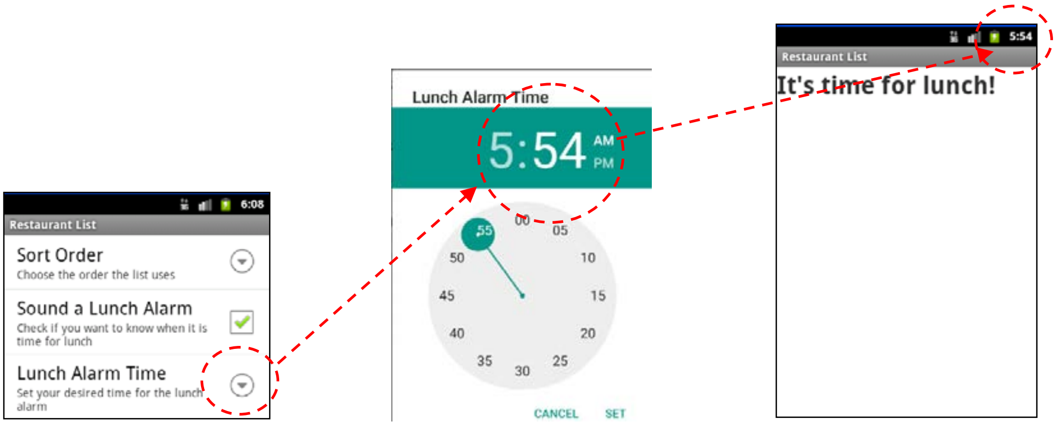
Practical 5: Alarm Managers & Notification

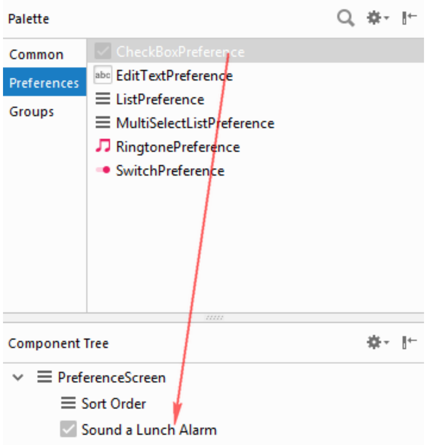
At the end of the session, you will learn how to make use of Broadcast Receiver to register an alarm service upon system boot completion and set notification at system notification

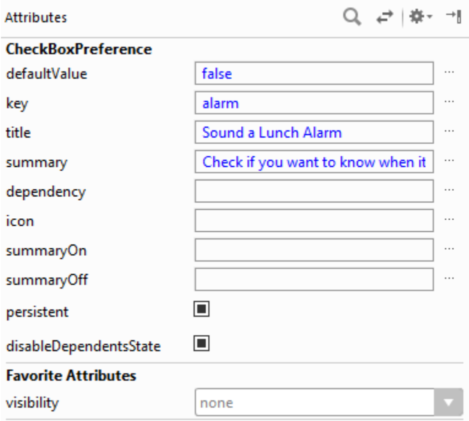
Part I – Binding to Alarm Manager

1. In this part of exercise, we will add an extra setting preference to allow user to set a lunch alarm alert



2. Create a new project with the following information:
   - **Application Name** : *Restaurant List*
   - **Company Domain** : *sp.com*
   - **Project Location** : *C:\MAD\AndroidStudioProjects\Lab5a or D:\MAD\AndroidStudioProjects\Lab5a*
   - **Minimum SDK** : *Android 5.0 (Lollipop)*
   - **Empty Activity name** : *RestaurantList*
   - **Layout name** : *main*
3. Close *RestaurantList.java* and *main.xml* files in the **Editor** pane
4. Open your **Windows File Explorer** and navigate to your **Android Studio** workspace where all you projects are created.
5. Copy ***AndroidManifest.xml*** file, **java** and **res** folders from **Lab4b\app\src\main** project folder and paste into **Lab5a\app\src\main** folder to overwrite the existing file and folder
6. To add the alarm checkbox and the time picker in **Setting** MENU, open the *preferences.xml* file from **res/xml** folder and add the following two preferences:
   - A **CheckBoxPreference** to enable or disable alarm :-
     Open *preferences.xml* in the Design mode, drag a "CheckBoxPreference" widget on to the "PreferenceScreen". Fill up the "CheckBoxPreference" attributes as shown.




