SEMESTER PROJECT

(FASTnFITNESS)

LINKS:

Video on github https://github.com/Faizan-5/fastnfitnessapp.git

(github)

GROUP MEMBERS:

- ALI HAMZA
- HUSSNAIN ZIA
- FAIZAN NABI

PROJECT DESCRIPTION: FITNESS APP

Project Goal:

To develop a comprehensive fitness app that empowers users to achieve their health and wellness objectives through personalized workout plans, nutrition tracking, progress monitoring, and a supportive community.

Target Audience:

The app will cater to a diverse range of users, including individuals seeking weight loss, muscle gain, improved fitness levels, and overall well-being.

Key Features:

- **Personalized Workout Plans:** Generate tailored workout routines based on user fitness goals, experience level, and equipment availability.
- **Exercise Library:** Offer a vast database of exercises with detailed instructions, video demonstrations, and difficulty levels.
- **Nutrition Tracking:** Provide tools for calorie counting, macronutrient tracking, and meal planning to support dietary goals.
- **Progress Tracking:** Monitor user progress through metrics like weight, body measurements, and exercise performance.

Technology Stack:

- SQLite □ Java

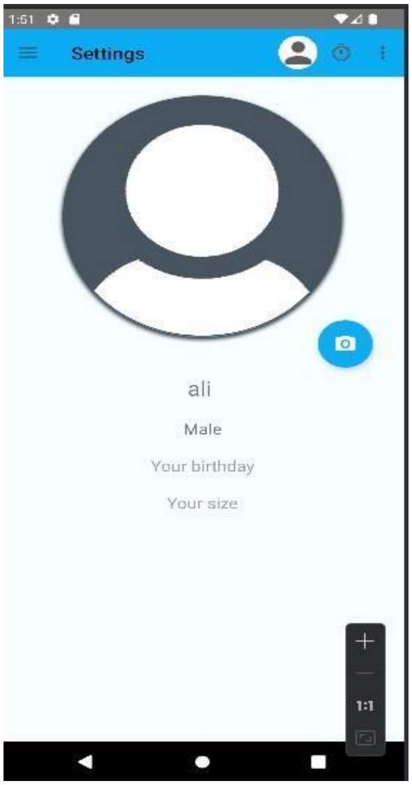
Key features/Functionalities/Screens:

1. **Profile Management**:
   - Creating, selecting, deleting, and renaming profiles.
   - Setting and displaying profile pictures.
2. **Workout Tracking**:
   - Logging and viewing strength training exercises (FontesPagerFragment). o Recording and tracking body weight (WeightFragment).
   - Logging cardio exercises and viewing workout history.
3. **Exercise Data**:
   - Viewing and managing exercise machines (MachineFragment).
   - Accessing and following workout programs (ProgramListFragment).
4. **Progress Tracking**:
   - Tracking body measurements (BodyPartListFragment).
   - Viewing progress pictures (ProgressImagesFragment).
5. **Settings and Preferences**:
   - Customizing app settings (SettingsFragment).
   - Day/night mode selection.
6. **Data Management**:
   - Exporting workout data to CSV format. o Importing workout data from CSV or ZIP files.
   - Database management (deleting and migrating).
7. **Additional Features**:
   - Music control integration. o Chronometer for tracking workout duration. o Introduction/tutorial screens for new users.

Dependencies/Libraries Used:

1. **Android Support Libraries**: For compatibility and UI elements.
2. **CircularImageView**: For displaying profile pictures in a circular format.
3. **KToast**: For displaying customized toast messages.
4. **SweetAlert**: For displaying customizable alert dialogs.
5. **SQLite**: For database management.
6. **MusicController**: A custom class for music control.
7. **ImageUtil**: A custom class for image handling.
8. **DateConverter**, **FileNameUtil**, **UnitConverter**: Custom utility class

SCREEN SHOT (UI):



1:51

Settings

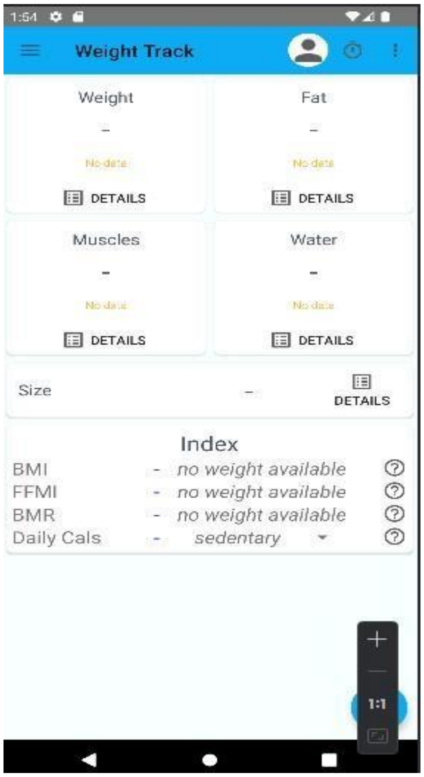
ali

Male

Your birthday

Your size

1:1



1:54

Weight Track

Weight
–
No data
DETAILS

Fat
–
No data
DETAILS

Muscles
–
No data
DETAILS

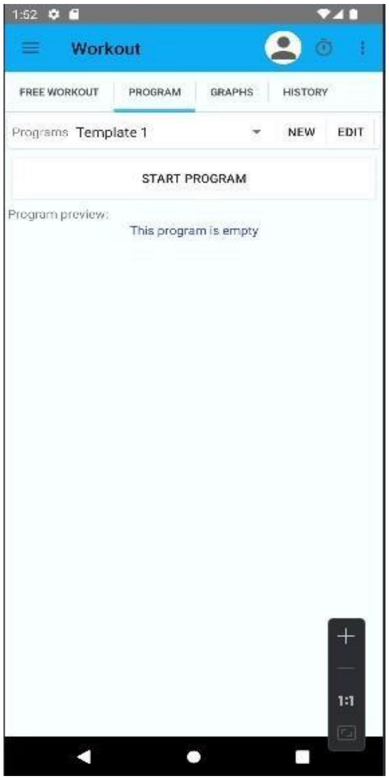
Water
–
No data
DETAILS

Size – DETAILS

Index

BMI	-	*no weight available*	?
FFMI	-	*no weight available*	?
BMR	-	*no weight available*	?
Daily Cals	-	*sedentary*	?

