

The background of the slide is a repeating pattern of various burgers. There are several types of burgers shown, including ones with cheese, lettuce, tomato, and meat. The pattern is dense and covers the entire slide area.

你想吃什麼？

姓名：胡坤程

學號：B10423056

指導教授：古東明

Xml 程式碼 (Main) :

```

<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@android:color/black"
    tools:background="@android:color/background_dark"
    tools:context="com.example.b10423056.wywe.MainActivity">

    <TextView
        android:id="@+id/result"
        android:layout_width="wrap_content"
        android:layout_height="0dp"
        android:layout_marginTop="100dp"
        android:gravity="center_vertical|center_horizontal|center"
        android:text="來看看你要吃什麼吧"
        android:textColor="@android:color/white"
        android:textSize="30sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.502"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <ImageButton
        android:id="@+id/btn_editNote"
        android:layout_width="61dp"
        android:layout_height="52dp"
        android:layout_marginBottom="32dp"
        android:layout_marginRight="32dp"
        android:background="@null"
        android:onClick="onNote"
        android:shape="oval"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:srcCompat="@mipmap/ic_launcher_round" />

    <Button
        android:id="@+id/btn_start"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="84dp"
        android:background="@android:color/black"
        android:onClick="onClick"

```

```

        android:text="START!"
        android:textColor="@android:color/holo_green_light"
        android:textSize="24sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintStart_toStartOf="parent" />

```

<Button

```

        android:id="@+id/btn_restart"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
        android:layout_marginEnd="28dp"
        android:onClick="onRestart"
        android:text="重抽"
        android:visibility="invisible"
        app:layout_constraintBottom_toTopOf="@+id/btn_start"
        app:layout_constraintEnd_toStartOf="@+id/guideline" />

```

<Button

```

        android:id="@+id/btn_confirm"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
        android:layout_marginStart="28dp"
        android:onClick="onConfirm"
        android:text="就是這家！"
        android:visibility="invisible"
        app:layout_constraintBottom_toTopOf="@+id/btn_start"
        app:layout_constraintStart_toStartOf="@+id/guideline" />

```

<android.support.constraint.Guideline

```

        android:id="@+id/guideline"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintGuide_percent="0.5"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent" />

```

```

</android.support.constraint.ConstraintLayout>

```

Xml 程式碼 (note_store) :

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@android:color/background_dark"
    android:backgroundTint="@android:color/background_dark"
    tools:context="com.example.b10423056.wywe.NoteStore">

    <Button
        android:id="@+id/btn_new"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
        android:layout_marginRight="28dp"
        android:layout_marginStart="8dp"
        android:layout_marginTop="28dp"
        android:background="@android:color/black"
        android:text="NEW"
        android:textColor="@android:color/holo_green_light"
        android:textSize="18sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintStart_toStartOf="@+id/guideline2"
        app:layout_constraintTop_toBottomOf="@+id/lv_store" />

    <ListView
        android:id="@+id/lv_store"
        android:layout_width="326dp"
        android:layout_height="361dp"
        android:layout_marginTop="28dp"
        android:background="@android:color/transparent"
        android:cacheColorHint="@android:color/background_light"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <ImageButton
        android:id="@+id/imageBut ton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
```

```
android:layout_marginEnd="8dp"  
android:layout_marginLeft="28dp"  
android:layout_marginTop="28dp"  
android:background="@android:color/black"  
android:onClick="onBack"  
app:layout_constraintBottom_toBottomOf="parent"  
app:layout_constraintEnd_toStartOf="@+id/guideline2"  
app:layout_constraintLeft_toLeftOf="parent"  
app:layout_constraintTop_toBottomOf="@+id/lv_store"  
app:srcCompat="@android:drawable/ic_menu_revert" />
```

```
<android.support.constraint.Guideline  
    android:id="@+id/guideline2"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:orientation="vertical"  
    app:layout_constraintGuide_percent="0.5" />
```

```
</android.support.constraint.ConstraintLayout>
```



程式概要：個人化隨機選擇店家吃，使用權重影響機率，資料儲存在資料庫。

主程式(MainActivity.java)：

