1. Briefly explain the role of the three input data types to the AsyncTask.

Private class getLogo extends AsyncTask<String, Integer, Bitmap>

* 1st data type (String) represents the data type of AsyncTask’s executing parameters which also tells the type of doInBackground’s input parameters
* 2nd data type (Integer) represents the type of progress units published during the background computation
* 3rd data type (Bitmap) represents the type of return value of doInBackground, which is also known as the type of onPostExecute’s input parameter

1. Questions:
   1. What should you use to transform a LatLong point into a street address.

Reverse geocoding should be done via a geocoder instance’s **getFromLocation** method to translate the LatLong value to an address

* 1. What is the meaning of “AsyncTask is a Generic class”?

Since AsyncTask’s three input data types are not finalised and could be parameterised over types (could be modified), AsyncTask is a Generic class.

1. Questions:
   1. How can an input parameter be accessed in doInBackground?

After it is passed into doInBackground, the parameter behaves like an array, so the index is used to access the input parameter (for example, String selectedCountry = params[0])

* 1. When and on what thread does doInBackground execute?

After execute() is called and on the background thread

* 1. When and on what thread does onPostExecute execute?

After doInBackground finishes and on the main UI thread.

* 1. When and on what thread does onProgressUpdate execute?

Runs on the UI thread after publishProgress() is invoked