1:1



1:52

Workout

FREE WORKOUT | PROGRAM | GRAPHS | HISTORY

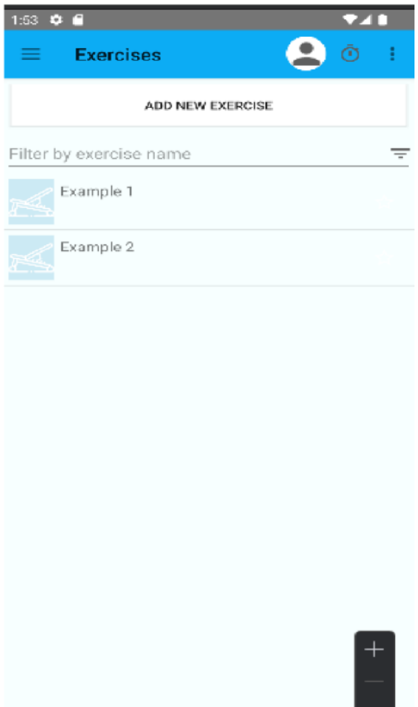
Programs Template 1 NEW EDIT

START PROGRAM

Program preview:

This program is empty

1:1



1:53

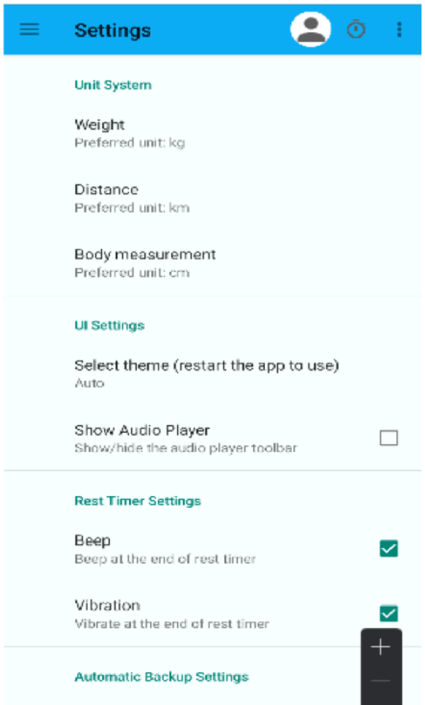
Exercises

ADD NEW EXERCISE

Filter by exercise name

Example 1

Example 2



Settings

Unit System

Weight
Preferred unit: kg

Distance
Preferred unit: km

Body measurement
Preferred unit: cm

UI Settings

Select theme (restart the app to use)
Auto

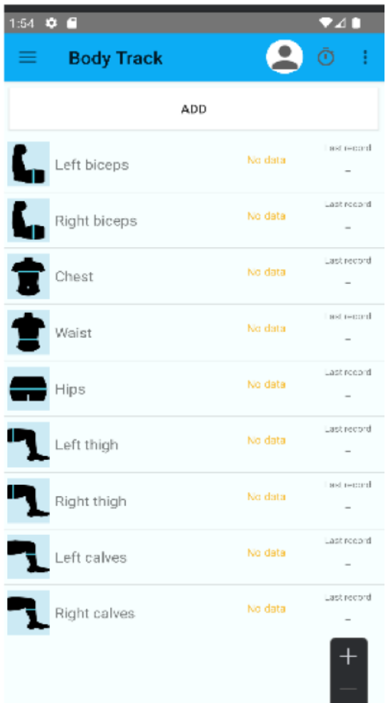
Show Audio Player
Show/hide the audio player toolbar □

