Android Application Development
Lab 3 - Ασκήσεις μελέτης Β2

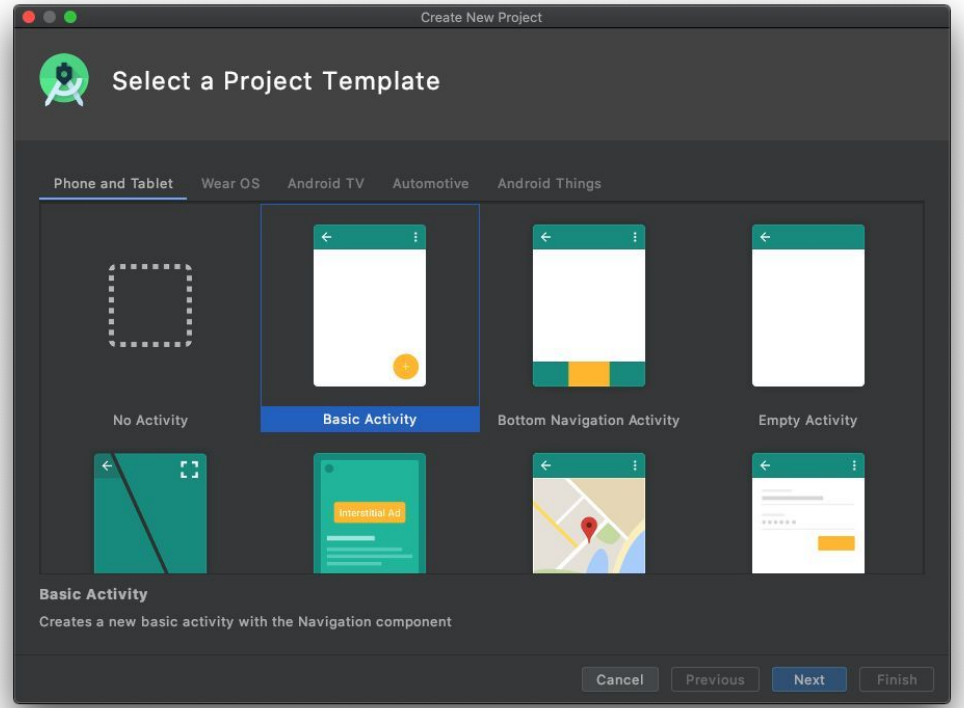
Human-Computer Interaction, AUEB
Εαρινό εξάμηνο 2024-2025

Lab Assistant: Sofia Eleftheriou

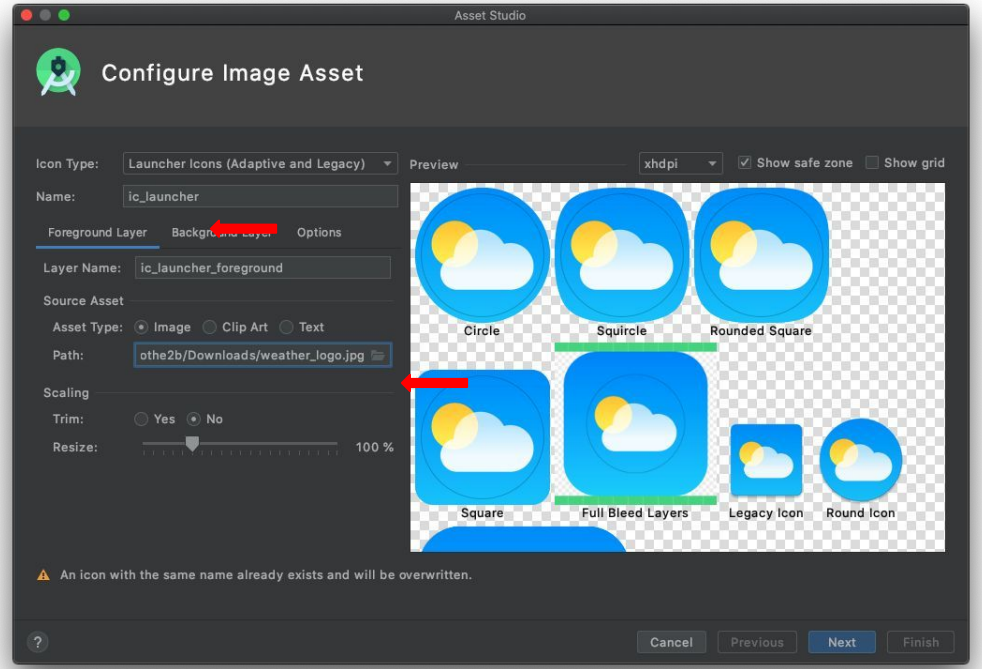
Android Development Advanced Use

- Catch up with Lab 1-2
- Add Voice Recognition functionality
  - Update Business Logic to use voice commands
  - Demonstrate updates