Attributes

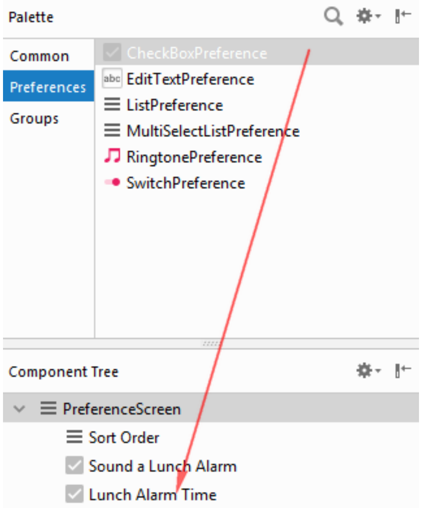
CheckBoxPreference

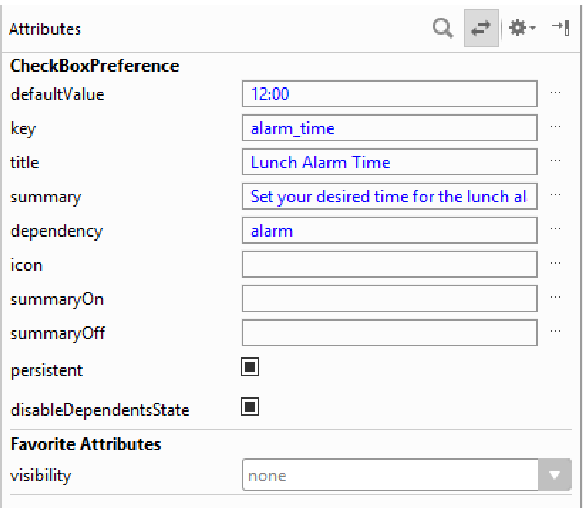
defaultValue	false
key	alarm
title	Sound a Lunch Alarm
summary	Check if you want to know when it
dependency	
icon	
summaryOn	
summaryOff	
persistent	■
disableDependentsState	■

Favorite Attributes

visibility	none

- A **TimePreference** (**DialogPreference** subclass) with Time Picker :-
  Open *preferences.xml* in the Design mode, drag a "CheckBoxPreference" widget on to the "PreferenceScreen". Fill up the "CheckBoxPreference" attributes as shown.




Attributes

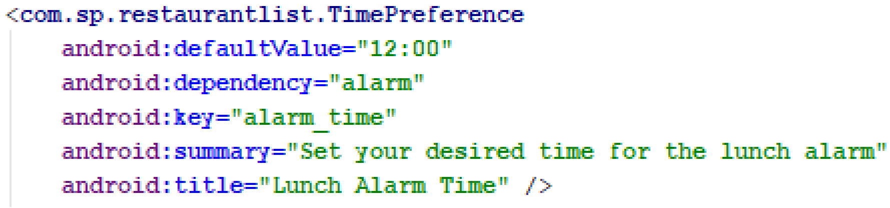
CheckBoxPreference

defaultValue	12:00
key	alarm_time
title	Lunch Alarm Time
summary	Set your desired time for the lunch al
dependency	alarm
icon	
summaryOn	
summaryOff	
persistent	■
disableDependentsState	■

Favorite Attributes

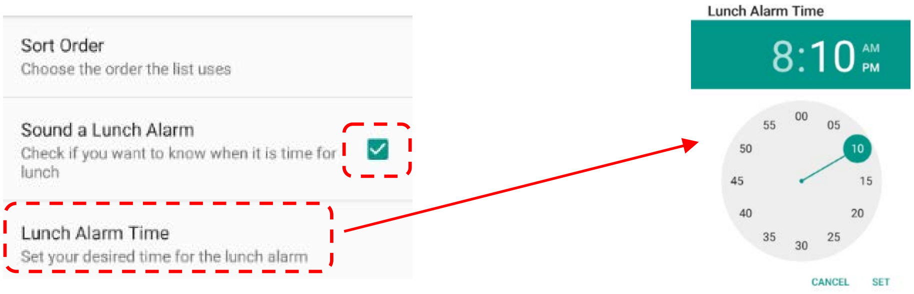
visibility	none

- Set *preferences.xml* in the Text mode, edit "CheckBoxPreference" to " `com.sp.restaurantlist.TimePreference`"


```
<com.sp.restaurantlist.TimePreference
    android:defaultValue="12:00"
    android:dependency="alarm"
    android:key="alarm_time"
    android:summary="Set your desired time for the lunch alarm"
    android:title="Lunch Alarm Time" />
```

