

Tuto 2 : Premiers pas de développement sous Android

On propose de mettre en place un exemple d'Application android pour le calcul de L'IMC : Indice de masse corporelle.

- La formule pour calculer l'IMC est poids (en kilogrammes) / taille (en mètres).
- Le bouton RAZ remet à zéro tous les champs.
- Dès qu'on change les valeurs dans les champs Poids et Taille, on remet le texte du résultat par défaut puisque la valeur calculée n'est plus valable pour les nouvelles valeurs.
- On enverra un message d'erreur si l'utilisateur essaie de faire le calcul avec une taille égale à zéro grâce à un Toast.
- Un Toast est un widget un peu particulier qui permet d'afficher un message à n'importe quel moment sans avoir à créer de vue. Il est destiné à informer l'utilisateur sans le déranger outre mesure ; ainsi l'utilisateur peut continuer à utiliser l'application comme.
- La CheckBox de megafonction permet de changer le résultat du calcul en un message.
- Pour détecter le moment où l'utilisateur écrit dans un EditText, on peut utiliser l'évènement onKey. Problème, cette technique ne fonctionne que sur les claviers virtuels.
- Le design de l'App sera comme suit



Pour faire simple on utilise le linearLayout qui permet d'organiser tous les objets children linéairement dans une direction.

On utilisera uniquement le XML le res/layout/activity_mainxml:

Pour les objets TextView
android:text="Poids:"
android:textStyle="bold"

Texte de la TextView Police en gras





```
Couleur
android:textColor="#FF0000"
                                      Centrer le texte dans le champ
android:gravity="center"
android:background="#c5e1b0"
                                      Couleur des champ TextView
Pour les objets EditText on utilise les attributs:
<EditText
    android:id="@+id/poids"
    android:layout width="fill parent"
                                                 largeur
    android:layout_height="wrap_content"
                                                 longueur
                                                 Donne une idée sur l'entrée à insérer
    android:hint="Poids"
    android:inputType="numberDecimal"
                                                 on utilise l'attribut android:inputType
pour insérer la valeur numbers dans la EditText.
    />
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"</pre>
    android:layout width="fill parent"
    android:layout height="fill parent"
    android:orientation="vertical">
    <TextView
        android:layout width="fill parent"
        android:layout height="36dp"
        android:gravity="center"
        android:text="Poids : "
        android:textColor="#009688"
        android:textStyle="bold"
        android:background="#c5e1b0"/>
    <EditText
        android:id="@+id/poids"
        android:layout width="fill parent"
        android:layout_height="wrap_content"
        android:hint="Poids"
        android:inputType="numberDecimal"
        />
    <TextView
        android:layout width="fill parent"
        android:layout height="31dp"
        android: gravity="center"
        android:text="Taille : "
```

<EditText

android:textColor="#009688"
android:textStyle="bold"

android:id="@+id/taille"

android:hint="Taille"

android:background="#c5e1b0"/>

android:layout_width="fill_parent"
android:layout height="wrap content"

android:inputType="numberDecimal"



```
< Radio Group
        android:id="@+id/group"
        android:layout width="292dp"
        android:layout height="wrap content"
        android: checkedButton="@+id/radio2"
        android:orientation="horizontal">
        < RadioButton
            android:id="@+id/radio1"
            android:layout width="wrap content"
            android:layout height="wrap content"
            android:text="Metre" />
        < Radio Button
            android:id="@+id/radio2"
            android:layout width="wrap content"
            android:layout height="wrap content"
            android:text="Centimètre" />
    </RadioGroup>
    <CheckBox
        android:id="@+id/mega"
    android:layout width="wrap content"
    android: layout height="wrap content"
    android:text="Mega fonction !"
    />
    <Button
        android:id="@+id/calcul"
        android:layout width="wrap content"
        android: layout height="wrap content"
        android:text="Calculer 1'IMC"
        />
    <Button
        android:id="@+id/raz"
        android:layout_width="wrap_content"
android:layout_height="wrap_content"
        android: text="RAZ"
        />
    <TextView
        android:layout width="wrap content"
        android:layout height="wrap content"
        android:text="Résultat:"
        />
    <TextView
        android:id="@+id/result"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:text="c"
        />
</LinearLayout>
```



Une fois terminé on Remarque le nouveau design de l'app sur l'aperçu graphique.