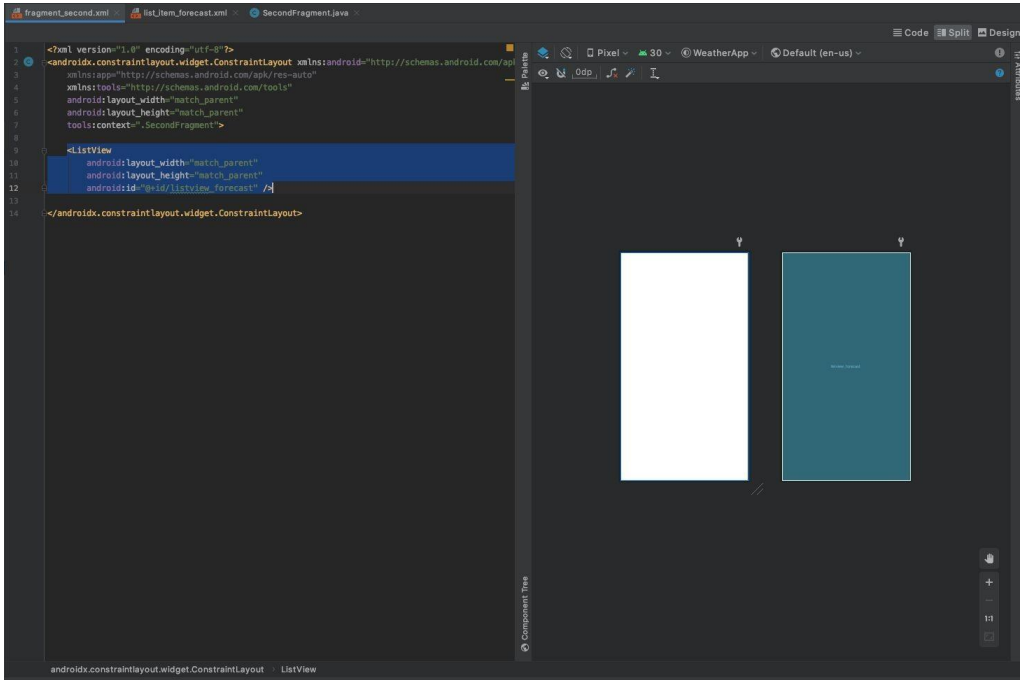
Catch up with Lab 1



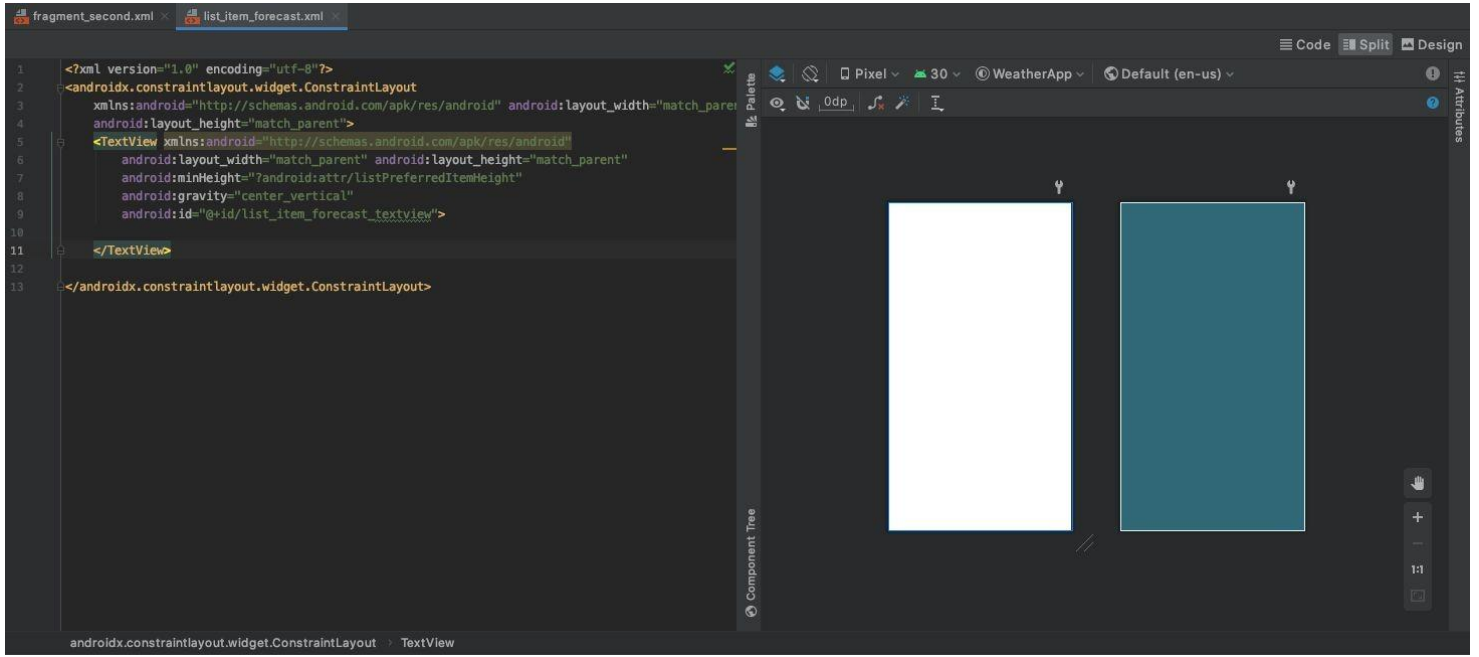
Create new project with Basic Activity



Change Application Icon



Add ListView to layout

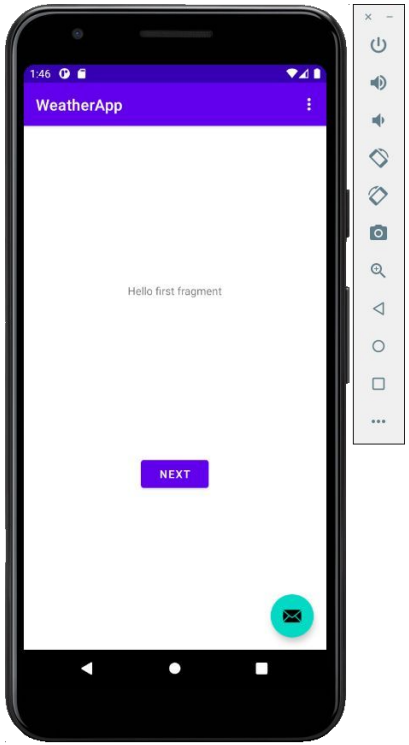


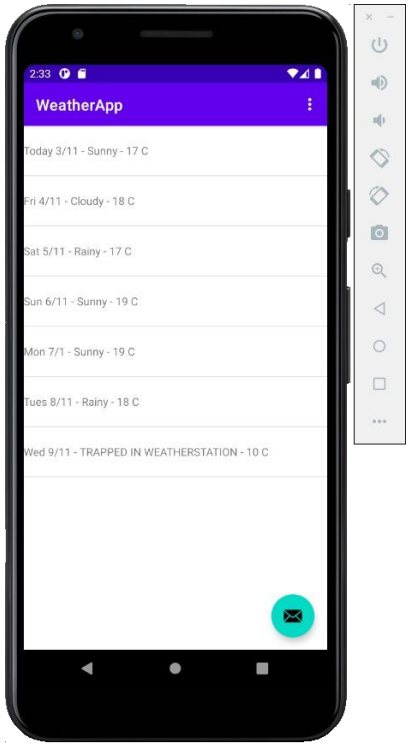
Add new layout with TextView for list items

SecondFragment.java

```java
package com.example.weatherapp;

import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

import androidx.fragment.app.Fragment;
import java.util.*;
import android.widget.ArrayAdapter;
import android.widget.ListView;

public class SecondFragment extends Fragment {

    @Override
    public View onCreateView(
            LayoutInflater inflater, ViewGroup container,
            Bundle savedInstanceState
    ) {
        ArrayAdapter<String> mForecastAdapter;
        String[] data = {
                "Today 3/11 - Sunny - 17 C",
                "Fri 4/11 - Cloudy - 18 C",
                "Sat 5/11 - Rainy - 17 C",
                "Sun 6/11 - Sunny - 19 C",
                "Mon 7/1 - Sunny - 19 C",
                "Tues 8/11 - Rainy - 18 C",
                "Wed 9/11 - TRAPPED IN WEATHERSTATION - 10 C"
        };

        List<String> weekForecast = new ArrayList<String>(Arrays.asList(data));
        mForecastAdapter = new ArrayAdapter<String>(getActivity(), // The current context (this activity)
                R.layout.list_item_forecast, // The name of the layout ID.
                R.id.list_item_forecast_textview, // The ID of the textview to populate.
                weekForecast);
        View rootView = inflater.inflate(R.layout. fragment_second, container,  attachToRoot: false);
        ListView listView = (ListView) rootView.findViewById(R.id. listview_forecast);
        listView.setAdapter(mForecastAdapter);
        return rootView;

    }

}
```

New business logic / Populate list with mock (fake) data


1:46
WeatherApp
Hello first fragment
NEXT

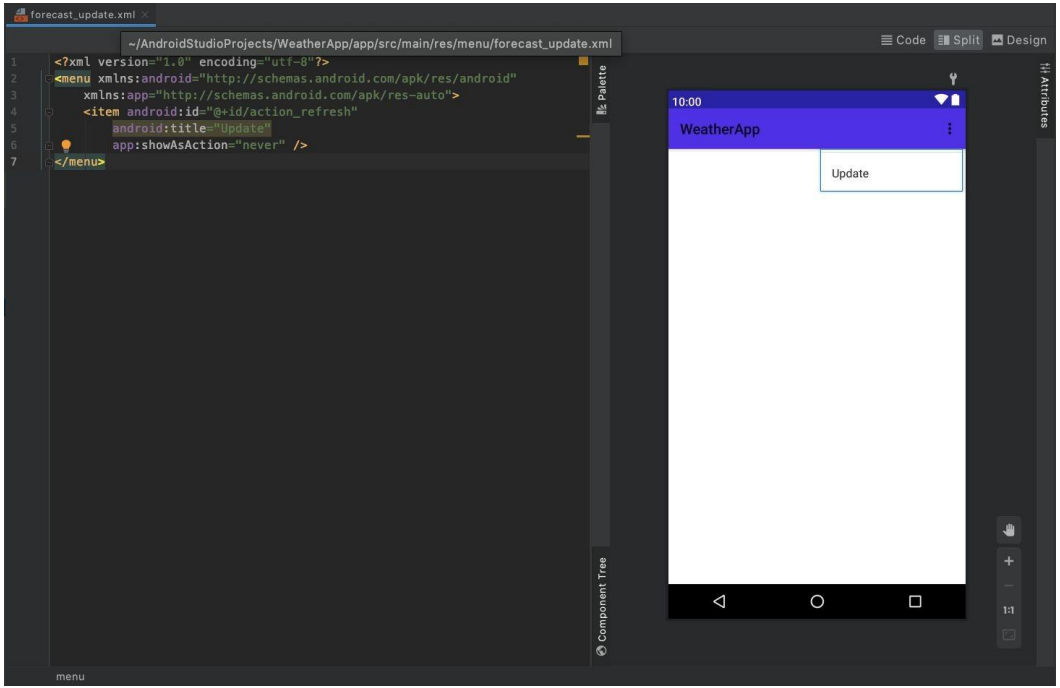

2:33
WeatherApp
Today 3/11 - Sunny - 17 C
Fri 4/11 - Cloudy - 18 C
Sat 5/11 - Rainy - 17 C
Sun 6/11 - Sunny - 19 C
Mon 7/1 - Sunny - 19 C
Tues 8/11 - Rainy - 18 C
Wed 9/11 - TRAPPED IN WEATHERSTATION - 10 C

Catch up with Lab 2

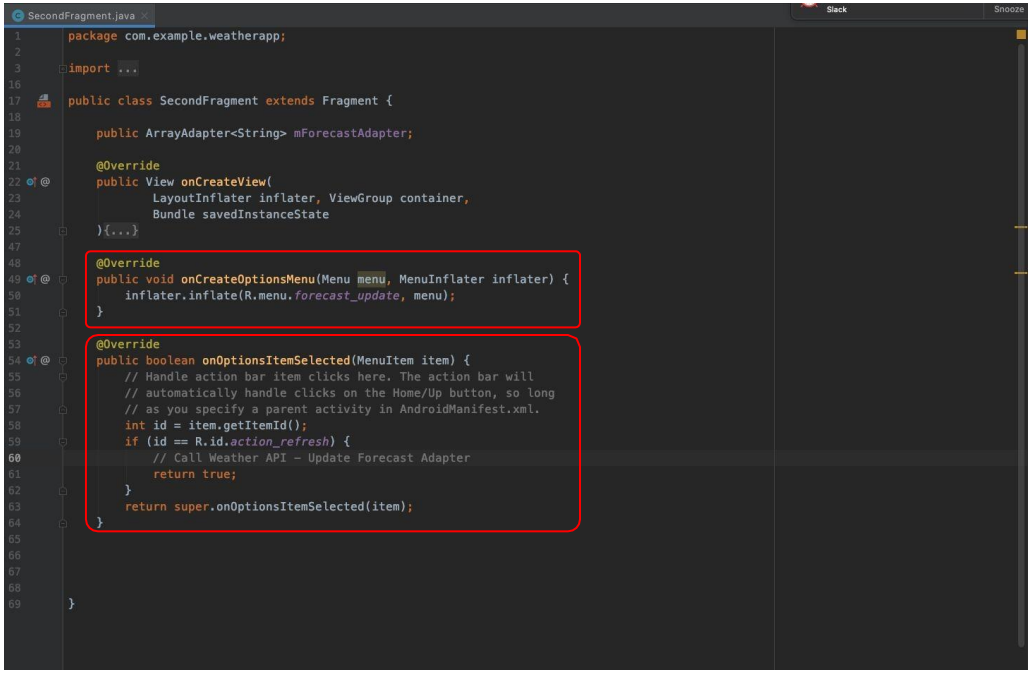
SecondFragment.java | FirstFragment.java

```
import ...

public class SecondFragment extends Fragment {

    public ArrayAdapter<String> mForecastAdapter;

    @Override
    public View onCreateView(
            LayoutInflater inflater, ViewGroup container,
            Bundle savedInstanceState
    ){
        String[] data = {
                "Today 3/11 - Sunny - 17 C",
                "Fri 4/11 - Cloudy - 18 C",
                "Sat 5/11 - Rainy - 17 C",
                "Sun 6/11 - Sunny - 19 C",
                "Mon 7/1 - Sunny - 19 C",
                "Tues 8/11 - Rainy - 18 C",
                "Wed 9/11 - TRAPPED IN WEATHERSTATION - 10 C"
        };

        List<String> weekForecast = new ArrayList<>(Arrays.asList(data));
        mForecastAdapter = new ArrayAdapter<String>(getActivity(), // The current context (this activity)
                R.layout.list_item_forecast, // The name of the layout ID.
                R.id.list_item_forecast_textview, // The ID of the textview to populate.
                weekForecast);
        View rootView = inflater.inflate(R.layout. fragment_second, container, attachToRoot: false);
        ListView listView = (ListView) rootView.findViewById(R.id. listview_forecast);
        listView.setAdapter(mForecastAdapter);
        return rootView;

    }

}
```