```
package com.example.b10423056.wyhte;

import android.content.ContentValues;
import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteException;
import android.os.Handler;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.MotionEvent;
import android.view.View;
import android.widget.Button;
import android.widget.ImageButton;
import android.widget.TextView;

import java.util.ArrayList;
import java.util.Collections;
import java.util.Random;

public class MainActivity extends AppCompatActivity implements View.OnTouchListener, View.OnClickListener,
View.OnLongClickListener {

    static final String db_name="wyhteDB";
    static final String tb_name="store";
    SQLiteDatabase db;

    Cursor c ;
    ArrayList<String> storeList = new ArrayList<String>();    //儲存 Note 裡邊的資料
    ArrayList<String> choosenList = new ArrayList<String>();
    //Button
    Button btn_start;
    Button btn_restart;
    Button btn_confirm;
    ImageButton btn_editNote;
    TextView result;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        btn_start = (Button)findViewById(R.id.btn_start);
        btn_editNote = (ImageButton)findViewById(R.id.btn_editNote);
```

```

        btn_restart = (Button)findViewById(R.id.btn_restart);
        btn_confirm = (Button)findViewById(R.id.btn_confirm);
        result = (TextView)findViewById(R.id.result);

        choosenListReset();

        btn_start.setOnClickListener(this);

        //database
        if(checkDataBaseExist())
        {
            query();
        }
        else
        {
            //if note nothing
            result.setText("請點擊右下角增加店家");
        }

        String out = "";
        for(String str:storeList)
        {
            out = out + str + "\n";
        }
        //result.setText(out);
    }

    public void onNote(View v)
    {
        Intent it = new Intent(this, NoteStore.class);
        startActivityForResult(it,123);
    }

    protected void onActivityResult(int requestCode, int resultCode, Intent data) {
        super.onActivityResult(requestCode, resultCode, getIntent());
        if(resultCode==RESULT_OK && requestCode==123){
            query();
            choosenListReset();
        }
    }

    private void choosenListReset()
    {
        choosenList.clear();
        choosenList.add("你已經抽完全部選項了\n我也幫不了你...");
        choosenList.add("你 TMD 到底想吃什麼");
    }

```



```

private void query()
{
    db = openOrCreateDatabase(db_name, Context.MODE_PRIVATE, null);
    c = db.rawQuery("SELECT * FROM " + tb_name, null);
    storeList.clear();
    if(c.moveToFirst())
    {
        do {
            String sName = c.getString(0);
            double w = c.getDouble(1);
            while(w>0)
            {
                storeList.add(sName);
                w-=1;
            }
        }while(c.moveToNext());
    }
    Collections.shuffle(storeList);
    db.close();
}

private boolean checkDataBaseExist() {
    String path = "/data/data/com.example.b10423056.wyhte/databases/wyhteDB";
    SQLiteDatabase checkDB = null;
    try {
        checkDB = SQLiteDatabase.openDatabase(path, null,
            SQLiteDatabase.OPEN_READONLY);
        checkDB.close();
    } catch (SQLException e) {
        // database doesn't exist yet.
    }
    return checkDB != null;
}

@Override
public boolean onLongClick(View view) {
    return false;
}

private boolean started = false;
private Handler handler = new Handler();

private Runnable runnable = new Runnable() {
    @Override
    public void run() {
        final Random random = new Random();
        int i = random.nextInt(storeList.size());
    }
}

```



```

        result.setText(storeList.get(i));
        if(started) {
            handler.postDelayed(this,10);
        }
    }
};

@Override
public boolean onTouch(View view, MotionEvent motionEvent) {
    /*
    if(motionEvent.getAction() == MotionEvent.ACTION_DOWN)
    {
        started = true;
        handler.post(runnable);
    }
    if(motionEvent.getAction() == MotionEvent.ACTION_UP)
    {
        started = false;
    }
    return true;
    */
    return false;
}

@Override
public void onClick(View view) {
    if(storeList.size() > 0)
    {
        if(started)
        {
            started = false;
            btn_start.setText("START!");
            btn_start.setVisibility(View.INVISIBLE);
            btn_confirm.setVisibility(View.VISIBLE);
            btn_restart.setVisibility(View.VISIBLE);
        }
        else
        {
            started = true;
            btn_start.setText("STOP");
            btn_visibility_origin();
        }
        handler.post(runnable);
    }
    else if(chooseList.size() > 2)
    {
        Random rnd = new Random();

```

