

PROJET ANDROID STUDIO

```

package com.example.myapplicationproject;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;

import java.util.ArrayList;
import java.util.List;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    public Button buttonrdv;
    public Button buttonplann;
    public EditText editTextNom;
    public EditText editTextPrenom;
    public Spinner spinner1;
    public EditText editTextAdresse;
    public EditText editTextMail;
    public EditText editTextTel;
    public Button buttonEnregistrer;
    ArrayList<String> listeTypes;
    Bd bd;
    int idSelected;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        buttonrdv = findViewById(R.id.button);
        buttonplann = findViewById(R.id.button2);
        editTextNom = findViewById(R.id.editTextTextPersonName2);
        editTextPrenom = findViewById(R.id.editTextTextPersonName);
        spinner1 = findViewById(R.id.spinner);
        editTextAdresse = findViewById(R.id.editTextTextPersonName11);
        editTextTel = findViewById(R.id.editTextTextPersonName12);
        buttonEnregistrer = findViewById(R.id.button3);
        editTextMail = findViewById(R.id.editTextTextPersonName3);
        bd = new Bd(this);
        majTypes();
    }

    public void majTypes() {
        try {
            listeTypes = new ArrayList<>();
            listeTypes.add("Pharmacien");
        }
    }
}

```

```

        listeTypes.add("Médecin");
        listeTypes.add("Dentiste");
        ArrayAdapter<String> aaTypes = new ArrayAdapter<>(this,
            android.R.layout.simple_spinner_item, listeTypes);
        aaTypes.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
        spinner1.setAdapter(aaTypes);
        spinner1.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> adapterView, View view,
                int i, long l) {
                idSelected = i;
            }
        });
    } catch (Exception e) {
    }
}

public void clic1(View view) {
    Intent intentAfficher = new Intent(this, MainActivity2.class);
    startActivity(intentAfficher);
}

public void clic2(View view) {
    Intent intentAfficher = new Intent(this, MainActivity3.class);
    startActivity(intentAfficher);
}

public void clic3(View view) {
    String nom, prenom, types, adresse, mail, tel;
    nom = editTextNom.getText().toString();
    prenom = editTextPrenom.getText().toString();
    types = listeTypes.get(idSelected);
    adresse = editTextAdresse.getText().toString();
    mail = editTextMail.getText().toString();
    tel = editTextTel.getText().toString();
    bd.insertDataPro(nom, prenom, types, adresse, mail, tel);
}
}

```

EXPLICATION DE CODE :

L'activité principale (MainActivity) comprend des champs de saisie pour le nom, prénom, adresse, e-mail, numéro de téléphone, ainsi qu'un bouton "Enregistrer". Deux autres boutons ("buttonrdv" et "buttonplann") déclenchent le lancement d'autres activités (MainActivity2 et MainActivity3).

Un élément clé est le Spinner permettant de choisir entre différents types de professionnels (pharmacien, médecin, dentiste). La méthode majTypes gère la mise à jour et la configuration de ce Spinner, bien qu'il soit nécessaire d'utiliser setSelectedListener au lieu de onItemClick pour gérer correctement les sélections.

Lorsque l'utilisateur clique sur "Enregistrer", les données saisies sont extraites des champs, puis insérées dans une base de données locale (bd) via la méthode insertDataPro. En résumé,

l'application vise à faciliter l'enregistrement de données professionnelles avec une structure modulaire, facilitant ainsi son évolution et sa maintenance.