Turn Array Adapter should be public to be amendable



Add menu option (Update Forecast)



Update business logic to support new menu

AndroidManifest.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.weatherapp">
    <uses-permission android:name="android.permission.INTERNET"/>
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="WeatherApp"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.WeatherApp">
        <activity
            android:name=".MainActivity"
            android:label="WeatherApp"
            android:theme="@style/Theme.WeatherApp.NoActionBar">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

manifest › uses-permission

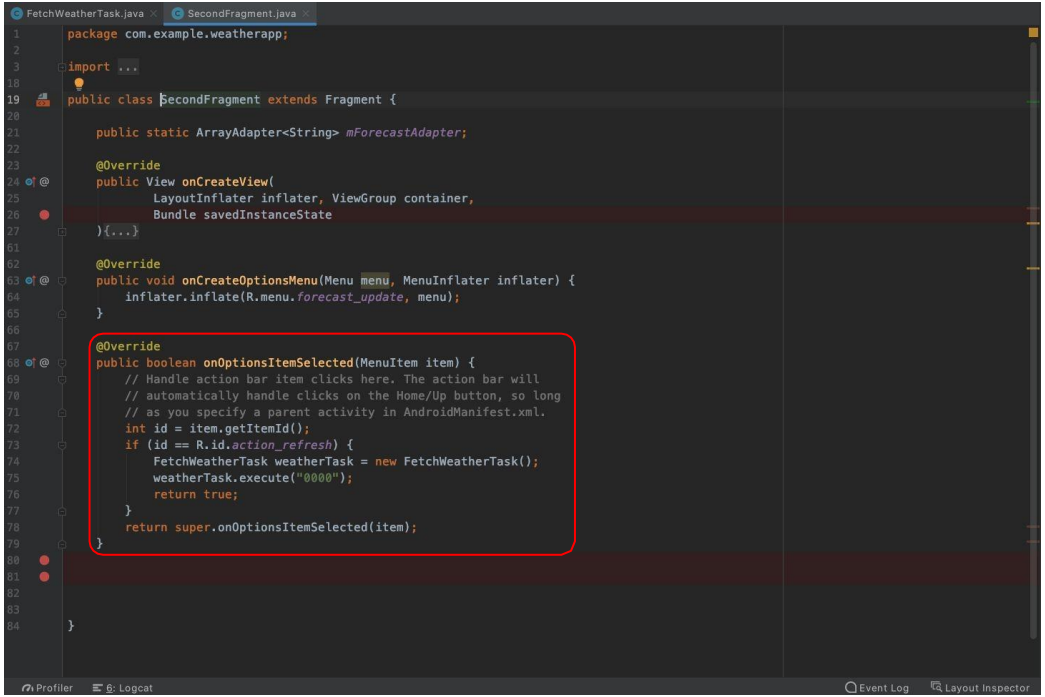
Text | Merged Manifest

Add permission for internet in manifest

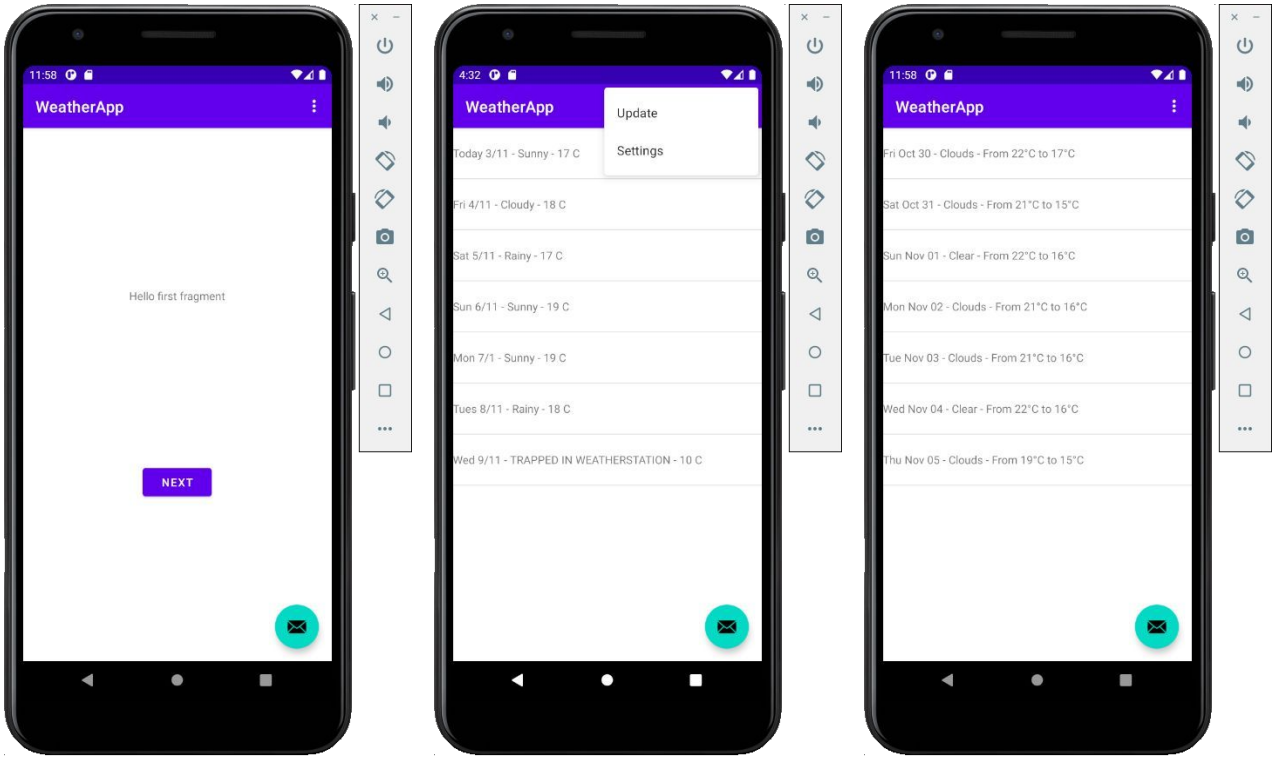
SecondFragment.java | FetchWeatherTask.java

```
1   package com.example.weatherapp;
2   import ...
17
18  public class FetchWeatherTask extends AsyncTask<String, Void, String[]> {
19      private final String LOG_TAG = FetchWeatherTask.class.getSimpleName();
20
21      /** Prepare the weather high/lows for presentation. */
24      private String formatHighLows(double high, double low) {...}
32
33      /** Take the String representing the complete forecast in JSON Format and ...*/
40      private String[] getWeatherDataFromJson(String forecastJsonStr, int numDays)
41              throws JSONException {...}
96
97      @Override
98      protected String[] doInBackground(String... params) {...}
176
177     @Override
178     protected void onPostExecute(String[] result) {...}
187 }
188
```

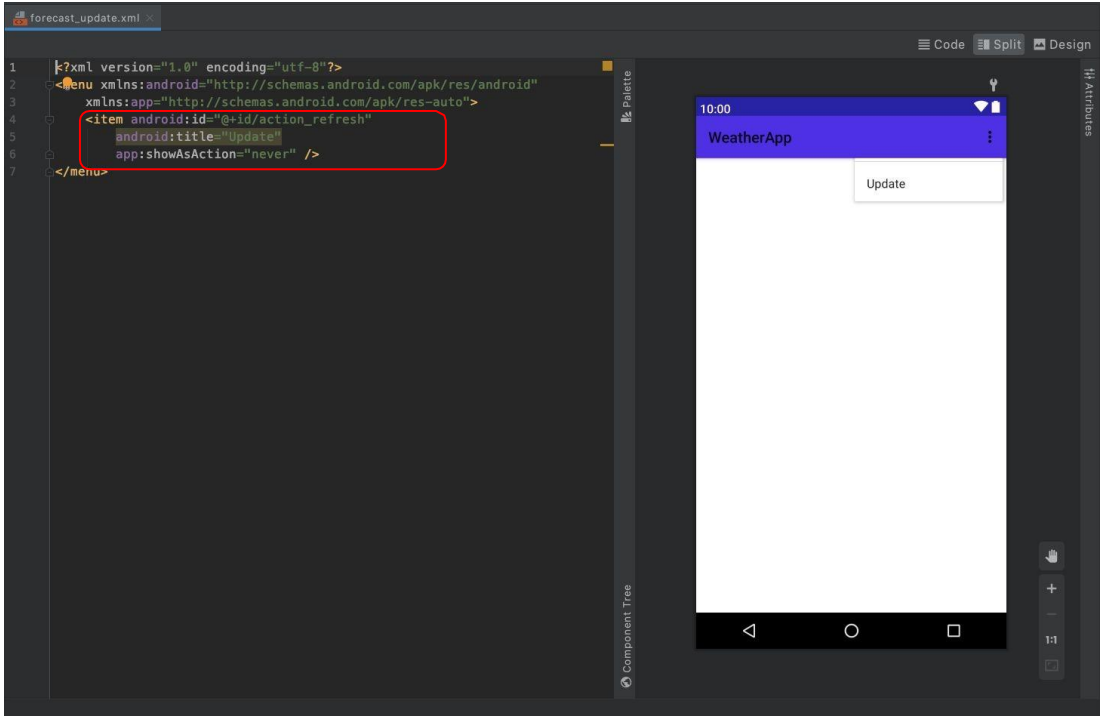
Build business logic to fetch and preview forecast



