current location tutorials, we covered showing a simple Google map and your current location, respectively. In this tutorial we will now implement the Google map location search function. Searching for a location in the Google Map API is done using the Geocoder class. The Geocoder class is used to support geocoding and reverse geocoding. Geocoding is the process of converting a street address into coordinates (latitude and longitude). Reverse geocoding is the process of converting coordinates (latitude and longitude) into an address. Methods of the Geocoder Class List getFromLocation(double latitude, double longitude, int maxResults): This method returns an array of addresses specifying the latitude and longitude of the local area. List getFromLocationName(String location, int Results, double leftWidth, double leftLength, double rightWidth, double rightLength): This method returns an array of addresses describing a specific location, e.g. a location, an address, etc. List getFromLocationName(String location, int Results): This method returns an array of addresses that describe a specific location, such as city, address, etc. static boolean isPresent(): This method returns true if the getFromLocation() and getFromLocationName() methods are implemented. Let's look at the code that converts location names to coordinates. List addressList = geocoder.getFromLocationName(location, 1); address address = list of addresses.get(0); LatLng latLng = new LatLng (address.getLatitude(), address.getLongitude()); Let's look at an example of a Google map entering a location. activity_maps.xml Add a SupportMapFragment, EditText, and Button to the activity_maps.xml file. = Build.VERSION_CODES.M) { if (ContextCompat.checkSelfPermission(this, Manifest.permission.ACCESS_FINE_LOCATION) == PackageManager.PERMISSION_GRANTED) { buildGoogleApiClient(); mMap.setMyLocationEnabled(true); } } else { buildGoogleApiClient(); mMap.setMyLocationEnabled(true); } } protected synchronized void buildGoogleApiClient() { mGoogleApiClient = new GoogleApiClient.Builder(this).addConnectionCallbacks(this).addOnConnectionFailedListener(this).addApi(LocationServices.API).build(); mGoogleApiClient.connect(); } @Override public void onConnected(Bundle Bundle) { mLocationRequest = new LocationRequest(); mLocationRequest.setInterval(1000); mLocationRequest.setFastestInterval(1000); mLocationRequest.setPriority(LocationRequest.PRIORITY_BALANCED_POWER_ACCURACY); if(Manifest.permission.ACCESS_FINE_LOCATION) == PackageManager.PERMISSION_GRANTED) { LocationServices.FusedLocationApi.requestLocationUpdates(mGoogleApiClient, mLocationRequest, this); } } @Override public void onConnectionSuspended(int i) { } @Override public void onLocationChanged(Location location) { mLastLocation = location; if (mCurrLocationMarker != null) { mCurrLocationMarker.remove(); } // place current location marker LatLng latLng = new LatLng(location.getLatitude(), location.getLongitude()); marker parameters marker parameters = new marker parameters(); tokenOptions.position(latLng); markerOptions.title("Current position"); markerOptions.icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_GREEN)); mCurrLocationMarker = mMap.addMarker(markerOptions); //Move the map camera mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng)); mMap.animateCamera(CameraUpdateFactory.zoomTo(11)); // stop updating location if (mGoogleApiClient != null) { LocationServices.FusedLocationApi.removeLocationUpdates(mGoogleApiClient, this); } } @Override public void onConnectionFailed(ConnectionResult connectionResult) { } public void searchLocation(View view) { EditText locationSearch = (EditText) findViewById(R.id.editText); String location = locationSearch.getText().toString(); list addressList = null; if (location != null || !location.equals("")) { Geocoder geocoder = new Geocoder(this); try { addressList = geocoder.getFromLocationName(location, 1); } catch (IOException e) { e.printStackTrace(); } address address = addressList.get(0); LatLng latLng = new LatLng(address.getLatitude(), address.getLongitude()); mMap.addMarker(new MarkerOptions().position(latLng).title(location)); mMap.animateCamera(CameraUpdateFactory.newLatLng(latLng)); Toast.makeText(getApplicationContext(),address.getLatitude()+" "+address.getLongitude(),Toast.LENGTH_LONG).show(); } } } Add the following user permission to the AndroidManifest.xml file. < use-permission/>