Tout ce qu'il reste à faire c'est de programmer les Items avec le code JAVA

En dessous le code java à implémenter dans le MainActivity.java

```
package com.example.myapp1;
import androidx.appcompat.app.AppCompatActivity;
import android.app.Activity;
import android.os.Bundle;
import android.text.Editable;
import android.text.TextWatcher;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.RadioGroup;
import android.widget.TextView;
import android.widget.Toast;
public class MainActivity extends Activity {
    // La chaîne de caractères par défaut
    private final String defaut = "Cliquez sur le bouton « Calculer l'IMC
» pour le résultat.";
    // La chaîne de caractères de la megafonction
    private final String megaString = "Vous faites un poids parfait !";
    Button envoyer = null;
    Button raz = null;
    EditText poids = null;
    EditText taille = null;
    RadioGroup group = null;
    TextView result = null;
    CheckBox mega = null;
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity main);
// On récupère toutes les vues dont on a besoin
        envoyer = (Button) findViewById(R.id.calcul);
        raz = (Button) findViewById(R.id.raz);
        taille = (EditText) findViewById(R.id.taille);
        poids = (EditText) findViewById(R.id.poids);
        mega = (CheckBox)findViewById(R.id.mega);
        group = (RadioGroup) findViewById(R.id.group);
        result = (TextView) findViewById(R.id.result);
// On attribue un listener adapté aux vues qui en ont besoin
```



```
envoyer.setOnClickListener(envoyerListener);
        raz.setOnClickListener(razListener);
        taille.addTextChangedListener(textWatcher);
        poids.addTextChangedListener(textWatcher);
// Solution avec des onKey
//taille.setOnKeyListener(modificationListener);
//poids.setOnKeyListener(modificationListener);
        mega.setOnClickListener(checkedListener);
    }
    /*
    // Se lance à chaque fois qu'on appuie sur une touche en étant sur un
    private OnKeyListener modificationListener = new OnKeyListener() {
    @Override
    public boolean onKey(View v, int keyCode, KeyEvent event) {
    // On remet le texte à sa valeur par défaut pour ne pas avoir de
résultat incohérent
    result.setText(defaut);
    return false;
    };*/
    private TextWatcher textWatcher = new TextWatcher() {
        @Override
        public void onTextChanged(CharSequence s, int start, int before,
int count)
            result.setText(defaut);
        @Override
        public void beforeTextChanged(CharSequence s, int start, int
count,
                                       int after) {
        @Override
        public void afterTextChanged(Editable s) {
    };
    // Uniquement pour le bouton "envoyer"
    private OnClickListener envoyerListener = new OnClickListener() {
        @Override
        public void onClick(View v) {
            if(!mega.isChecked()) {
// Si la megafonction n'est pas activée
// On récupère la taille
                String t = taille.getText().toString();
// On récupère le poids
                String p = poids.getText().toString();
                float tValue = Float.valueOf(t);
// Puis on vérifie que la taille est cohérente
                if(tValue == 0)
                    Toast.makeText(MainActivity.this, "Hého, tu es un
Minipouce ou quoi ?",
                            Toast. LENGTH SHORT) . show();
                else {
```



```
float pValue = Float.valueOf(p);
// Si l'utilisateur a indiqué que la taille était en centimètres
// On vérifie que la Checkbox sélectionnée est la deuxième à l'aide de son
identifiant
                    if(group.getCheckedRadioButtonId() == R.id.radio2)
                        tValue = tValue / 100;
                    tValue = (float) Math.pow(tValue, 2);
                    float imc = pValue / tValue;
                    result.setText("Votre IMC est " +
String.valueOf(imc));
                }
            } else
                result.setText (megaString);
        }
    } ;
    // Listener du bouton de remise à zéro
    private OnClickListener razListener = new OnClickListener() {
        @Override
        public void onClick(View v) {
            poids.getText().clear();
            taille.getText().clear();
            result.setText(defaut);
        }
    } ;
    // Listener du bouton de la megafonction.
    private OnClickListener checkedListener = new OnClickListener() {
        @Override
        public void onClick(View v) {
// On remet le texte par défaut si c'était le texte de la megafonction qui
était écrit
            if(!((CheckBox)v).isChecked() &&
result.getText().equals(megaString))
result.setText(defaut);
};
```