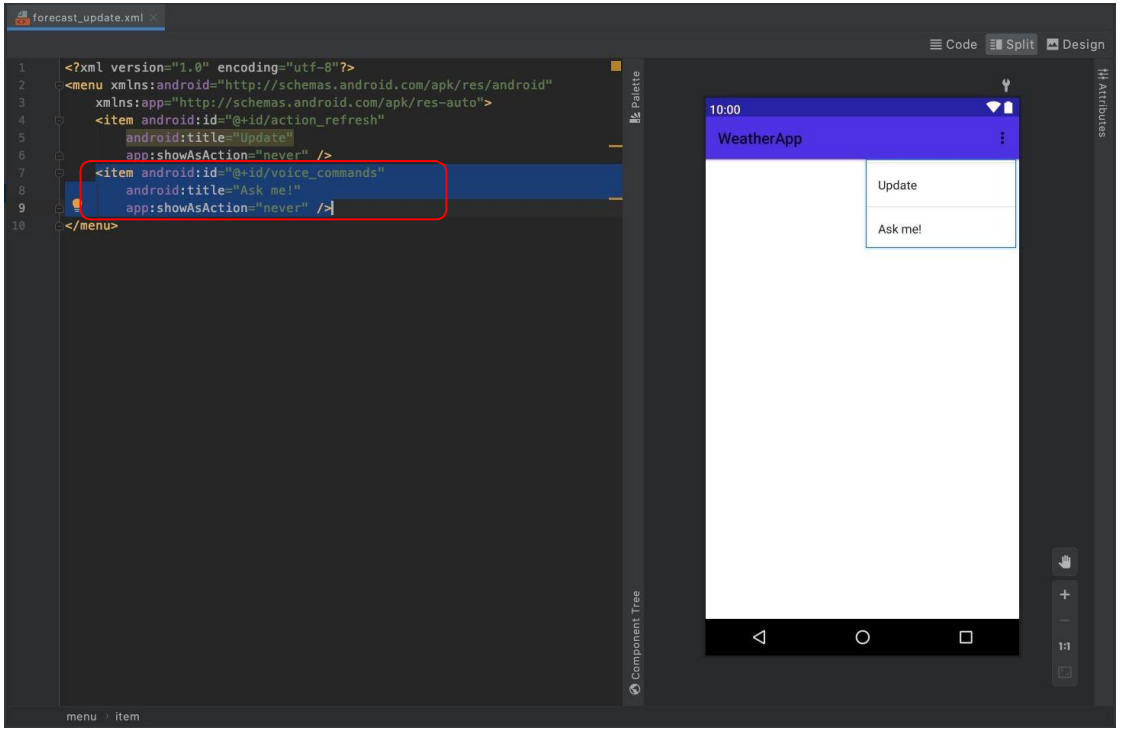
Build business logic to fetch and preview forecast


11:58
WeatherApp
Hello first fragment
NEXT
4:32
WeatherApp
Update
Settings
Today 3/11 - Sunny - 17 C
Fri 4/11 - Cloudy - 18 C
Sat 5/11 - Rainy - 17 C
Sun 6/11 - Sunny - 19 C
Mon 7/1 - Sunny - 19 C
Tues 8/11 - Rainy - 18 C
Wed 9/11 - TRAPPED IN WEATHERSTATION - 10 C
11:58
WeatherApp
Fri Oct 30 - Clouds - From 22°C to 17°C
Sat Oct 31 - Clouds - From 21°C to 15°C
Sun Nov 01 - Clear - From 22°C to 16°C
Mon Nov 02 - Clouds - From 21°C to 16°C
Tue Nov 03 - Clouds - From 21°C to 16°C
Wed Nov 04 - Clear - From 22°C to 16°C
Thu Nov 05 - Clouds - From 19°C to 15°C

Add Voice Recognition Functionality



We need a new option in the menu to initiate speech recognition



Add new option “Ask me!”

```
<?xml version="1.0" encoding="utf-8"?>
<menu
xmlns:android="http://schemas.android.com/apk/res/android"
  >
  <item android:id="@+id/action_refresh"
      android:title="Refresh"
      app:showAsAction="never" />
  <item android:id="@+id/voice_commands"
      android:title="Ask me!"  />
</menu>
```

SecondFragment.java

```java
package com.example.weatherapp;

import ...

public class SecondFragment extends Fragment {

    public static ArrayAdapter<String> mForecastAdapter;

    @Override
    public View onCreateView(
            LayoutInflater inflater, ViewGroup container,
            Bundle savedInstanceState
    ){...}

    @Override
    public void onCreateOptionsMenu(Menu menu, MenuInflater inflater) {...}

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        //...
        int id = item.getItemId();
        if (id == R.id.action_refresh) {
            FetchWeatherTask weatherTask = new FetchWeatherTask();
            weatherTask.execute("0000");
            return true;
        }
        return super.onOptionsItemSelected(item);
    }



}
```

We also need to support this option in our code

SecondFragment.java

```java
package com.example.weatherapp;

import ...

public class SecondFragment extends Fragment {

    public static ArrayAdapter<String> mForecastAdapter;

    @Override
    public View onCreateView(
            LayoutInflater inflater, ViewGroup container,
            Bundle savedInstanceState
    ){...}

    @Override
    public void onCreateOptionsMenu(Menu menu, MenuInflater inflater) {...}

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        //...
        int id = item.getItemId();
        if (id == R.id.action_refresh) {
            FetchWeatherTask weatherTask = new FetchWeatherTask();
            weatherTask.execute("0000");
            return true;
        }
        else if (id == R.id.voice_commands) {
            // Start voice recognition
            return true;

        }

        return super.onOptionsItemSelected(item);
    }



}
```

We also need to support this option in our code

SecondFragment.java

```java
public class SecondFragment extends Fragment {

    public static ArrayAdapter<String> mForecastAdapter;

    @Override
    public View onCreateView(
            LayoutInflater inflater, ViewGroup container,
            Bundle savedInstanceState
    ){...}

    @Override
    public void onCreateOptionsMenu(Menu menu, MenuInflater inflater) {...}

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        //...
        int id = item.getItemId();
        if (id == R.id.action_refresh) {
            FetchWeatherTask weatherTask = new FetchWeatherTask();
            weatherTask.execute("0000");
            return true;
        }
        else if (id == R.id.voice_commands) {
            // Start voice recognition
            startSpeechRecognizer();
            return true;

        }

        return super.onOptionsItemSelected(item);
    }

    private void startSpeechRecognizer() {
        int REQUEST_SPEECH_RECOGNIZER = 3000;
        Intent intent = new Intent
                (RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
        intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL, value: "en-US");
        intent.putExtra(RecognizerIntent.EXTRA_PROMPT, value: "City of Interest?");
        startActivityForResult(intent, REQUEST_SPEECH_RECOGNIZER);
    }
```

Add new function to initiate speech recognition using API

```
@Override
public boolean onMenuItemSelected(@NonNull MenuItem item) {
   // Handle menu item clicks
   if (item.getItemId() == R.id.action_refresh) {
       FetchWeatherForecast weatherTask = new FetchWeatherForecast();
       double city_lat =38.04;
       double city_long=23.54;
       weatherTask.executeOnExecutor(AsyncTask.THREAD_POOL_EXECUTOR, Double.toString(city_lat),
Double.toString(city_long));
       return true; // Indicate that the event was handled
   }
   else if (item.getItemId() == R.id.voice_commands) {
   // Start voice recognition
       startSpeechRecognizer();
       return true;

   }
   return false; // Let the system handle other menu actions
}
}, getViewLifecycleOwner(), Lifecycle.State.RESUMED); // Ensure the menu is only active when the fragment is visible
```

```
private void startSpeechRecognizer() {
  int REQUEST_SPEECH_RECOGNIZER = 3000;
  Intent intent = new Intent
        (RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
  intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL, "en-US");
  intent.putExtra(RecognizerIntent.EXTRA_PROMPT,"City of Interest?");
  startActivityForResult(intent, REQUEST_SPEECH_RECOGNIZER);
}
```

SecondFragment.java

```java
public class SecondFragment extends Fragment {

    public static ArrayAdapter<String> mForecastAdapter;
    public static String mAnswer;

    @Override
    public View onCreateView(
            LayoutInflater inflater, ViewGroup container,
            Bundle savedInstanceState
    ){...}

    @Override
    public void onCreateOptionsMenu(Menu menu, MenuInflater inflater) {...}

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {...}

    private void startSpeechRecognizer() {...}

    @Override
    public void onActivityResult(int requestCode, int resultCode,
                                 Intent data) {
        int REQUEST_SPEECH_RECOGNIZER = 3000;
        super.onActivityResult(requestCode, resultCode, data);
        Log.i( tag: "DEMO-REQUESTCODE", Integer.toString(requestCode));
        Log.i( tag: "DEMO-RESULTCODE", Integer.toString(resultCode));
        if (requestCode == REQUEST_SPEECH_RECOGNIZER && resultCode == Activity.RESULT_OK && data != null){
            ArrayList<String> text = data
                    .getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
            mAnswer = text.get(0);
            Log.i( tag: "DEMO-ANSWER", text.get(0));
            FetchWeatherTask weatherTask = new FetchWeatherTask();
            int city_id = 0;
            switch (mAnswer.toLowerCase()){...}
            Log.i( tag: "DEMO-CITY_ID", Integer.toString(city_id));
            weatherTask.execute(Integer.toString(city_id));
        }
        else{...}
    }
```