Rest Timer Settings

Beep
Beep at the end of rest timer ☑

Vibration
Vibrate at the end of rest timer ☑

Automatic Backup Settings

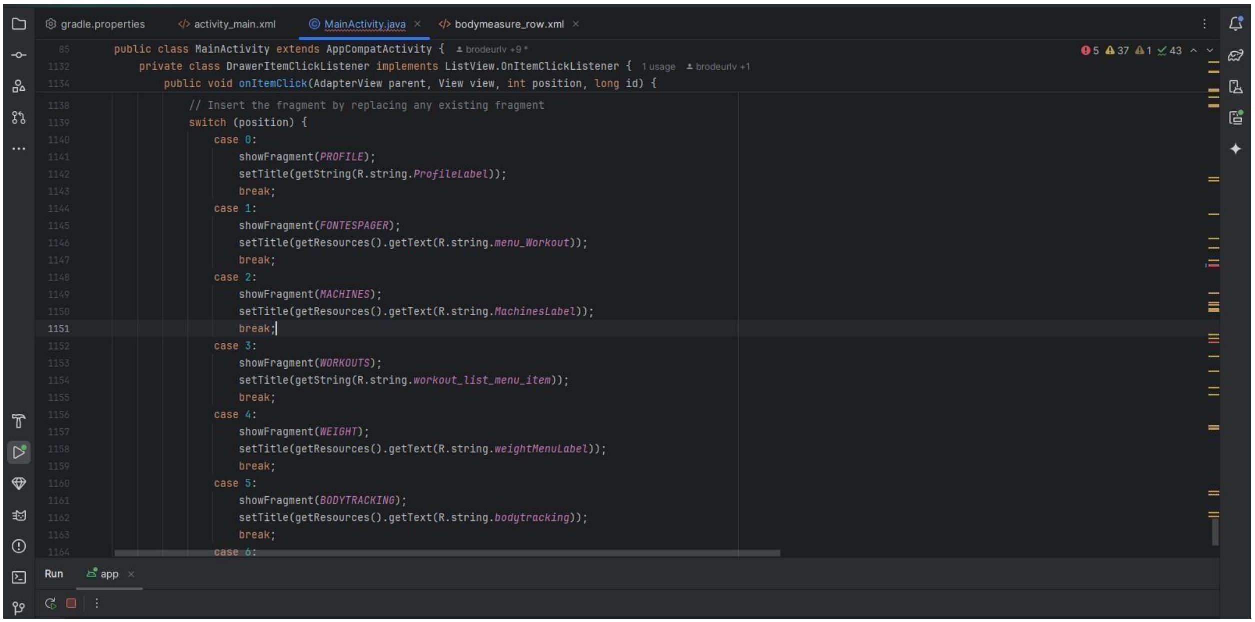


1:54

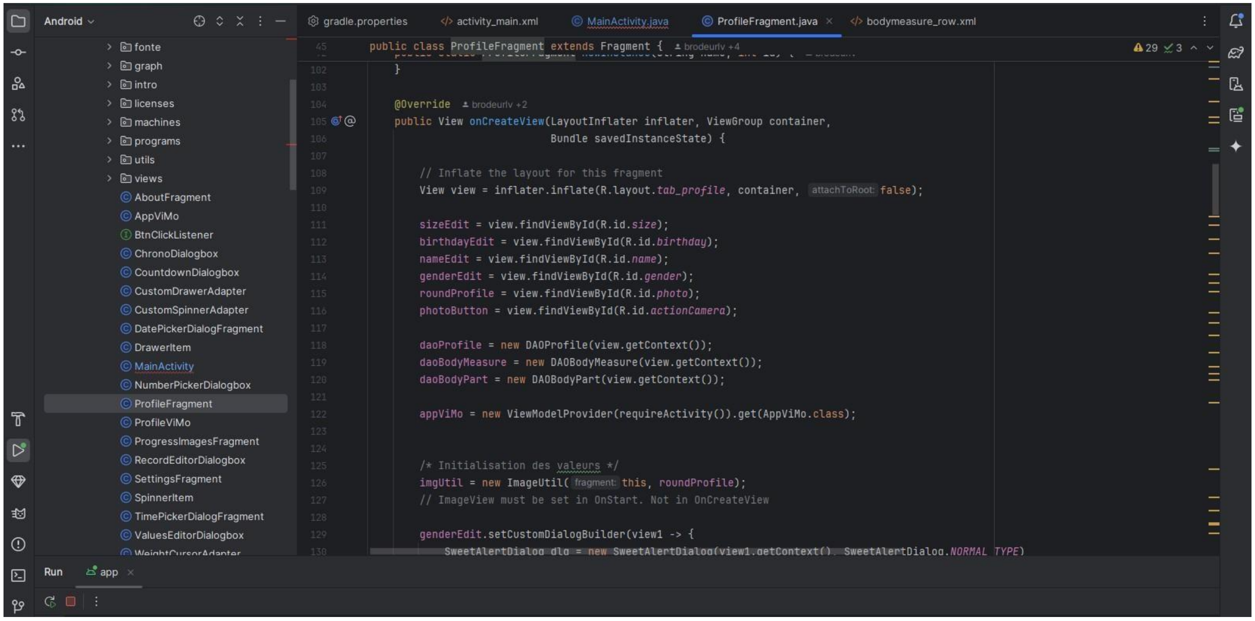
Body Track

ADD

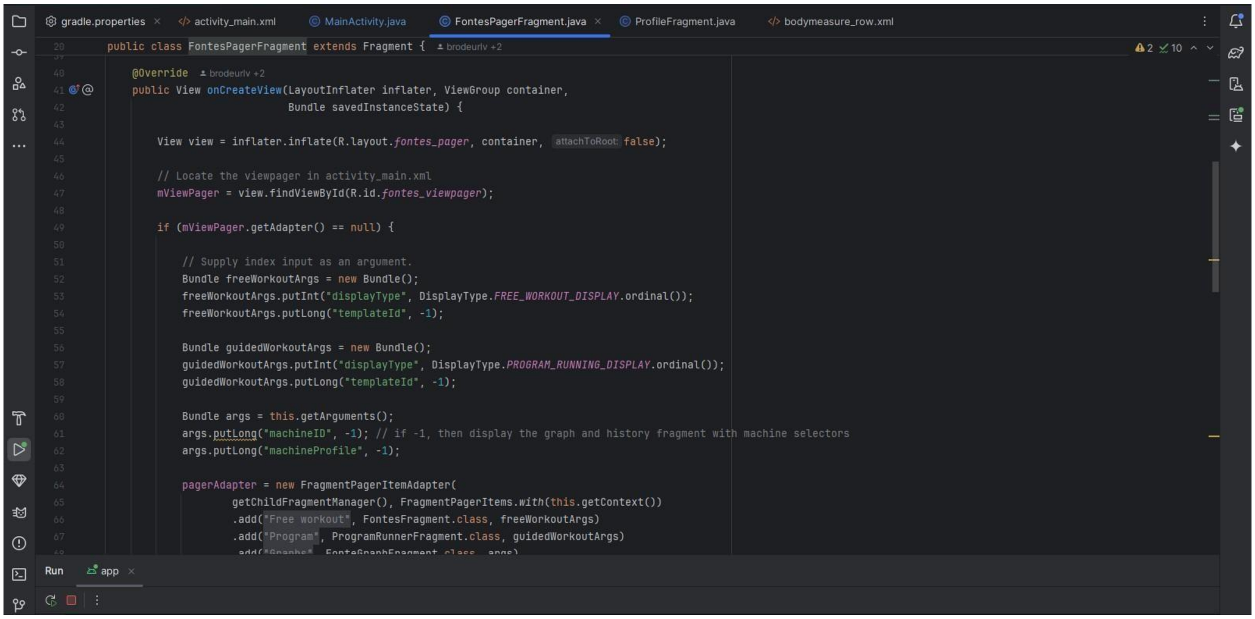
Left biceps	No data	Last record –
Right biceps	No data	Last record –
Chest	No data	Last record –
Waist	No data	Last record –
Hips	No data	Last record –
Left thigh	No data	Last record –
Right thigh	No data	Last record –
Left calves	No data	Last record –
Right calves	No data	Last record –