7. Check that preferences.xml is as shown.

```
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen xmlns:tools="http://schemas.android.com/tools"
    xmlns:android="http://schemas.android.com/apk/res/android">

    <ListPreference
        android:defaultValue="1"
        android:dialogTitle="Choose a sort order"
        android:entries="@array/sort_names"
        android:entryValues="@array/sort_clauses"
        android:key="sort_order"
        android:summary="Choose the order the list uses"
        android:title="Sort Order" />
    <CheckBoxPreference
        android:defaultValue="false"
        android:key="alarm"
        android:summary="Check if you want to know when it is time for lunch"
        android:title="Sound a Lunch Alarm" />
    <com.sp.restaurantlist.TimePreference
        android:defaultValue="12:00"
        android:dependency="alarm"
        android:key="alarm_time"
        android:summary="Set your desired time for the lunch alarm"
        android:title="Lunch Alarm Time" />
</PreferenceScreen>
```



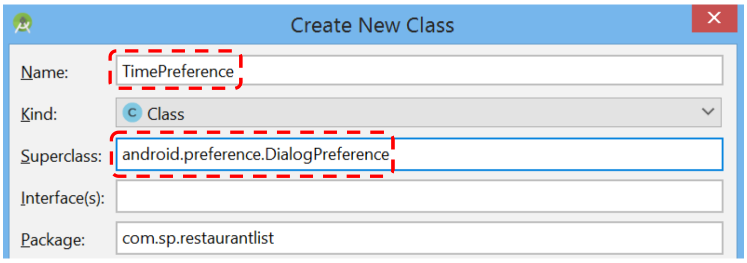
Sort Order
Choose the order the list uses

Sound a Lunch Alarm ☑
Check if you want to know when it is time for lunch

Lunch Alarm Time
Set your desired time for the lunch alarm

TimePreference.java – a **DialogPreference** with *TimePicker*

8. The pop-up time picker is actually implemented using **DialogPreference**. Add a new Java Class named *TimePreference.java* (**DialogPreference subclass**) to **java/com.sp.restaurantlist** folder



Create New Class

Name: TimePreference

Kind: Class

Superclass: android.preference.DialogPreference

Interface(s):