Handle transcription of speech

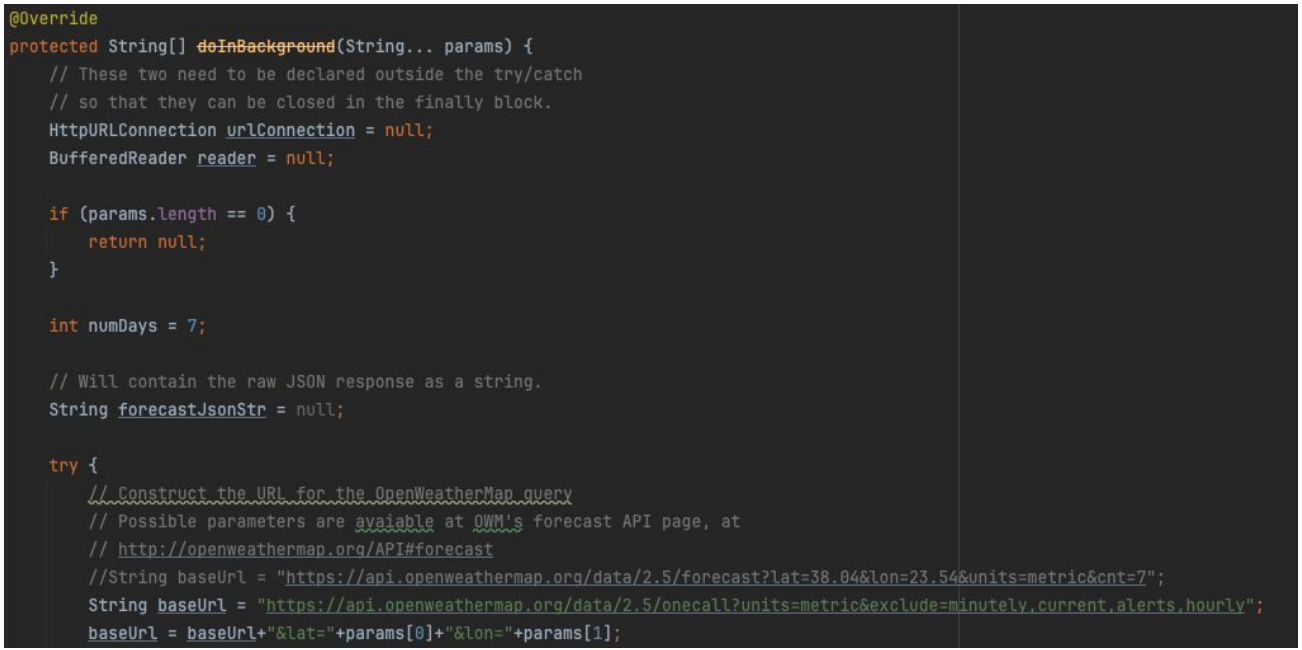
SecondFragment.java

```
    @Override
    public void onActivityResult(int requestCode, int resultCode,
                                 Intent data) {
        int REQUEST_SPEECH_RECOGNIZER = 3000;
        super.onActivityResult(requestCode, resultCode, data);
        Log.i( tag: "DEMO-REQUESTCODE", Integer.toString(requestCode));
        Log.i( tag: "DEMO-RESULTCODE", Integer.toString(resultCode));
        if (requestCode == REQUEST_SPEECH_RECOGNIZER && resultCode == Activity.RESULT_OK && data != null){
            ArrayList<String> text = data
                    .getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
            mAnswer = text.get(0);
            Log.i( tag: "DEMO-ANSWER", text.get(0));
            FetchWeatherTask weatherTask = new FetchWeatherTask();
            int city_id = 0;
            switch (mAnswer.toLowerCase()){
                case "athens" : city_id = 264371; break;
                case "london" : city_id = 2643743; break;
                case "paris"  : city_id = 2988507; break;
                default: city_id = 0;
            }
            Log.i( tag: "DEMO-CITY_ID", Integer.toString(city_id));
            weatherTask.execute(Integer.toString(city_id));
        }
        else{
            System.out.println("Recognizer API error");
        }
    }



}
```

Naive control for voice commands (Support 3 cities)

```
@Override
public void onActivityResult(int requestCode, int resultCode,
                             Intent data) {
  int REQUEST_SPEECH_RECOGNIZER = 3000;
  super.onActivityResult(requestCode, resultCode, data);  Log.i("DEMO-REQUESTCODE",
  Integer.toString(requestCode));  Log.i("DEMO-RESULTCODE", Integer.toString(resultCode));
  if (requestCode == REQUEST_SPEECH_RECOGNIZER && resultCode == Activity.RESULT_OK && data != null){
        ArrayList<String> text = data
              .getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
        mAnswer = text.get(0);
        Log.i("DEMO-ANSWER", text.get(0));
        FetchWeatherForecast weatherTask = new FetchWeatherForecast();
        double city_lat  = 0;
        double city_long  = 0;
        switch (mAnswer.toLowerCase()){
          case "athens"      : city_lat = 37.98; city_long = 23.73; break;
          case "elefsina"    : city_lat = 38.04; city_long = 23.54; break;
          case "london"      : city_lat = 51.51; city_long =  0.13; break;
          case "paris"       : city_lat = 48.86; city_long =  2.35; break;
          case "αθήνα"       : city_lat = 37.98; city_long = 23.73; break;
          case "ελευσίνα"    : city_lat = 38.04; city_long = 23.54; break;
          case "λονδίνο"     : city_lat = 51.51; city_long =  0.13; break;
          case "παρίσι"      : city_lat = 48.86; city_long =  2.35; break;
          default            : city_lat = 38.04; city_long = 23.54;
        }
        Log.i("LAT",    Double.toString(city_lat));
        Log.i("LONG",   Double.toString(city_long));
        weatherTask.executeOnExecutor(AsyncTask.THREAD_POOL_EXECUTOR,
        Double.toString(city_lat), Double.toString(city_long));
  }
  else{
     System.out.println("Recognizer API error");
  }
}
```

```
import android.content.Intent;
import android.speech.RecognizerIntent;
import android.app.Activity;
import android.util.Log;

public static String mAnswer;
```

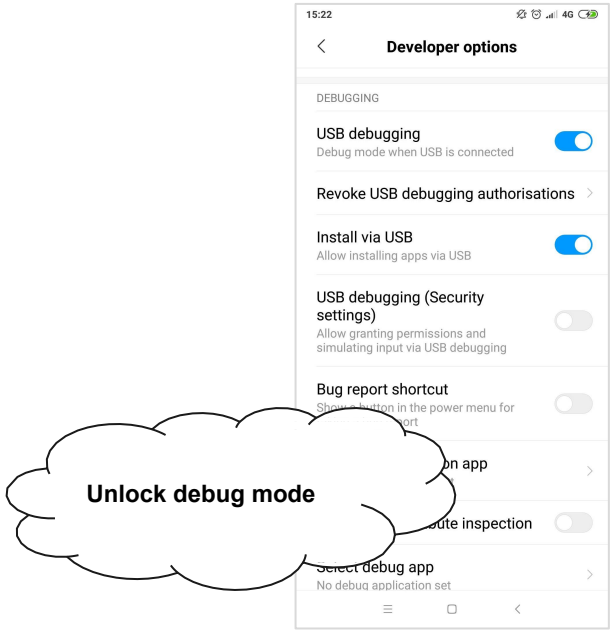


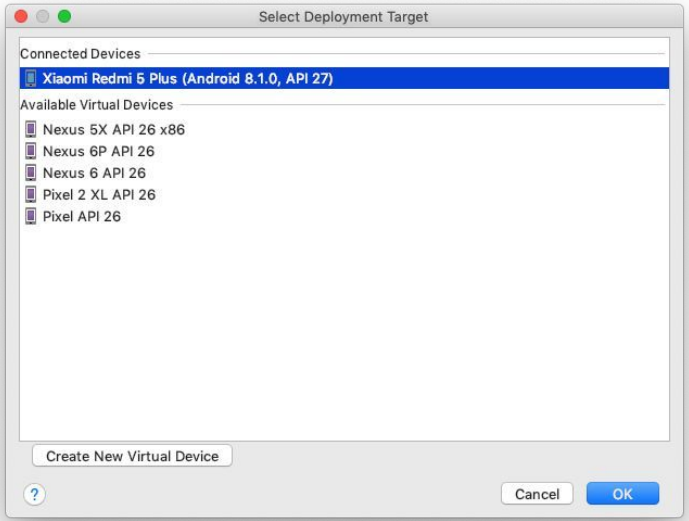
```java
@Override
protected String[] doInBackground(String... params) {
    // These two need to be declared outside the try/catch
    // so that they can be closed in the finally block.
    HttpURLConnection urlConnection = null;
    BufferedReader reader = null;

    if (params.length == 0) {
        return null;
    }

    int numDays = 7;

    // Will contain the raw JSON response as a string.
    String forecastJsonStr = null;

    try {
        // Construct the URL for the OpenWeatherMap query
        // Possible parameters are avaiable at OWM's forecast API page, at
        // http://openweathermap.org/API#forecast
        //String baseUrl = "https://api.openweathermap.org/data/2.5/forecast?lat=38.04&lon=23.54&units=metric&cnt=7";
        String baseUrl = "https://api.openweathermap.org/data/2.5/onecall?units=metric&exclude=minutely,current,alerts,hourly";
        baseUrl = baseUrl+"&lat="+params[0]+"&lon="+params[1];
```

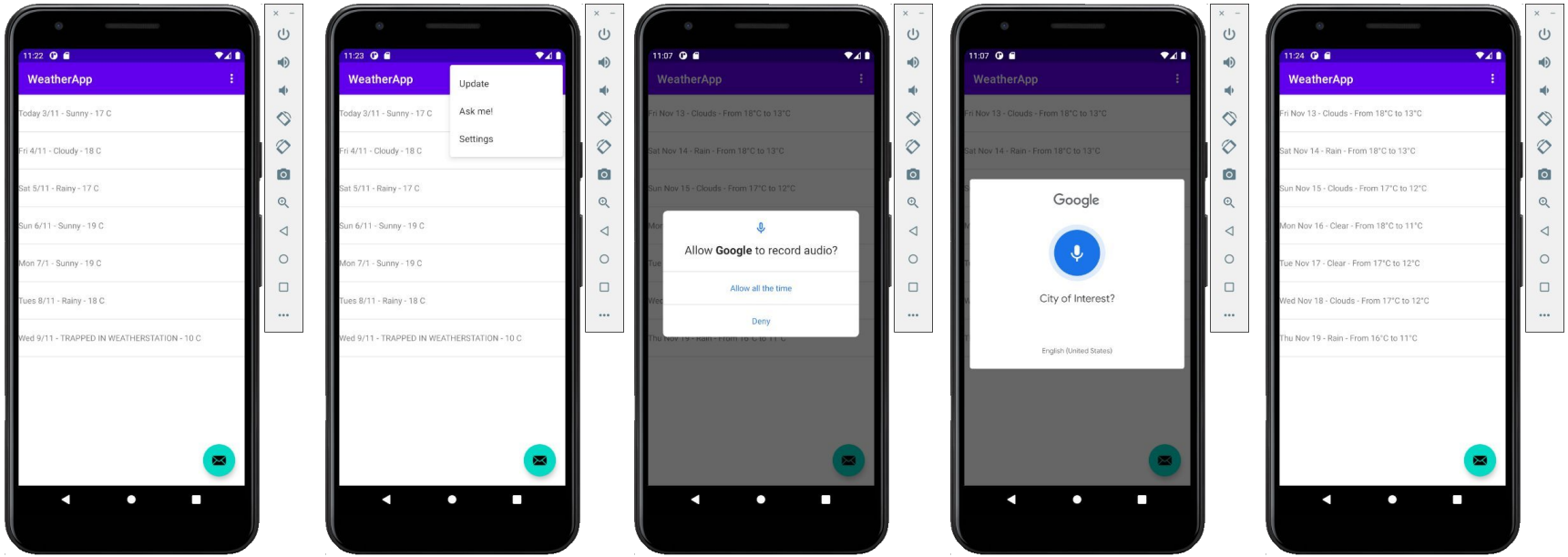
Use input params in FetchWeatherForecast Task