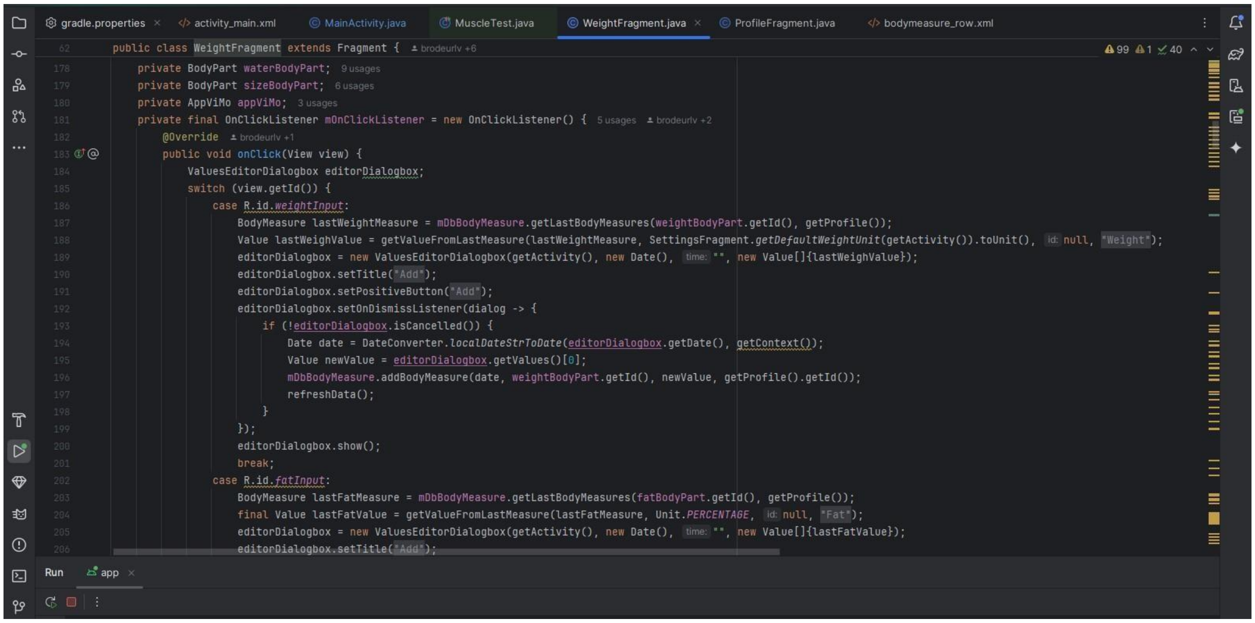
```
public class MainActivity extends AppCompatActivity {
    private class DrawerItemClickListener implements ListView.OnItemClickListener {
        public void onItemClick(AdapterView parent, View view, int position, long id) {
            // Insert the fragment by replacing any existing fragment
            switch (position) {
                case 0:
                    showFragment(PROFILE);
                    setTitle(getString(R.string.ProfileLabel));
                    break;
                case 1:
                    showFragment(FONTESPAGER);
                    setTitle(getResources().getText(R.string.menu_Workout));
                    break;
                case 2:
                    showFragment(MACHINES);
                    setTitle(getResources().getText(R.string.MachinesLabel));
                    break;
                case 3:
                    showFragment(WORKOUTS);
                    setTitle(getString(R.string.workout_list_menu_item));
                    break;
                case 4:
                    showFragment(WEIGHT);
                    setTitle(getResources().getText(R.string.weightMenuLabel));
                    break;
                case 5:
                    showFragment(BODYTRACKING);
                    setTitle(getResources().getText(R.string.bodytracking));
                    break;
                case 6:
```



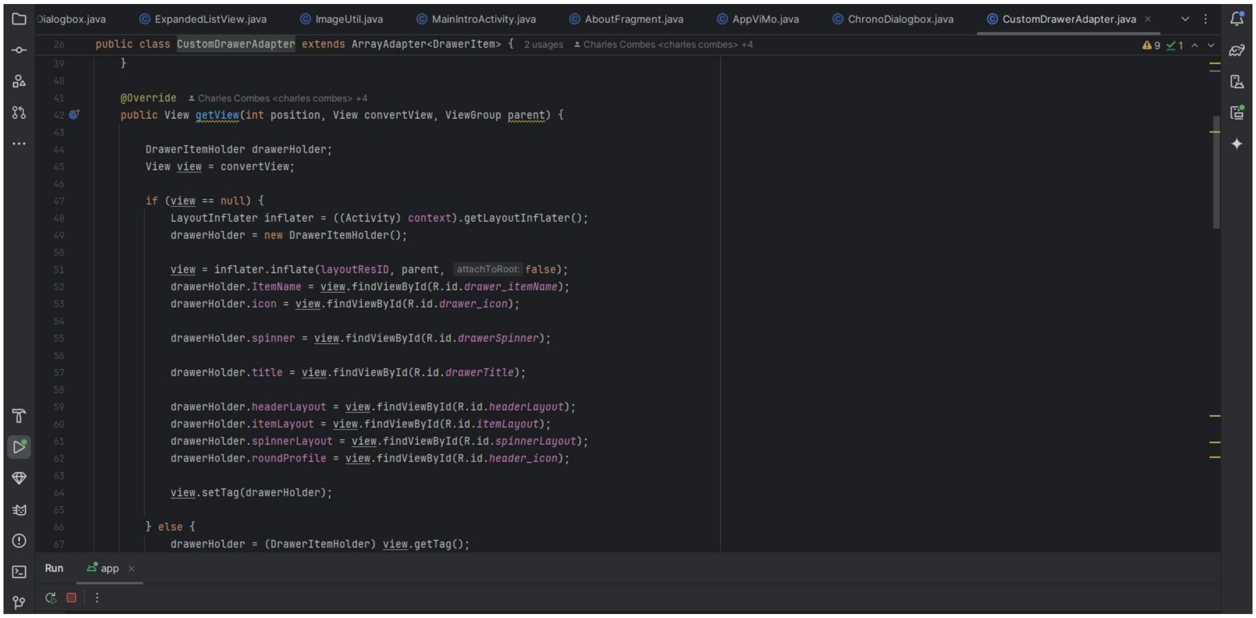
```
public class ProfileFragment extends Fragment {
    }

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

        // Inflate the layout for this fragment
        View view = inflater.inflate(R.layout.tab_profile, container, attachToRoot: false);

        sizeEdit = view.findViewById(R.id.size);
        birthdayEdit = view.findViewById(R.id.birthday);
        nameEdit = view.findViewById(R.id.name);
        genderEdit = view.findViewById(R.id.gender);
        roundProfile = view.findViewById(R.id.photo);
        photoButton = view.findViewById(R.id.actionCamera);

        daoProfile = new DAOProfile(view.getContext());
        daoBodyMeasure = new DAOBodyMeasure(view.getContext());
        daoBodyPart = new DAOBodyPart(view.getContext());

        appViMo = new ViewModelProvider(requireActivity()).get(AppViMo.class);


        /* Initialisation des valeurs */
        imgUtil = new ImageUtil( fragment: this, roundProfile);
        // ImageView must be set in OnStart. Not in OnCreateView

        genderEdit.setCustomDialogBuilder(view1 -> {
            SweetAlertDialog dlg = new SweetAlertDialog(view1.getContext(), SweetAlertDialog.NORMAL_TYPE)
```