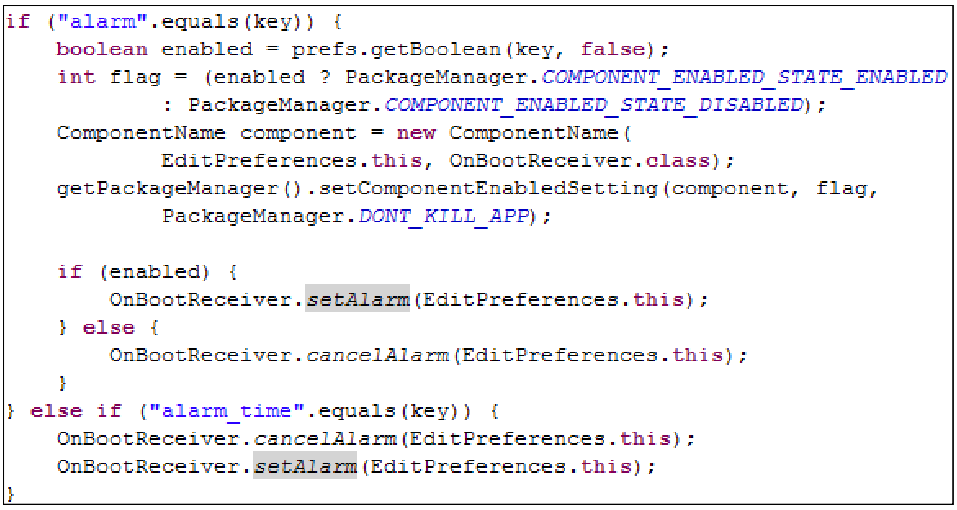
Package: com.sp.restaurantlist

9. The time set in the Time Picker will be stored to the “alarm_time” (Line 19 of *preferences.xml* file) through the `persistString` method (Line 53 of *TimePreference.java* file)
10. Update the file with the following content and save

```
1.     package com.sp.restaurantlist;
2.
3.     import android.content.Context;
4.     import android.content.res.TypedArray;
5.     import android.preference.DialogPreference;
6.     import android.util.AttributeSet;
7.     import android.view.View;
8.     import android.widget.TimePicker;
9.
10.    public class TimePreference extends DialogPreference {
11.        private int lastHour = 0;
12.        private int lastMinute = 0;
13.        private TimePicker picker = null;
14.
15.        public static int getHour(String time) {
16.            String[] pieces = time.split(":");
17.            return (Integer.parseInt(pieces[0]));
18.        }
19.
20.        public static int getMinute(String time) {
21.            String[] pieces = time.split(":");
22.            return (Integer.parseInt(pieces[1]));
23.        }
24.
25.        public TimePreference(Context context, AttributeSet attrs) {
26.            super(context, attrs);
27.            setPositiveButtonText("Set");
28.            setNegativeButtonText("Cancel");
29.        }
30.
31.        @Override
32.        protected View onCreateDialogView() {
33.            picker = new TimePicker(getContext());
34.            return (picker);
35.        }
36.
37.        @Override
```

```
    protected void onBindDialogView(View v) {
        super.onBindDialogView(v);
        picker.setCurrentHour(lastHour);
        picker.setCurrentMinute(lastMinute);
    }

    @Override
    protected void onDialogClosed(boolean positiveResult) {
        super.onDialogClosed(positiveResult);
        if (positiveResult) {
            lastHour = picker.getCurrentHour();
            lastMinute = picker.getCurrentMinute();

            String time = String.valueOf(lastHour) + ":"
                    + String.valueOf(lastMinute);
            if (callChangeListener(time)) {
                persistString(time);
            }
        }
    }

    @Override
    protected Object onGetDefaultValue(TypedArray a, int index) {
        return (a.getString(index));
    }

    @Override
    protected void onSetInitialValue(boolean restoreValue, Object defaultValue) {
        String time = null;

        if (restoreValue) {
            if (defaultValue == null) {
                time = getPersistedString("00:00");
            } else {
                time = getPersistedString(defaultValue.toString());
            }
        } else {
            time = defaultValue.toString();
        }`

        lastHour = getHour(time);
        lastMinute = getMinute(time);
    }
}
```

11. If the CheckBoxPreference is set, the *EditPreference.java* **PreferenceActivity** will turn on the **Alarm Manager** using setAlarm method provided in *OnBootReceiver.java* (**BroadcastReceiver** subclass). Otherwise, it will cancel the **Alarm Manager** using cancelAlarm method provided in *OnBootReceiver.java* **BroadcastReceiver**



```
if ("alarm".equals(key)) {
    boolean enabled = prefs.getBoolean(key, false);
    int flag = (enabled ? PackageManager.COMPONENT_ENABLED_STATE_ENABLED
            : PackageManager.COMPONENT_ENABLED_STATE_DISABLED);
    ComponentName component = new ComponentName(
            EditPreferences.this, OnBootReceiver.class);
    getPackageManager().setComponentEnabledSetting(component, flag,
            PackageManager.DONT_KILL_APP);

    if (enabled) {
        OnBootReceiver.setAlarm(EditPreferences.this);
    } else {
        OnBootReceiver.cancelAlarm(EditPreferences.this);
    }
} else if ("alarm_time".equals(key)) {
    OnBootReceiver.cancelAlarm(EditPreferences.this);
    OnBootReceiver.setAlarm(EditPreferences.this);
}
```