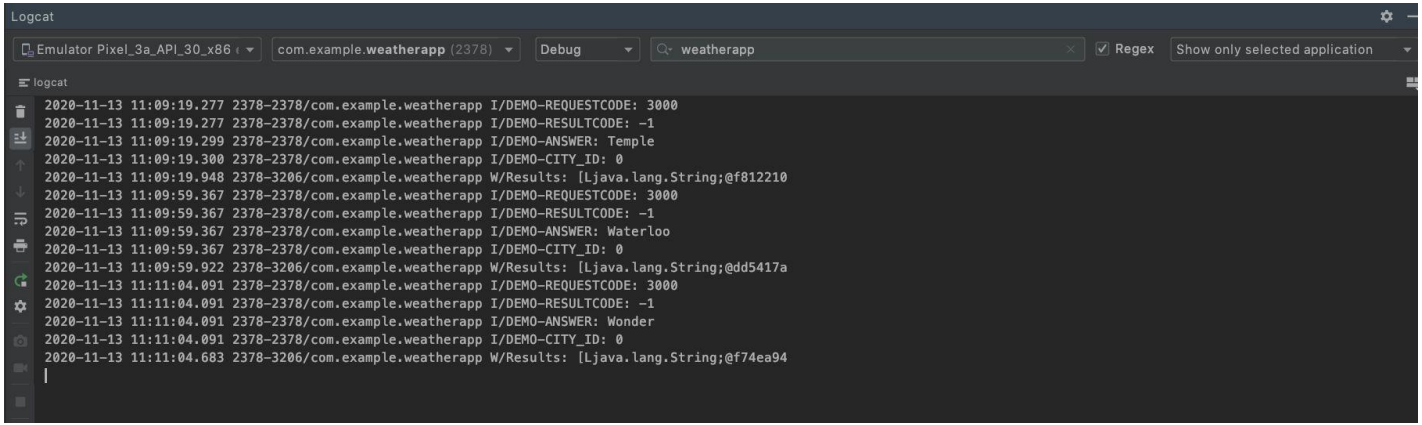
```
String baseUrl =
"https://api.openweathermap.org/data/2.5/forecast/daily?&units=metric&exclude=minute
ly,current,alerts,hourly";
baseUrl = baseUrl+"&lat="+params[0]+"&lon="+params[1];
```

Deploy Application on real device


15:22
4G
Developer options
DEBUGGING
USB debugging
Debug mode when USB is connected
Revoke USB debugging authorisations
Install via USB
Allow installing apps via USB
USB debugging (Security settings)
Allow granting permissions and simulating input via USB debugging
Bug report shortcut
Unlock debug mode
No debug application set


Select Deployment Target
Connected Devices
Xiaomi Redmi 5 Plus (Android 8.1.0, API 27)
Available Virtual Devices
Nexus 5X API 26 x86
Nexus 6P API 26
Nexus 6 API 26
Pixel 2 XL API 26
Pixel API 26
Create New Virtual Device
Cancel
OK


11:22
WeatherApp
Today 3/11 - Sunny - 17 C
Fri 4/11 - Cloudy - 18 C
Sat 5/11 - Rainy - 17 C
Sun 6/11 - Sunny - 19 C
Mon 7/1 - Sunny - 19 C
Tues 8/11 - Rainy - 18 C
Wed 9/11 - TRAPPED IN WEATHERSTATION - 10 C
11:23
WeatherApp
Update
Ask me!
Settings
Today 3/11 - Sunny - 17 C
Fri 4/11 - Cloudy - 18 C
Sat 5/11 - Rainy - 17 C
Sun 6/11 - Sunny - 19 C
Mon 7/1 - Sunny - 19 C
Tues 8/11 - Rainy - 18 C
Wed 9/11 - TRAPPED IN WEATHERSTATION - 10 C
11:07
WeatherApp
Allow Google to record audio?
Allow all the time
Deny
11:07
WeatherApp
Google
City of Interest?
English (United States)
11:24
WeatherApp
Fri Nov 13 - Clouds - From 18°C to 13°C
Sat Nov 14 - Rain - From 18°C to 13°C
Sun Nov 15 - Clouds - From 17°C to 12°C
Mon Nov 16 - Clear - From 18°C to 11°C
Tue Nov 17 - Clear - From 17°C to 12°C
Wed Nov 18 - Clouds - From 17°C to 12°C
Thu Nov 19 - Rain - From 16°C to 11°C



Logging of interaction through voice commands

Review of Lab 3 - Android Application Development

- Catch up with Lab 1-2
- Add Voice Recognition functionality
  - Rename Activity
  - Make property publicly available
  - Update Business Logic to use the Asynchronous task - Involve “Ask me!” Option
  - Demonstrate updates

Ασκήσεις μελέτης Β2

Άσκηση Β2.2

α) Χρησιμοποιώντας τις παρακάτω προτάσεις ως (μικροσκοπικό) σώμα κειμένων εκπαίδευσης:
<start> he plays football
<start> he plays cricket
<start> she enjoys good football
<start> she plays good music
<start> he prays to god
<start> please buy me the other ball
<start> he pleases the other players by playing good football

εκτιμήστε τις πιθανότητες $P(t_1^4)$ που θα επέστρεφε ένα γλωσσικό μοντέλο **διγραμμάτων** με **εξομάλυνση Laplace** για κάθε μία από τις δύο παρακάτω προτάσεις

t_1^4 :<start> he please god football
t_1^4 : <start> he plays good football

Υποθέστε ότι το λεξιλόγιο V περιέχει όλες τις λέξεις του σώματος κειμένων (εξαιρώντας το <start>), οπότε $|V|$ = 21. Δείξτε λεπτομερώς τους υπολογισμούς σας, χωρίς να εκτελέσετε τις τελικές αριθμητικές πράξεις.

- P(<start>, he, please, god, football) = P(he | <start>) P(please | he) P(god | please) P(football | god)
  P(he | <start>) = (4+1)/(7+21)
  P(please | he) = (0+1)/(4+21)
  P(god | please) = (0+1)/(1+21)
  P(football | god) = (0+1)/(1+21)

- P(<start>, he, plays, good, football) = P(he | <start>) P(plays | he) P(good | plays) P(football | good)
  P(he | <start>) = (4+1)/(7+21)
  P(plays | he) = (2+1)/(4+21)
  P(good | plays) = (1+1)/(3+21)
  P(football | good) = (2+1)/(3+21)

Σημείωση: Στην πράξη αποφεύγουμε διαδοχικούς πολλαπλασιασμούς πιθανοτήτων, υπολογίζουμε συνήθως τον λογάριθμο της πιθανότητας μιας ακολουθίας λέξεων, δηλαδή θα υπολογίζαμε το log P(<start>, he, please, god, football), αντί του P(<start>, he, please, god, football), οπότε θα καταλήγαμε στο παρακάτω άθροισμα τεσσάρων λογαρίθμων, αντί του παραπάνω γινομένου τεσσάρων πιθανοτήτων:
log[(4+1)/(7+21)] + log[(0+1)/(4+21)] + log[(0+1)/(1+21)] + log[(0+1)/(1+21)]

β) Υποθέστε ότι ένας χρήστης έγραψε στο πληκτρολόγιο του κινητού του την παρακάτω ακολουθία λέξεων

w_1^4: <start> he pls gd ftball

Εκτιμήστε τις πιθανότητες $P(t_1^4 | w_1^4)$ των δύο υποθέσεων (ακολουθιών λέξεων που ίσως ήθελε να γράψει) t_1^4 του σκέλους (α), χρησιμοποιώντας ένα μοντέλο θορυβώδους καναλιού (βλ. σχετικές διαφάνειες) και το γλωσσικό μοντέλο διγραμμάτων του σκέλους (α).

t_1^4 :<start> he please god football
t_1^4 : <start> he plays good football

$$\hat{t}_1^k = \arg\max_{t_1^k} P(t_1^k \mid w_1^k) = \arg\max_{t_1^k} \frac{P(t_1^k) \cdot P(w_1^k \mid t_1^k)}{\cancel{P(w_1^k)}}$$

$$P(w_1^k \mid t_1^k) \simeq \prod_{i=1}^{k} P(w_i \mid t_i)$$

Θεωρήστε ότι $P(w_i | t_i) \cong 1/(LD(w_i , t_i)+1)$, όπου $LD(w_i , t_i)$ η απόσταση Levenshtein από τη λέξη w_i στην t_i. Δείξτε λεπτομερώς τους υπολογισμούς σας, χωρίς να εκτελέσετε τις τελικές αριθμητικές πράξεις και χω να υπολογίσετε τις αποστάσεις Levenshtein.