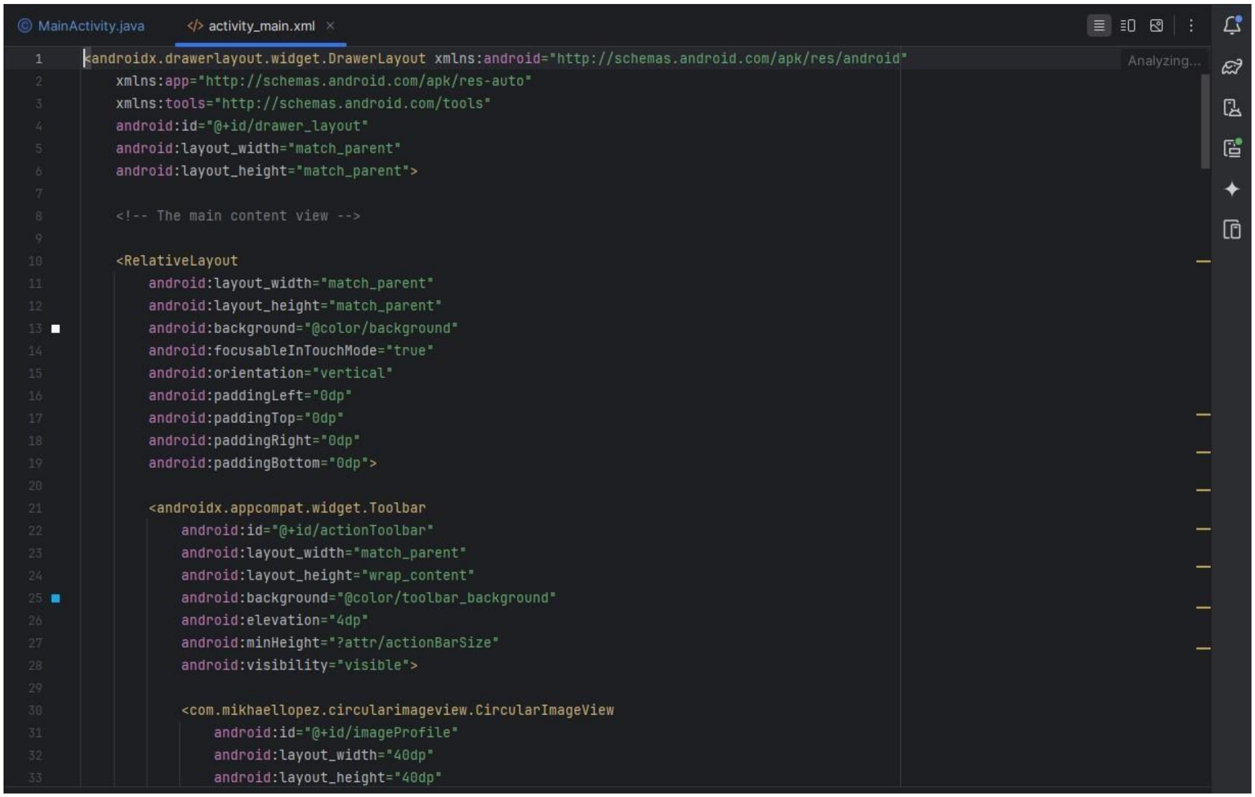
```java
public class FontesPagerFragment extends Fragment {  brodeurlv +2
    @Override  brodeurlv +2
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {

        View view = inflater.inflate(R.layout.fontes_pager, container, attachToRoot: false);

        // Locate the viewpager in activity_main.xml
        mViewPager = view.findViewById(R.id.fontes_viewpager);

        if (mViewPager.getAdapter() == null) {

            // Supply index input as an argument.
            Bundle freeWorkoutArgs = new Bundle();
            freeWorkoutArgs.putInt("displayType", DisplayType.FREE_WORKOUT_DISPLAY.ordinal());
            freeWorkoutArgs.putLong("templateId", -1);

            Bundle guidedWorkoutArgs = new Bundle();
            guidedWorkoutArgs.putInt("displayType", DisplayType.PROGRAM_RUNNING_DISPLAY.ordinal());
            guidedWorkoutArgs.putLong("templateId", -1);

            Bundle args = this.getArguments();
            args.putLong("machineID", -1); // if -1, then display the graph and history fragment with machine selectors
            args.putLong("machineProfile", -1);

            pagerAdapter = new FragmentPagerItemAdapter(
                    getChildFragmentManager(), FragmentPagerItems.with(this.getContext())
                    .add("Free workout", FontesFragment.class, freeWorkoutArgs)
                    .add("Program", ProgramRunnerFragment.class, guidedWorkoutArgs)
                    .add("Graphs", FonteGraphFragment.class, args)
```