12. Update the *EditPreference.java* file with the following content

```
package com.sp.restaurantlist;

import android.content.ComponentName;
import android.content.SharedPreferences;
import android.content.pm.PackageManager;
import android.os.Bundle;
import android.preference.PreferenceActivity;
import android.preference.PreferenceManager;

public class EditPreferences extends PreferenceActivity {
    private SharedPreferences prefs = null;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        getFragmentManager().beginTransaction().replace(android.R.id.content, new
        MyPreferenceFragment()).commit();
    }

    public static class MyPreferenceFragment extends PreferenceFragment{
        @Override
        public void onCreate(final Bundle savedInstanceState){
            super.onCreate(savedInstanceState);
            addPreferencesFromResource(R.xml.preferences);
        }
    }

    @Override
    public void onResume() {
        super.onResume();
        prefs = PreferenceManager.getDefaultSharedPreferences(this);
        prefs.registerOnSharedPreferenceChangeListener(onChange);
    }

    @Override
    public void onPause() {
        prefs.unregisterOnSharedPreferenceChangeListener(onChange);
        super.onPause();
    }

    SharedPreferences.OnSharedPreferenceChangeListener onChange = new
SharedPreferences.OnSharedPreferenceChangeListener() {
        public void onSharedPreferenceChanged(SharedPreferences prefs,
                                              String key) {
            if ("alarm".equals(key)) {
                boolean enabled = prefs.getBoolean(key, false);
                int flag = (enabled ?
PackageManager.COMPONENT_ENABLED_STATE_ENABLED
                        : PackageManager.COMPONENT_ENABLED_STATE_DISABLED);
                ComponentName component = new ComponentName(
                        EditPreferences.this, OnBootReceiver.class);
                getPackageManager().setComponentEnabledSetting(component, flag,
                        PackageManager.DONT_KILL_APP);

                if (enabled) {
                    OnBootReceiver.setAlarm(EditPreferences.this);
                } else {
                    OnBootReceiver.cancelAlarm(EditPreferences.this);
                }
            } else if ("alarm_time".equals(key)) {
                OnBootReceiver.cancelAlarm(EditPreferences.this);
                OnBootReceiver.setAlarm(EditPreferences.this);
            }
        }
    };
}
```

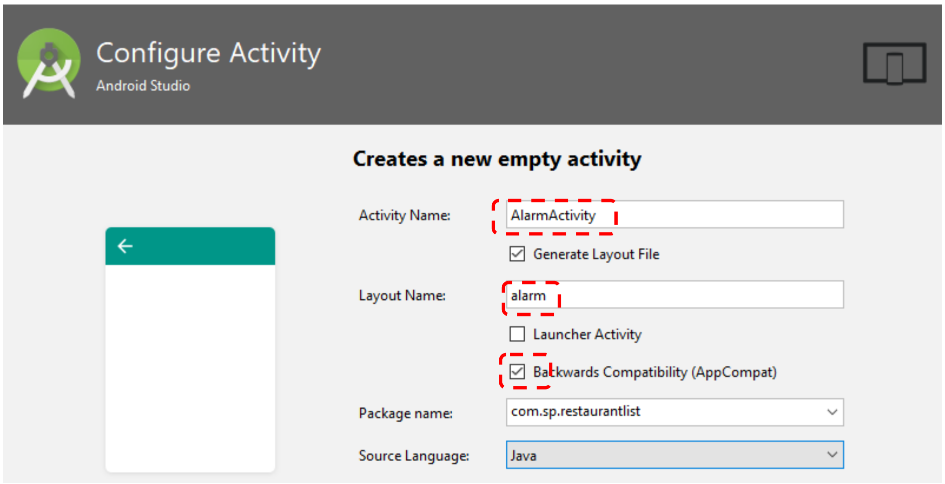
13. Setup a **Boot-Time Receiver** to allow **AlarmManager** to get control at boot time to return control to user when specified lunchtime arrives every day
14. Right click on **java/com.sp.restaurantlist** folder and select **New > Other > Broadcast Receiver**. Enter the file name as *OnBootReceiver*. The "real" work for a **BroadcastReceiver** is in the **onReceive()** method. The **PendingIntent** scheduled with **AlarmManager** should be invoked at the specified time each day, if user has enabled alarm

```java
package com.sp.restaurantlist;

import android.app.AlarmManager;
import android.app.PendingIntent;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.preference.PreferenceManager;

import java.util.Calendar;

public class OnBootReceiver extends BroadcastReceiver {
    public static void setAlarm(Context context) {
        AlarmManager mgr = (AlarmManager) context
                .getSystemService(Context.ALARM_SERVICE);
        Calendar cal = Calendar.getInstance();
        SharedPreferences prefs = PreferenceManager
                .getDefaultSharedPreferences(context);
        String time = prefs.getString("alarm_time", "12:00");

        cal.set(Calendar.HOUR_OF_DAY, TimePreference.getHour(time));
        cal.set(Calendar.MINUTE, TimePreference.getMinute(time));
        cal.set(Calendar.SECOND, 0);
        cal.set(Calendar.MILLISECOND, 0);

        if (cal.getTimeInMillis() < System.currentTimeMillis()) {
            cal.add(Calendar.DAY_OF_YEAR, 1);
        }

        mgr.setRepeating(AlarmManager.RTC_WAKEUP, cal.getTimeInMillis(),
                AlarmManager.INTERVAL_DAY, getPendingIntent(context));
    }

    public static void cancelAlarm(Context context) {
        AlarmManager mgr = (AlarmManager) context
                .getSystemService(Context.ALARM_SERVICE);
        mgr.cancel(getPendingIntent(context));
    }

    private static PendingIntent getPendingIntent(Context context) {
        Intent i = new Intent(context, OnAlarmReceiver.class);
        return (PendingIntent.getBroadcast(context, 0, i, 0));
    }

    @Override
    public void onReceive(Context context, Intent intent) {
        setAlarm(context);
    }
}
```