Χρησιμοποιώντας το θορυβώδες κανάλι των διαφανειών της διάλεξης, έχουμε:
$P(t_1^4 | w_1^4) = P(t_1^4)\, P(w_1^4 | t_1^4) / P(w_1^4)$

Για t_1^4 = <start> he please god football:
$P(t_1^4)$ = P(<start>, he, please, god, football)
$P(w_1^4 | t_1^4) / P(w_1^4)$ = P(<start> he pls gd ftball | <start>, he, please, god, football) / $P(w_1^4)$

Η πιθανότητα P(<start>, he, please, god, football) εκτιμάται από το γλωσσικό μοντέλο, όπως στο σκέλος (α).
Χρησιμοποιώντας την προσέγγιση $P(w_i | t_i) \cong 1/(LD(w_i, t_i)+1)$ της εκφώνησης (βλ. και διαφάνειες), η πιθανότητα $P(w_1^4 | t_1^4)$ = P(<start> he pls gd ftball | <start>, he, please, god, football) γίνεται:

P(he, he) P(pls, please) P(gd, god) P(ftball, football) =
1/(LD(he, he)+1) 1/(LD(pls, please)+1) 1/(LD(gd, god)+1) 1/(LD(ftball, football)+1)

Η πιθανότητα $P(w_1^4)$ δεν χρειάζεται να εκτιμηθεί, γιατί είναι ίδια και για τις δύο υποθέσεις t_1^4 = <start> he please god football και t_1^4 = <start> he plays good football.

Ομοίως εκτιμούμε την πιθανότητα $P(t_1^4 | w_1^4)$ για την υπόθεση t_1^4 = <start> he plays good football.
Επιλέγουμε τελικά την υπόθεση t_1^4 με το μεγαλύτερο $P(t_1^4 | w_1^4)$.

Σημείωση: Και πάλι στην πράξη θα υπολογίζαμε τον λογάριθμο κάθε γινομένου πιθανοτήτων, οπότε θα καταλήγαμε σε αθροίσματα λογαρίθμων πιθανοτήτων, αντί γινόμενα πιθανοτήτων.

γ) Εξηγήστε αναλυτικά πώς θα γινόταν η αποκωδικοποίηση με beam search (διαφάνειες «Beam search decoder»), αν ο χρήστης γράψει στο πληκτρολόγιο την ακολουθία λέξεων w_1^4 του σκέλους (β).

w_1^4: <start> he pls gd ftball

Χρησιμοποιούμε πάλι το γλωσσικό μοντέλο διγραμμάτων λέξεων του
σκέλους (α), εκπαιδευμένο στο μικροσκοπικό σώμα εκπαίδευσης εκείνου του σκέλους,
<start> he plays football
<start> he plays cricket
<start> she enjoys good football
<start> she plays good music
<start> he prays to god
<start> please buy me the other ball
<start> he pleases the other players by playing good football

μαζί με το μοντέλο θορυβώδους καναλιού του σκέλους (β).
Θεωρήστε ότι το **πλέγμα (lattice)** αναζήτησης είναι το ακόλουθο, δηλαδή περιλαμβάνει **4 κοντινές** (κατά απόσταση διόρθωσης) υποψήφιες σωστές λέξεις (του λεξικού), για κάθε λέξη w_i που έχει γράψει ο χρήστης.
Σε κάθε βήμα του beam search, κρατάμε τα b **= 2 καλύτερα μονοπάτια**.

k = 0

k = 1
w_1= he

k = 2
w_2= pls

k = 3
w_3= gd

k = 4
w_4= ftball

$P(t_1^1)\ P(w_1^1 \mid t_1^1)$

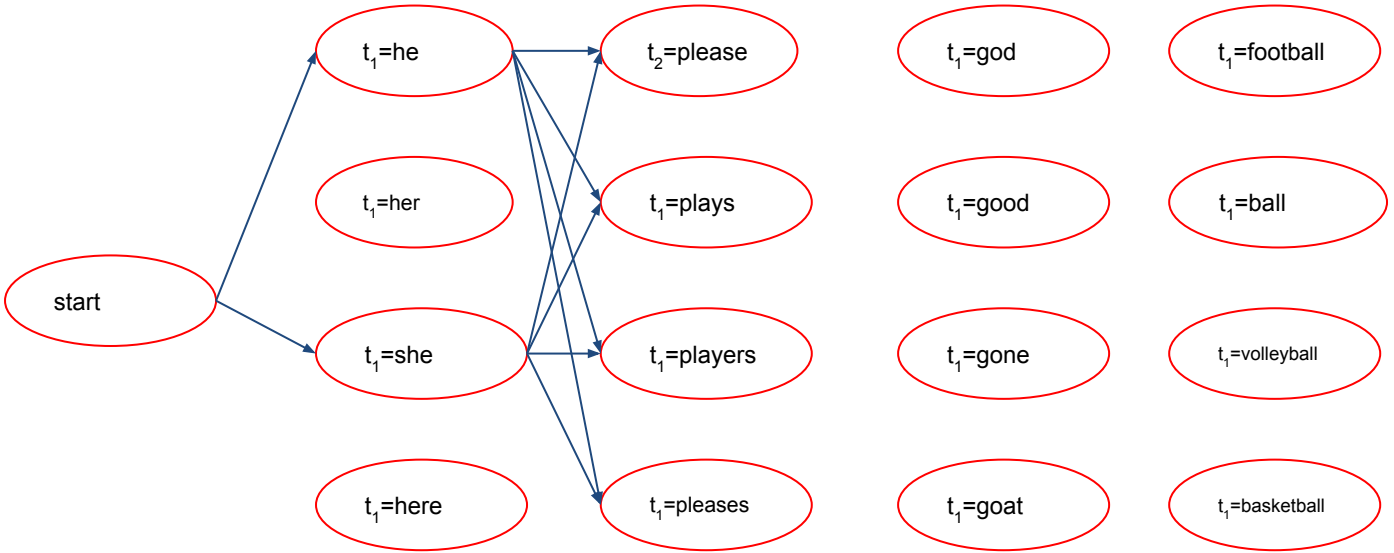
k = 0	k = 1	k = 2	k = 3	k = 4
	w_1= he	w_2= pls	w_3= gd	w_4= ftball

<start> he plays football
<start> he plays cricket
<start> she enjoys good football
<start> she plays good music
<start> he prays to god
<start> please buy me the other ball
<start> he pleases the other players by playing good football

Για k= 1

<start, he>: P(he|start) P(he|he) = (4+1)/(7+21) 1/(0+1) = 5/28 = **0.179** **
<start, her>: P(her|start) P(he|her) = (0+1)/(7+21) 1/(1+1) = 1/28 1/2 = 0.018
<start, she>: P(she|start) P(he|she) = (2+1)/(7+21) 1/(1+1) = 3/28 1/2 = **0.054** **
<start, here>: P(here|start) P(he|here) = (0+1)/(7+21) 1/(2+1) = 1/28 1/3 = 0.012

$$P(t_1^2)\ P(w_1^2 \mid t_1^2)$$



k = 0

k = 1

w_1= he

k = 2

w_2= pls

k = 3

w_3= gd

k = 4

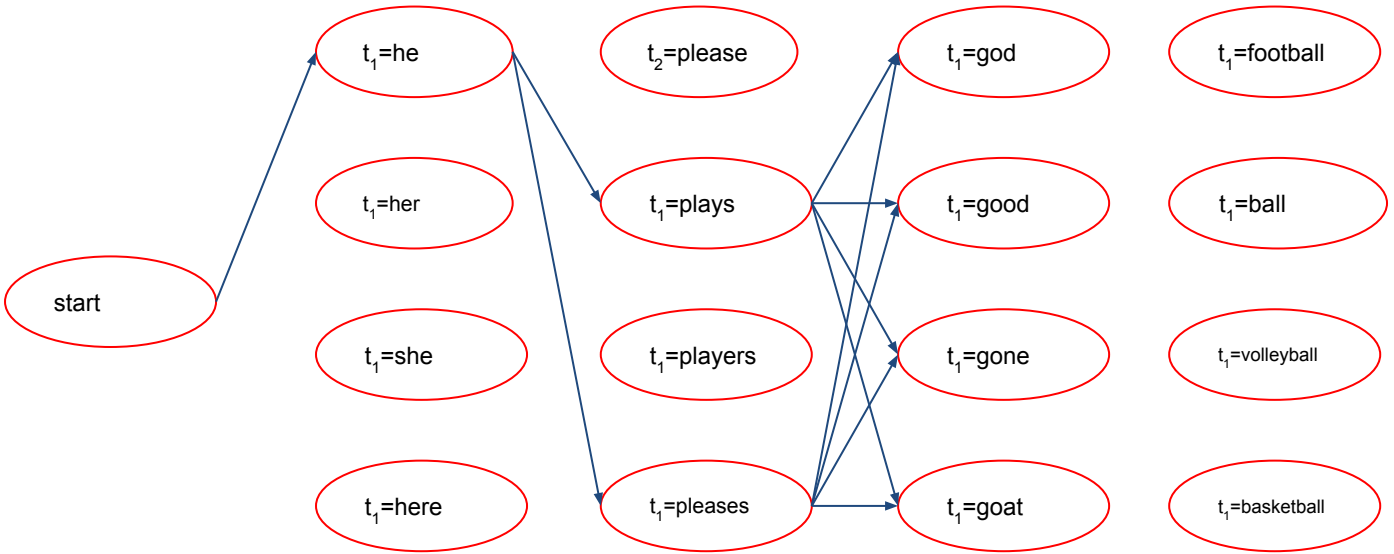
w_4= ftball

<start> he plays football
<start> he plays cricket
<start> she enjoys good football
<start> she plays good music
<start> he prays to god
<start> please buy me the other ball
<start> he pleases the other players by playing good football

Για k= 2

<start, he, please>: P(he|start) P(he|he) P(please|he) P(pls|please) = 5/28 (0+1)/(4+21) 1/(3+1) = 0.0018
<start, he, plays>: P(he|start) P(he|he) P(plays|he) P(pls|plays) = 5/28 (2+1)/(4+21) 1/(1+2) = **0.0071** **
<start, he, players>: P(he|start) P(he|he) P(players|he) P(pls|players) = 5/28 (0+1)/(4+21) 1/(4+1) = 0.0014
<start, he, pleases>: P(he|start) P(he|he) P(pleases|he) P(pls|pleases) = 5/28 (1+1)/(4+21) 1/(4+1) = **0.0029** **
<start, she, please>: P(she|start) P(he|she) P(please|she) P(pls|please) = 3/56 (0+1)/(2+21) 1/(3+1) = 0.0006
<start, she, plays>: P(she|start) P(he|she) P(plays|she) P(pls|plays) = 3/56 (1+1)/(2+21) 1/(2+1) = 0.0016
<start, she, players>: P(she|start) P(he|she) P(players|she) P(pls|players) = 3/56 (0+1)/(2+21) 1/(4+1) = 0.0005
<start, she, pleases>: P(she|start) P(he|she) P(pleases|she) P(pls|pleases) = 3/56 (0+1)/(2+21) 1/(4+1) = 0.0005