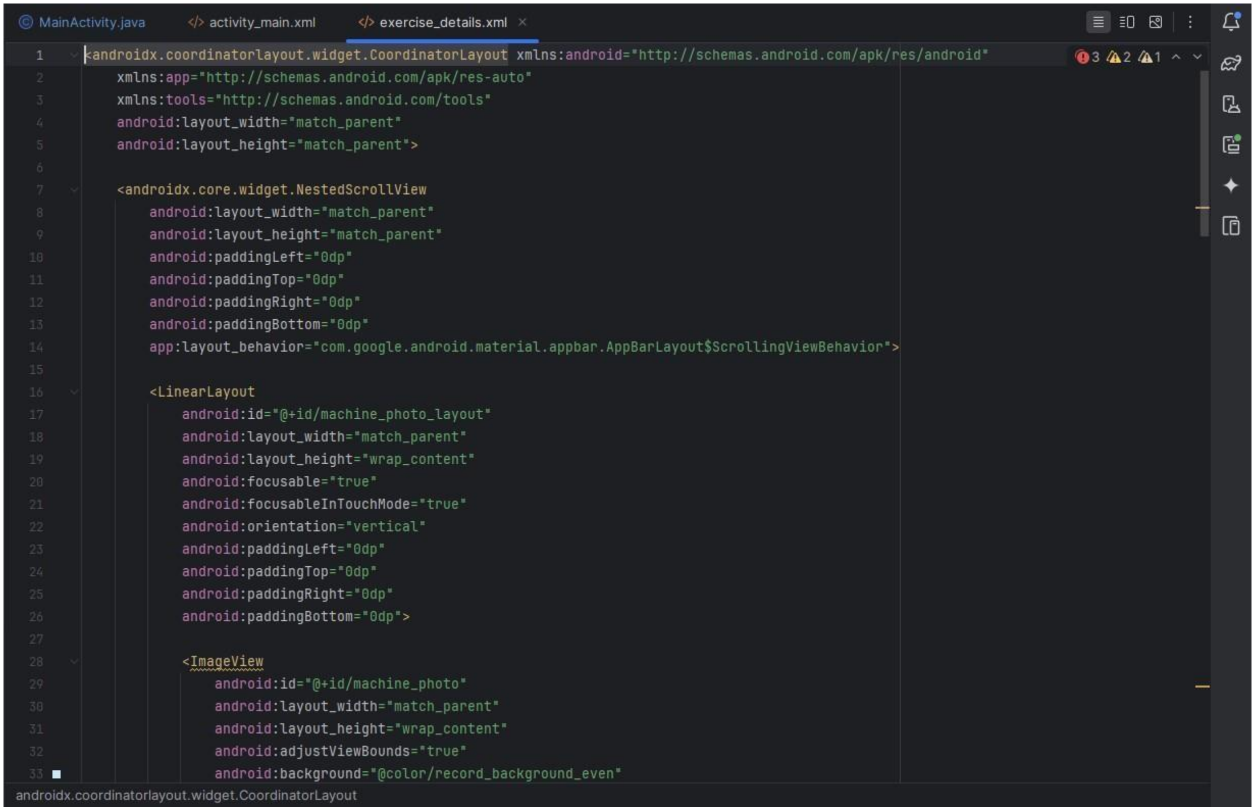
```java
public class WeightFragment extends Fragment {  brodeurlv +6
    private BodyPart waterBodyPart;  9 usages
    private BodyPart sizeBodyPart;  6 usages
    private AppViMo appViMo;  3 usages
    private final OnClickListener mOnClickListener = new OnClickListener() {  5 usages  brodeurlv +2
        @Override  brodeurlv +1
        public void onClick(View view) {
            ValuesEditorDialogbox editorDialogbox;
            switch (view.getId()) {
                case R.id.weightInput:
                    BodyMeasure lastWeightMeasure = mDbBodyMeasure.getLastBodyMeasures(weightBodyPart.getId(), getProfile());
                    Value lastWeighValue = getValueFromLastMeasure(lastWeightMeasure, SettingsFragment.getDefaultWeightUnit(getActivity()).toUnit(), id: null, "Weight");
                    editorDialogbox = new ValuesEditorDialogbox(getActivity(), new Date(), time: "", new Value[]{lastWeighValue});
                    editorDialogbox.setTitle("Add");
                    editorDialogbox.setPositiveButton("Add");
                    editorDialogbox.setOnDismissListener(dialog -> {
                        if (!editorDialogbox.isCancelled()) {
                            Date date = DateConverter.localDateStrToDate(editorDialogbox.getDate(), getContext());
                            Value newValue = editorDialogbox.getValues()[0];
                            mDbBodyMeasure.addBodyMeasure(date, weightBodyPart.getId(), newValue, getProfile().getId());
                            refreshData();
                        }
                    });
                    editorDialogbox.show();
                    break;
                case R.id.fatInput:
                    BodyMeasure lastFatMeasure = mDbBodyMeasure.getLastBodyMeasures(fatBodyPart.getId(), getProfile());
                    final Value lastFatValue = getValueFromLastMeasure(lastFatMeasure, Unit.PERCENTAGE, id: null, "Fat");
                    editorDialogbox = new ValuesEditorDialogbox(getActivity(), new Date(), time: "", new Value[]{lastFatValue});
                    editorDialogbox.setTitle("Add");
```



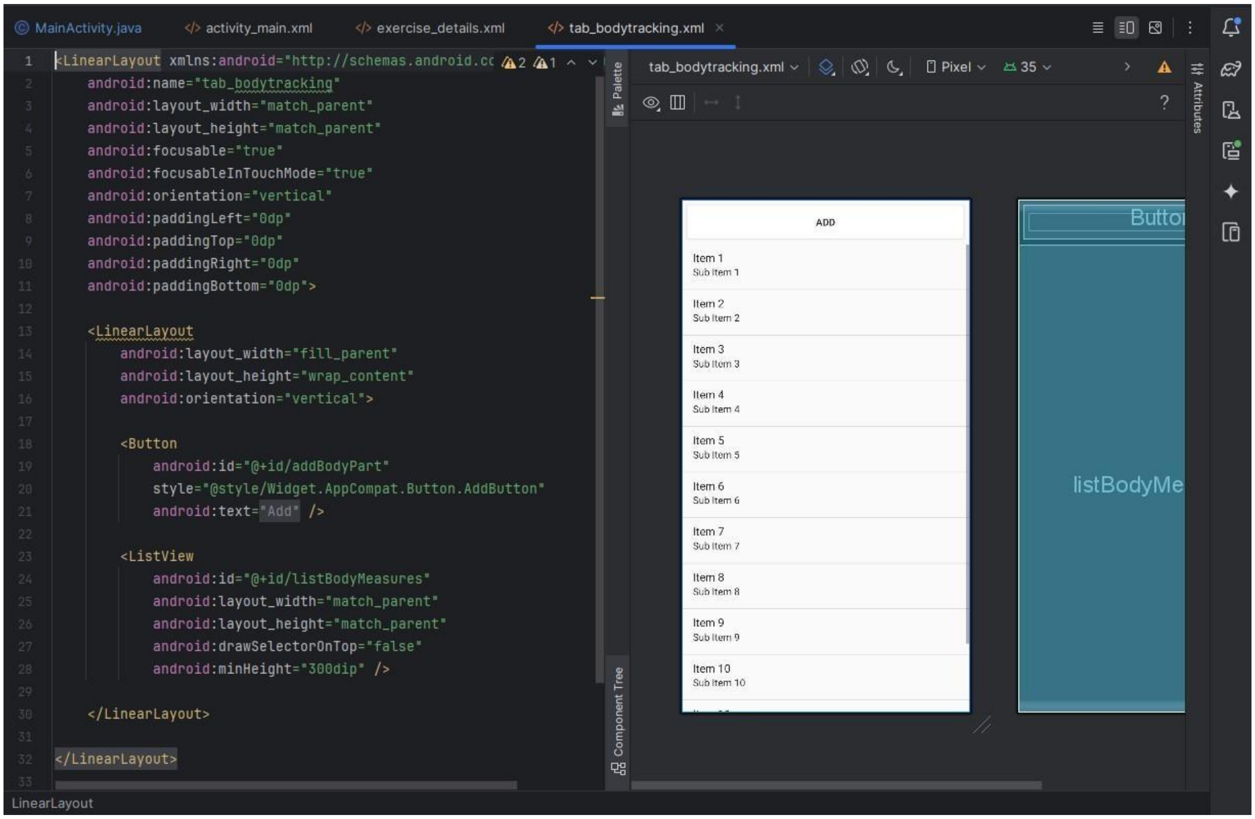
```java
public class CustomDrawerAdapter extends ArrayAdapter<DrawerItem> {
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent) {

        DrawerItemHolder drawerHolder;
        View view = convertView;

        if (view == null) {
            LayoutInflater inflater = ((Activity) context).getLayoutInflater();
            drawerHolder = new DrawerItemHolder();

            view = inflater.inflate(layoutResID, parent,  attachToRoot: false);
            drawerHolder.ItemName = view.findViewById(R.id.drawer_itemName);
            drawerHolder.icon = view.findViewById(R.id.drawer_icon);

            drawerHolder.spinner = view.findViewById(R.id.drawerSpinner);

            drawerHolder.title = view.findViewById(R.id.drawerTitle);

            drawerHolder.headerLayout = view.findViewById(R.id.headerLayout);
            drawerHolder.itemLayout = view.findViewById(R.id.itemLayout);
            drawerHolder.spinnerLayout = view.findViewById(R.id.spinnerLayout);
            drawerHolder.roundProfile = view.findViewById(R.id.header_icon);

            view.setTag(drawerHolder);

        } else {
            drawerHolder = (DrawerItemHolder) view.getTag();
```