15. Right click on **java/com.sp.restaurantlist** folder and select **New > Other > Broadcast Receiver.** Create an Enter the file name as *OnAlarmReceiver*. Whenever the alarm is triggered, the **Alarm Manager** will alert the *OnAlarmReceiver* **BroadcastReceiver.** The receiver will do an *Explicit Intent* call to *AlarmActivity.java* to load the UI view with alarm.xml layout
16. Update the content and save

```java
package com.sp.restaurantlist;

import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;

public class OnAlarmReceiver extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        intent = new Intent(context, AlarmActivity.class);
        intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);

        context.startActivity(intent);
    }
}
```

17. Right click on the **java/com.sp.restaurantlist** folder, create a **New > Activity > Empty Activity** named *AlarmActivity* and layout name as *alarm*

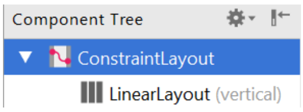


Configure Activity
Android Studio

Creates a new empty activity

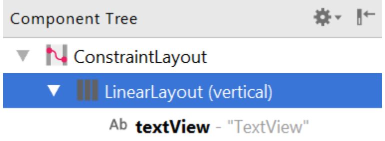
Activity Name: AlarmActivity
☑ Generate Layout File
Layout Name: alarm
□ Launcher Activity
☑ Backwards Compatibility (AppCompat)
Package name: com.sp.restaurantlist
Source Language: Java

18. At the **res/layout** folder, open *alarm.xml* file
19. At **Design Editor Palette**, drag a **LinearLayout (vertical)** widget into the **ConstraintLayout** as a child node



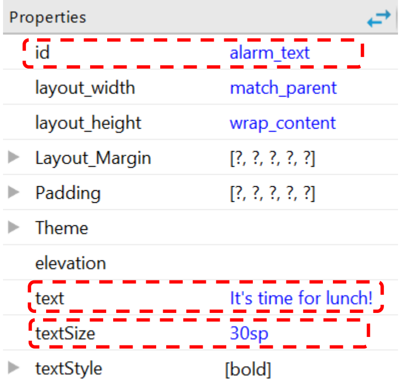
Component Tree
ConstraintLayout
LinearLayout (vertical)

20. Drag a **TextView** widget and drop into the **LinearLayout (vertical)** just added.



Component Tree
ConstraintLayout
LinearLayout (vertical)
Ab textView - "TextView"

21. Update the TextView widget with the following attributes



Properties

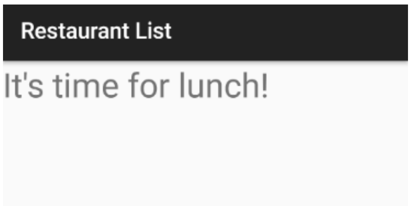
id	alarm_text
layout_width	match_parent
layout_height	wrap_content
Layout_Margin	[?, ?, ?, ?, ?]
Padding	[?, ?, ?, ?, ?]
Theme	
elevation	
text	It's time for lunch!
textSize	30sp
textStyle	[bold]

22. ***IMPORTANT**

All new activities and BroadcastReceiver added must be registered to the *AndroidManifest* as well as setting the necessary permission to allow Restaurant List app to have accessed to “Boot Completed” receiver. Check and add or edit the *AndroidManifest* as shown below.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    package="com.sp.restaurantlist">

    <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".RestaurantList">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".DetailForm" />
        <activity android:name=".EditPreferences" />
        <activity android:name=".AlarmActivity"></activity>

        <receiver
            android:name=".OnBootReceiver"
            android:enabled="false"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.BOOT_COMPLETED" />
            </intent-filter>
        </receiver>

        <receiver
            android:name=".OnAlarmReceiver"
            android:enabled="true"
            android:exported="true" />
    </application>