$P(t_1^3)\ P(w_1^3 |\ t_1^3)$



k = 0

k = 1
w_1= he

k = 2
w_2= pls

k = 3
w_3= gd

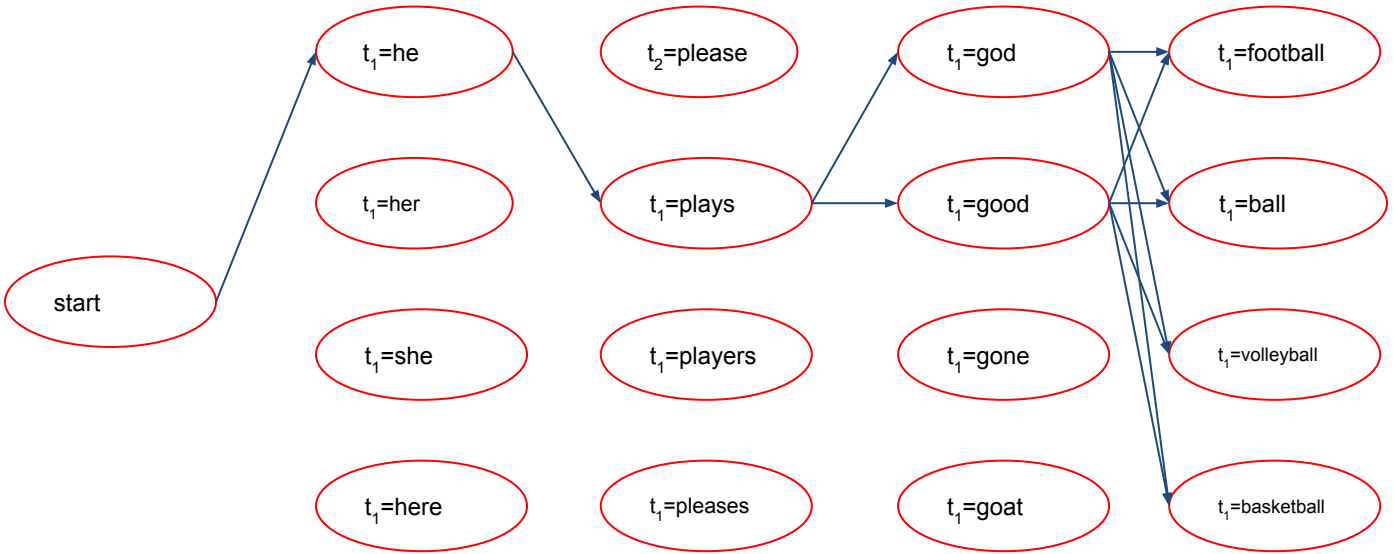
k = 4
w_4= ftball

<start> he plays football
<start> he plays cricket
<start> she enjoys good football
<start> she plays good music
<start> he prays to god
<start> please buy me the other ball
<start> he pleases the other players by playing good football

Για k= 3

<start, he, plays, god>: P(he|start) P(he|he) P(plays|he) P(pls|plays) P(god|plays) P(gd|god) = 0.0071 (0+1)/(3+21) 1/(1+1) = **0.00015** **
<start, he, plays, good>: P(he|start) P(he|he) P(plays|he) P(pls|plays) P(good|plays) P(gd|good) = 0.0071 (1+1)/(3+21) 1/(2+1) = **0.0002** **
<start, he, plays, gone>: P(he|start) P(he|he) P(plays|he) P(pls|plays) P(gone|plays) P(gd|gone) = 0.0071 (0+1)/(3+21) 1/(4+1) = 0.00006
<start, he, plays, goat>: P(he|start) P(he|he) P(plays|he) P(pls|plays) P(goat|plays) P(gd|goat) = 0.0071 (0+1)/(3+21) 1/(4+1) = 0.00006
<start, he, pleases, god>: P(he|start) P(he|he) P(pleases|he) P(pls|pleases) P(god|pleases) P(gd|god)= 0.0029 (0+1)/(1+21) 1/(1+1) = 0.00007
<start, he, pleases, good>: P(he|start) P(he|he) P(pleases|he) P(pls|pleases) P(good|pleases) P(gd|good) = 0.0029 (0+1)/(1+21) 1/(2+1) = 0.00004
<start, he, pleases, gone>: P(he|start) P(he|he) P(pleases|he) P(pls|pleases) P(gone|pleases) P(gd|gone) = 0.0029 (0+1)/(1+21) 1/(4+1) = 0.00003
<start, he, pleases, goat>: P(he|start) P(he|he) P(pleases|he) P(pls|pleases) P(goat|pleases) P(gd|goat) = 0.0029 (0+1)/(1+21) 1/(4+1) = 0.00003

$P(t_1^4)\ P(w_1^4 |\ t_1^4)$



k = 0

k = 1
w_1= he

k = 2
w_2= pls

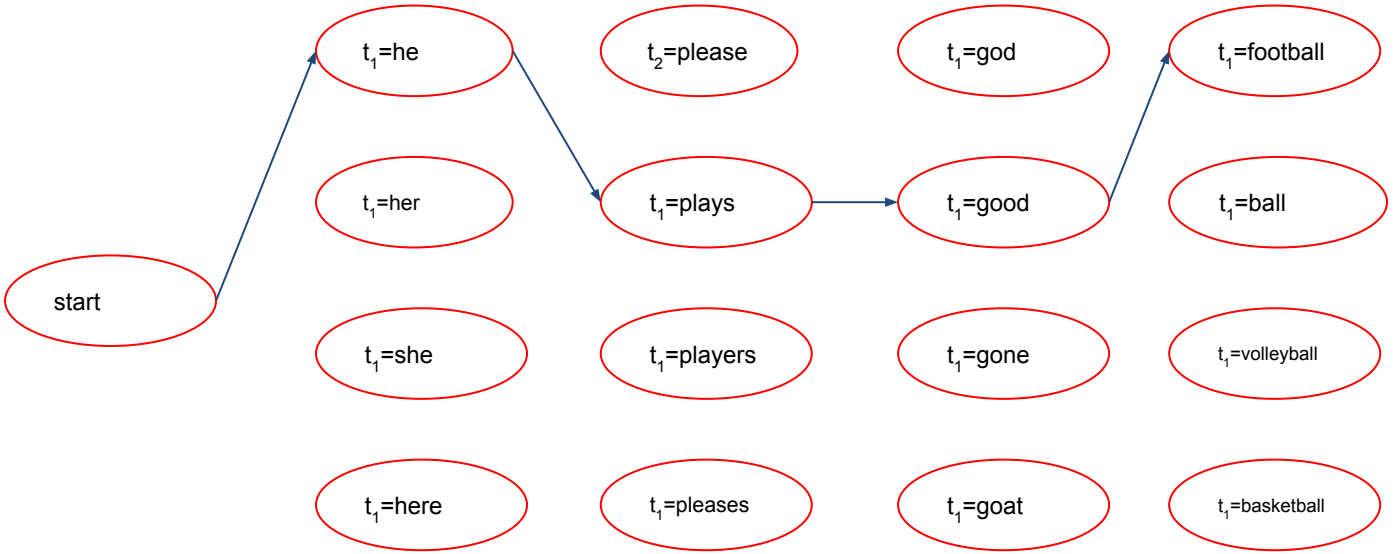
k = 3
w_3= gd

k = 4
w_4= ftball

<start> he plays football
<start> he plays cricket
<start> she enjoys good football
<start> she plays good music
<start> he prays to god
<start> please buy me the other ball
<start> he pleases the other players by playing good football

Για k= 3

<start, he, plays, god, football>: P(he|start) P(he|he) P(plays|he) P(pls|plays) P(god|plays) P(gd|god) P(football|god) P(ftball|football) = 0.00015 (0+1)/(1+21) 1/(2+1) = 0.0000023
<start, he, plays, god, ball>: P(he|start) P(he|he) P(plays|he) P(pls|plays) P(god|plays) P(gd|god) P(ball|god) P(ftball|ball)= 0.00015 (0+1)/(1+21) 1/(2+1) = 0.0000023
<start, he, plays, god, volleyball>: P(he|start) P(he|he) P(plays|he) P(pls|plays) P(god|plays) P(gd|god) P(volleyball|god) P(ftball|volleyball)= 0.00015 (0+1)/(1+21) 1/(8+1) = 0.00000076
<start, he, plays, god, basketball>: P(he|start) P(he|he) P(plays|he) P(pls|plays) P(god|plays) P(gd|god) P(basketball|god) P(ftball|basketball)= 0.00015 (0+1)/(1+21) 1/(6+1) = 0.00000097
<start, he, plays, good, football>: P(he|start) P(he|he) P(plays|he) P(pls|plays) P(good|plays) P(gd|good) P(football|good) P(ftball|football)= 0.0002 (2+1)/(3+21) 1/(2+1) = **0.0000083** **
<start, he, plays, good, ball>: P(he|start) P(he|he) P(plays|he) P(pls|plays) P(good|plays) P(gd|good) P(ball|good) P(ftball|ball)= 0.0002 (0+1)/(3+21) 1/(2+1) = 0.0000028
<start, he, plays, good, volleyball>: P(he|start) P(he|he) P(plays|he) P(pls|plays) P(good|plays) P(gd|good) P(volleyball|good) P(ftball|volleyball)= 0.0002 (0+1)/(3+21) 1/(8+1) = 0.00000093
<start, he, plays, good, basketball>: P(he|start) P(he|he) P(plays|he) P(pls|plays) P(good|plays) P(gd|good) P(basketball|good) P(ftball|basketball)= 0.0002 (0+1)/(3+21) 1/(6+1) = 0.0000012



k = 0

k = 1

w_1= he

k = 2

w_2= pls

k = 3

w_3= gd

k = 4

w_4= ftball