```

        result.setText(choosenList.get(rnd.nextInt(2)));
    }
    else
    {
        result.setText("請點擊右下角增加店家");
    }
}

private void btn_visibility_origin()
{
    btn_start.setVisibility(View.VISIBLE);
    btn_confirm.setVisibility(View.INVISIBLE);
    btn_restart.setVisibility(View.INVISIBLE);
}

public void onRestart(View v)
{
    db = openOrCreateDatabase(db_name, Context.MODE_PRIVATE, null);
    String sName = result.getText().toString();
    c = db.rawQuery("SELECT weight FROM " + tb_name + " WHERE storeName='" + sName + "'", null);
    c.moveToFirst();
    double new_weight = c.getDouble(0);
    if(new_weight > 3)
    {
        new_weight -= 0.3;
    }
    updateDB("weight", new_weight, sName);
    db.close();
    query();
    choosenList.add(sName);
    storeList.removeAll(choosenList);
    onClick(v);
}

public void onComfirm(View v)
{
    db = openOrCreateDatabase(db_name, Context.MODE_PRIVATE, null);
    String sName = result.getText().toString();

    c = db.rawQuery("SELECT weight FROM " + tb_name + " WHERE storeName='" + sName + "'", null);
    c.moveToFirst();
    double new_weight = c.getDouble(0);
    if(new_weight < ((double)storeList.size()/4.0))
    {
        new_weight += 0.7;
    }
    updateDB("weight", new_weight, sName);
}

```

```

c = db.rawQuery("SELECT freq FROM " + tb_name + " WHERE storeName='"+sName+"'", null);
c.moveToFirst();
int new_freq = c.getInt(0)+1;
updateDB("freq",new_freq, sName);
query();
db.close();

btn_visibility_origin();
result.setText("你選擇了：\n"+sName+"\n 祝你用餐愉快 :)");
}

private void updateDB(String weightORfreq, double newI, String sName)
{
    if(weightORfreq == "freq")
    {
        newI = (int)newI;
    }
    String update = "UPDATE " + tb_name + " SET " + weightORfreq + " = " + newI + " WHERE storeName = '" + sName + "'";
    c = db.rawQuery(update, null);
    c.moveToFirst();
}
}

```

程式碼說明：

1. 在這個主程式里，主要功能是：抽、顯示結果及進入選項儲存畫面。
2. 一開始系統會檢查是否有資料庫，若沒有則會提示前往增加選項；有的話則將資料表 **store** 的欄位 **storeName** 乘以欄位 **weight**，放入 **storeList** 中。(例：淇卡*權重 3，**storeList** 就會有 3 個淇卡)
3. 將資料全存到 **storeList** 後，會執行 **shuffle** 將 **storeList** 裡面的資料打亂。
4. 按下 **START!** 後，**button** 會轉換成 **STOP**，而且畫面會一直隨機更換店家名稱，直到按下 **STOP** 為止。
5. 當按下 **STOP** 後，會出現兩個選項：“重抽”及“就是這家！”。
6. 按下重抽：會將當下被拋棄選項加入 **choosenList**，**storeList** 則會依照 **choosenList** 剔除已選選項。其選項的權重也會降低 0.3。
7. 按下“就是這家！”：會增加其選項權重 0.7，並增加被選擇次數 1 次，最後出現祝福畫面。
8. 若使用者重抽了全部選項，則會出現兩種(隨機選擇一種)系統憤怒及悲哀的訊息。

NoteStore.java :

```

package com.example.b10423056.wyhte;

import android.app.AlertDialog;
import android.content.ContentValues;
import android.content.Context;
import android.content.DialogInterface;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.graphics.Color;
import android.support.v4.widget.SimpleCursorAdapter;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.view.ViewGroup;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.TextView;
import android.widget.Toast;
import java.util.ArrayList;
import java.util.Collections;

public class NoteStore extends AppCompatActivity implements AdapterView.OnItemClickListener {
    //database
    static final String db_name="wyhteDB";
    static final String tb_name="store";
    //static final String[] FROM = new String[] {"storeName", "weight", "freq"};
    SQLiteDatabase db;

    EditText input;
    Button newBtn;
    ListView lvStore;
    Cursor c ;
    ArrayList<String> storeList = new ArrayList<String>();    //儲存 Note 裡邊的資料
    ArrayList<String> weightList = new ArrayList<>();
    ArrayAdapter<String> adapter;
    //SimpleCursorAdapter adapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_note_store);
        //AlertDialog

```