</manifest>
```

23. Run the *Lab5a* project. Set the alarm time to 1 minute from the current time shows on the emulator. You should see an alert screen pop up when it is lunch time.
24. When an alarm is triggered, a simple lunch alarm UI view to be shown on screen



Restaurant List

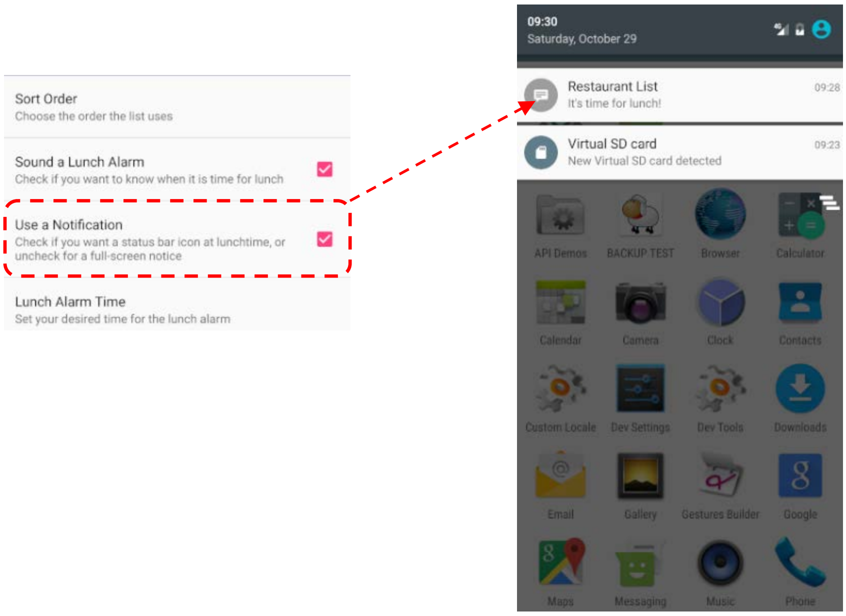
It's time for lunch!

25. Show your result to your lecturer when you have completed this section

Lecturer Signature : ____

Part II – Creating a System Notification

26. Instead of using a full-screen activity for lunch time alarm, a more subtle way of notification can be done. In this exercise, user can choose the style of lunch time alarm. User allows to set to be notified through Notifications which ties to **NotificationManager**



Sort Order
Choose the order the list uses

Sound a Lunch Alarm ☑
Check if you want to know when it is time for lunch

Use a Notification ☑
Check if you want a status bar icon at lunchtime, or uncheck for a full-screen notice

Lunch Alarm Time
Set your desired time for the lunch alarm

09:30
Saturday, October 29

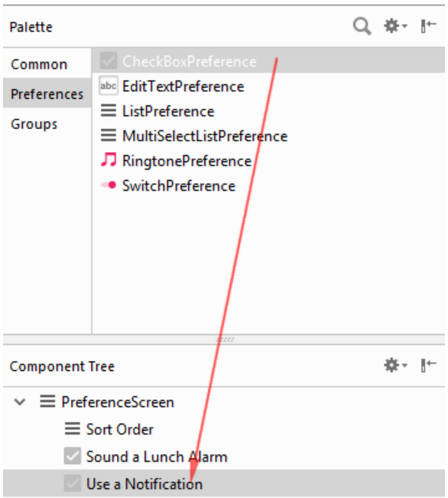
Restaurant List 09:28
It's time for lunch!

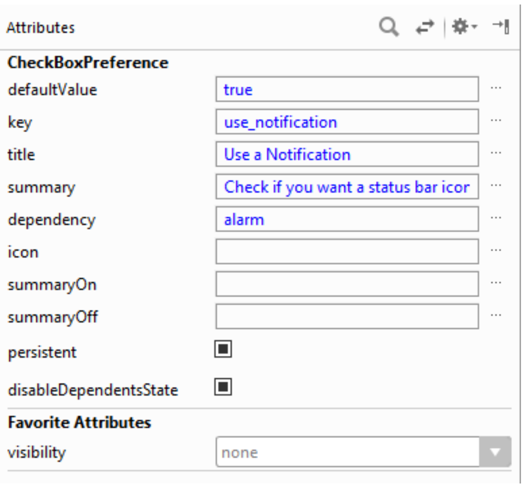
Virtual SD card 09:23
New Virtual SD card detected

27. Create a new project with the following information
    - **Application Name** : *Restaurant List*
    - **Company Domain** : *sp.com*
    - **Project Location** : *C:\MAD\AndroidStudioProjects\Lab5b or D:\MAD\AndroidStudioProjects\Lab5b*
    - **Minimum SDK** : *Android 5.0 (Lollipop)*
    - **Empty Activity name** : *RestaurantList*
    - **Layout name** : *main*

28. Close *RestaurantList.java* and *main.xml* files in the **Editor** pane
29. Open your **Windows File Explorer** and navigate to your **Android Studio** workspace where all you projects are created.
30. Copy *AndroidManifest.xml* file, **java** and **res** folders from **Lab5a\app\src\main** project folder and paste into **Lab5b\app\src\main** folder to overwrite the existing file and folder
31. Open the *preferences.xml* file from **res/xml** folder. We will add a new **CheckBoxPreference** named **use_notification** to provide a check box for user to choose the alarm alert format and save.

    In the Design mode, drag a "CheckBoxPreference" widget on to the "PreferenceScreen". Fill up the "CheckBoxPreference" attributes as shown.




Attributes

CheckBoxPreference

defaultValue	true
key	use_notification
title	Use a Notification
summary	Check if you want a status bar icor
dependency	alarm
icon	
summaryOn	
summaryOff	
persistent	▣
disableDependentsState	▣

Favorite Attributes

visibility	none

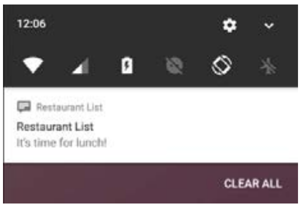
32. Check that preferences.xml is as shown.