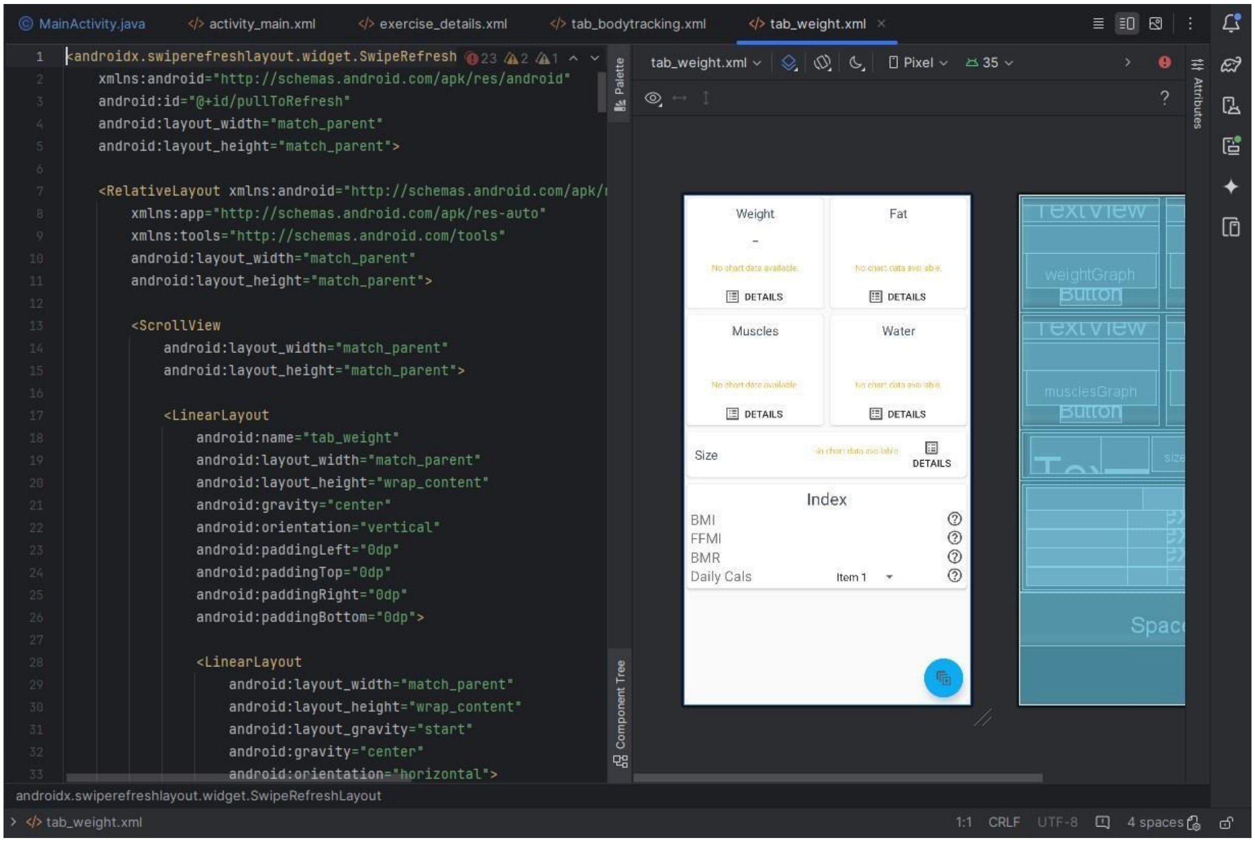
```xml
<androidx.drawerlayout.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/drawer_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <!-- The main content view -->

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="@color/background"
        android:focusableInTouchMode="true"
        android:orientation="vertical"
        android:paddingLeft="0dp"
        android:paddingTop="0dp"
        android:paddingRight="0dp"
        android:paddingBottom="0dp">

        <androidx.appcompat.widget.Toolbar
            android:id="@+id/actionToolbar"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@color/toolbar_background"
            android:elevation="4dp"
            android:minHeight="?attr/actionBarSize"
            android:visibility="visible">

            <com.mikhaellopez.circularimageview.CircularImageView
                android:id="@+id/imageProfile"
                android:layout_width="40dp"
                android:layout_height="40dp"
```