```

AlertDialog.Builder builder = new AlertDialog.Builder(this);
builder.setTitle("新建");
builder.setIcon(android.R.drawable.ic_input_add);
builder.setMessage("請輸入名稱：");

input = new EditText(this);
builder.setView(input);

//SET positive button
builder.setPositiveButton("確定", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialogInterface, int i) {
        String str = input.getText().toString().replaceAll("[\\t\\n\\r]+", " ");
        Toast.makeText(getApplicationContext(), str, Toast.LENGTH_SHORT).show();
        input.setText("");

        addData(str);
        query();
    }
});

//Set negative button
builder.setNegativeButton("取消", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialogInterface, int i) {
        dialogInterface.dismiss();
        input.setText("");
    }
});

//Create the dialog
final AlertDialog ad = builder.create();

//=====
//button
newBtn = (Button) findViewById(R.id.btn_new);
newBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        ad.show();
    }
});

//database
db = openOrCreateDatabase(db_name, Context.MODE_PRIVATE, null);
//create table
String createTable = "CREATE TABLE IF NOT EXISTS " + tb_name +
    "(storeName VARCHAR(255) PRIMARY KEY NOT NULL, " +

```

```

        "weight DOUBLE default 3, " +
        "freq INTEGER default 0) ";
db.execSQL(createTable);

//listView
lvStore = (ListView) findViewById(R.id.lv_store);

//Adapter
adapter = new ArrayAdapter<String>(this, android.R.layout.simple_list_item_1, storeList){
    @Override
    public View getView(int position, View convertView, ViewGroup parent){
        // Get the Item from ListView
        View view = super.getView(position, convertView, parent);
        // Initialize a TextView for ListView each Item
        TextView tv = (TextView) view.findViewById(android.R.id.text1);
        // Set the text color of TextView (ListView Item)
        tv.setTextColor(Color.GREEN);
        // Generate ListView Item using TextView
        return view;
    }
};
lvStore.setAdapter(adapter);
lvStore.setOnItemClickListener(this);
query();

}
private void addData(String name)
{
    ContentValues cv = new ContentValues(3);
    cv.put("storeName", name);
    db.insert(tb_name, null, cv);
}

private void delData(String name)
{
    db.delete(tb_name, "storeName = "+name,null);
    query();
}

@Override
public void onItemClick(AdapterView<?> adapterView, View view, int index, long l) {
    final String str = storeList.get(index).split("已")[0].replaceAll("\\s", "");
    AlertDialog.Builder bbb = new AlertDialog.Builder(this).setMessage("你確定要刪除\"" + str + "\"嗎？")
        .setCancelable(true)
        .setIcon(android.R.drawable.ic_delete)
        .setTitle("確定？");
    bbb.setPositiveButton("刪除", new DialogInterface.OnClickListener() {
        @Override

```

```

        public void onClick(DialogInterface dialogInterface, int i) {
            delData("'" + str + "'");
        }
    });
    bbb.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialogInterface, int i) {
            dialogInterface.dismiss();
        }
    });
    AlertDialog alert = bbb.create();
    alert.show();
}

private void query()
{
    c = db.rawQuery("SELECT * FROM " + tb_name, null);
    storeList.clear();
    if(c.moveToFirst())
    {
        String sName;
        int freq;
        do {
            sName = c.getString(0);
            freq = c.getInt(2);
            sName = sName + "          已去過次數：" + freq;
            storeList.add(sName);
        }while(c.moveToNext());
    }
    adapter.notifyDataSetChanged();
}

public void onBack(View v)
{
    db.close();
    setResult(RESULT_OK);
    finish();
}
}

```

程式碼說明：

9. 在 NoteStore.java 里，主要功能是：顯示以增加的選項、增加及刪除選項和返回鍵。
10. 這裡也會檢查是否已有資料庫，若沒有則創建新的。
11. 點一下 ListView 的選項，則會出現 Alert 視窗詢問是否真要刪除。
12. 點下新增則會出現 Alert 視窗要求輸入。
13. 返回鍵則會返回上一頁。

執行畫面/影片：

影片網址：<https://youtu.be/s2ZRArPOw0Y>