```
<PreferenceScreen xmlns:android="http://schemas.android.com/apk/res/android">
    <ListPreference
        android:dialogTitle="Choose a sort order"
        android:entries="@array/sort_names"
        android:entryValues="@array/sort_clauses"
        android:key="sort_order"
        android:summary="Choose the order the list uses"
        android:title="Sort Order" />

    <CheckBoxPreference
        android:defaultValue="false"
        android:key="alarm"
        android:summary="Check if you want to know when it is time for lunch"
        android:title="Sound a Lunch Alarm" />

    <CheckBoxPreference
        android:defaultValue="true"
        android:dependency="alarm"
        android:key="use_notification"
        android:summary="Check if you want a status bar icon at lunchtime, or uncheck for a full-screen notice"
        android:title="Use a Notification" />

    <com.sp.restaurantlist.TimePreference
        android:defaultValue="12:00"
        android:dependency="alarm"
        android:key="alarm_time"
        android:summary="Set your desired time for the lunch alarm"
        android:title="Lunch Alarm Time" />

</PreferenceScreen>
```

33. Update the *OnAlarmReceiver.java* file to support choice of lunch time alarm alert by either a **Notification** or **AlarmActivity** and save

```
package com.sp.restaurantlist;

import android.app.Notification;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.preference.PreferenceManager;

public class OnAlarmReceiver extends BroadcastReceiver {
    private static final int NOTIFY_ME_ID = 1337;

    @Override
    public void onReceive(Context context, Intent intent) {
        SharedPreferences prefs = PreferenceManager
                .getDefaultSharedPreferences(context);
        boolean useNotification = prefs.getBoolean("use_notification", true);
        if (useNotification) {
            NotificationManager mgr = (NotificationManager)
context.getSystemService(Context.NOTIFICATION_SERVICE);
            PendingIntent i = PendingIntent.getActivity(context, 0, new
Intent(context, AlarmActivity.class), 0);
            Notification note = new Notification.Builder(context)
                    .setContentTitle("Restaurant List")
                    .setContentText("It's time for lunch!")
                    .setSmallIcon(android.R.drawable.stat_notify_chat)
                    .setContentIntent(i)
                    .setAutoCancel(true).build();
            note.flags |= Notification.FLAG_AUTO_CANCEL;
            mgr.notify(NOTIFY_ME_ID, note);
        } else {
            Intent i = new Intent(context, AlarmActivity.class);
            i.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
            context.startActivity(i);
        }
    }
}
```

34. Run the *Lab5b* project. At **Setting**, choose the **Notification** mode and set the alarm time to be 1 minute different from the time shown on emulator. The lunch time alarm will be displayed on the status bar or record in the **System Notifications** list



Restaurant List
Restaurant List
It's time for lunch!

35. Show to your lecturer when you have completed this section

Lecturer Signature : ___________________________________

-END-