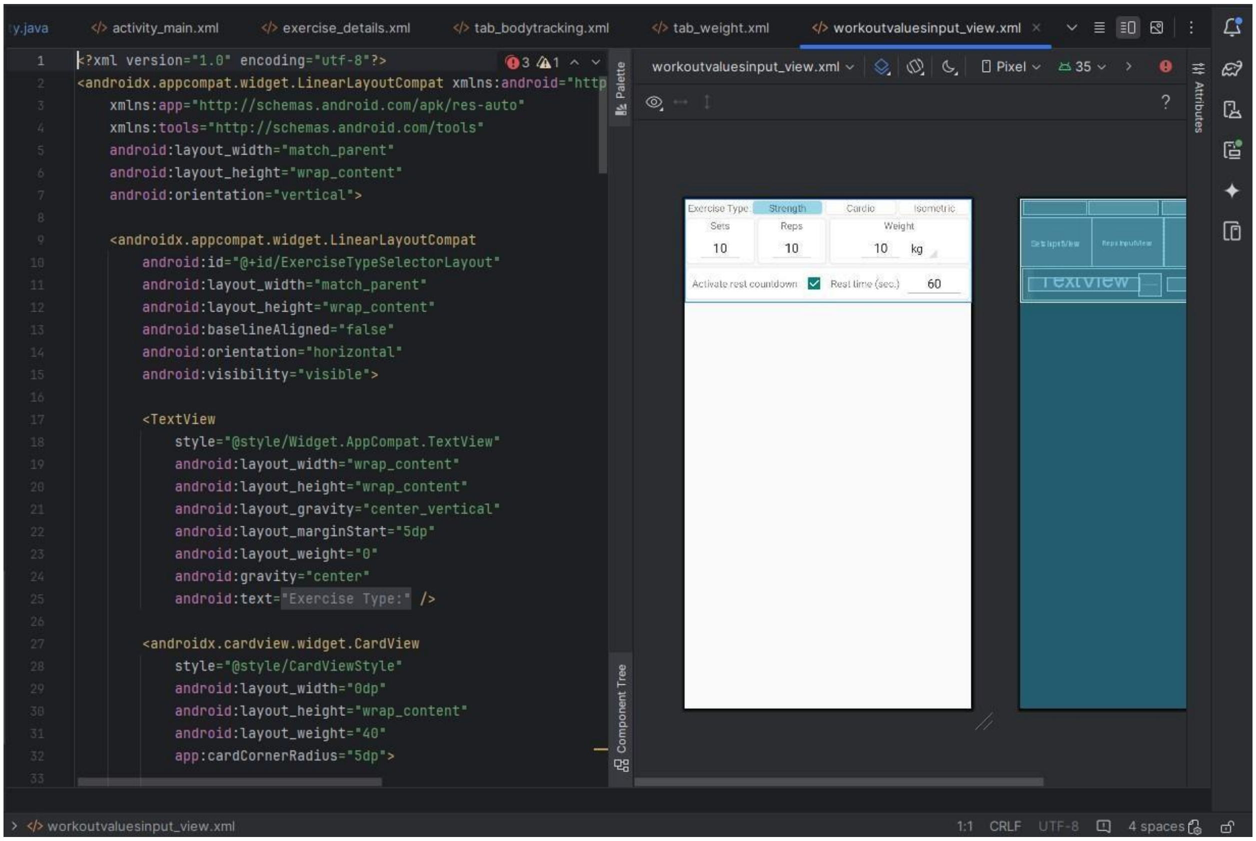
```xml
<androidx.coordinatorlayout.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <androidx.core.widget.NestedScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:paddingLeft="0dp"
        android:paddingTop="0dp"
        android:paddingRight="0dp"
        android:paddingBottom="0dp"
        app:layout_behavior="com.google.android.material.appbar.AppBarLayout$ScrollingViewBehavior">

        <LinearLayout
            android:id="@+id/machine_photo_layout"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:focusable="true"
            android:focusableInTouchMode="true"
            android:orientation="vertical"
            android:paddingLeft="0dp"
            android:paddingTop="0dp"
            android:paddingRight="0dp"
            android:paddingBottom="0dp">

            <ImageView
                android:id="@+id/machine_photo"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:adjustViewBounds="true"
                android:background="@color/record_background_even"
```



```xml
<LinearLayout xmlns:android="http://schemas.android.cc
    android:name="tab_bodytracking"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:focusable="true"
    android:focusableInTouchMode="true"
    android:orientation="vertical"
    android:paddingLeft="0dp"
    android:paddingTop="0dp"
    android:paddingRight="0dp"
    android:paddingBottom="0dp">

    <LinearLayout
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <Button
            android:id="@+id/addBodyPart"
            style="@style/Widget.AppCompat.Button.AddButton"
            android:text="Add" />

        <ListView
            android:id="@+id/listBodyMeasures"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:drawSelectorOnTop="false"
            android:minHeight="300dip" />

    </LinearLayout>

</LinearLayout>
```



```xml
<androidx.swiperefreshlayout.widget.SwipeRefresh
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/pullToRefresh"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <RelativeLayout xmlns:android="http://schemas.android.com/apk/r
        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">

        <ScrollView
            android:layout_width="match_parent"
            android:layout_height="match_parent">

            <LinearLayout
                android:name="tab_weight"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:gravity="center"
                android:orientation="vertical"
                android:paddingLeft="0dp"
                android:paddingTop="0dp"
                android:paddingRight="0dp"
                android:paddingBottom="0dp">

                <LinearLayout
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:layout_gravity="start"
                    android:gravity="center"
                    android:orientation="horizontal">
```



```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.appcompat.widget.LinearLayoutCompat xmlns:android="http
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">

    <androidx.appcompat.widget.LinearLayoutCompat
        android:id="@+id/ExerciseTypeSelectorLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:baselineAligned="false"
        android:orientation="horizontal"
        android:visibility="visible">

        <TextView
            style="@style/Widget.AppCompat.TextView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_vertical"
            android:layout_marginStart="5dp"
            android:layout_weight="0"
            android:gravity="center"
            android:text="Exercise Type:" />

        <androidx.cardview.widget.CardView
            style="@style/CardViewStyle"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="40"
            app:cardCornerRadius="5dp">

```