CODE XML :

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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"
    tools:context=".MainActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal">

            <Button
                android:id="@+id/button"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:onClick="clic1"
                android:text="Prendre un rdv" />

            <Button
                android:id="@+id/button2"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:onClick="clic2"
                android:text="AFFICHER LE PLANNING" />
        </LinearLayout>

        <TextView
            android:id="@+id/textView"
            android:layout_width="match_parent"
            android:layout_height="43dp"
            android:gravity="center"
            android:text="ENREGISTRER UN PROFESSIONNEL" />

    </LinearLayout>
```

```
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="horizontal">
```

```
<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="29dp"
    android:layout_weight="1"
    android:text="nom :" />
```

```
<EditText
    android:id="@+id/editTextTextPersonName2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:ems="10"
    android:inputType="textPersonName"
    android:text="Name" />
```

```
</LinearLayout>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
```

```
<TextView
    android:id="@+id/textView6"
    android:layout_width="37dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="Prénom :" />
```

```
<EditText
    android:id="@+id/editTextTextPersonName"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:ems="10"
    android:inputType="textPersonName"
    android:text="Name" />
```

```
</LinearLayout>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
```

```
<TextView
    android:id="@+id/textView16"
    android:layout_width="233dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:text="type :" />
```

```
<Spinner
```

```

        android:id="@+id/spinner"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_weight="1" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/textView20"
        android:layout_width="31dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="adresse : " />

    <EditText
        android:id="@+id/editTextTextPersonName11"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="Name" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/textView4"
        android:layout_width="30dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="mail" />

    <EditText
        android:id="@+id/editTextTextPersonName3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="Name" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <TextView

```

```

        android:id="@+id/textView21"
        android:layout_width="33dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Tél" />

        <EditText
            android:id="@+id/editTextTextPersonName12"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:ems="10"
            android:inputType="textPersonName"
            android:text="Name" />
    </LinearLayout>

    <Button
        android:id="@+id/button3"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="clic3"
        android:text="ENREGISTRER" />
</LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>

```

CODE RDV JAVA :

```

package com.example.myapplicationproject;

import android.content.Intent;
import android.database.Cursor;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.CalendarView;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import java.util.ArrayList;

public class MainActivity2 extends AppCompatActivity {

```

```

public Button buttonrdv;
public Button buttonplann;
public Spinner spinnerpro;
public EditText r1;
public Button buttonEnregistrer;
public CalendarView calendar1;
public String selecdte;
public int idSelect;
ArrayList<String> listePros;
public Bd bd;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main2);

    buttonEnregistrer = findViewById(R.id.button3);
    r1 = findViewById(R.id.editTextTextPersonName6);
    spinnerpro = findViewById(R.id.spinner4);
    calendar1 = findViewById(R.id.calendarView2);
    bd = new Bd(this);

    calendar1.setOnDateChangeListener(new CalendarView.OnDateChangeListener() {
        @Override
        public void onSelectedDayChange(@NonNull CalendarView calendarView, int year, int
month, int dayOfMonth) {
            selecdte = dayOfMonth + "/" + month + "/" + year;
        }
    });

    listePros = new ArrayList<>();
    listePros = bd.getDPro();
    ArrayAdapter<String> aaTypes = new ArrayAdapter<>(this,
android.R.layout.simple_spinner_item, listePros);
    aaTypes.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
    spinnerpro.setAdapter(aaTypes);

    spinnerpro.setOnItemClickListener(new AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {
            idSelect = i;
        }

        @Override
        public void onNothingSelected(AdapterView<?> adapterView) {
        }
    });
}

public void clic(View view) {
    String heure = r1.getText().toString();
    String pro = spinnerpro.getItemAtPosition(idSelect).toString();
    bd.insertDataRDV(selecdte, heure, Integer.parseInt(pro));
}

public void clic11(View view) {

```

```

        Intent intentAfficher = new Intent(this, MainActivity.class);
        startActivity(intentAfficher);
    }

    public void clic12(View view) {
        Intent intentAfficher = new Intent(this, MainActivity3.class);
        startActivity(intentAfficher);
    }
}

```

EXPLICATION :

MainActivity2 gère la prise de rendez-vous. L'interface utilisateur inclut des éléments tels qu'un sélecteur de date (calendar1), un champ de saisie d'heure (r1), et un bouton "Enregistrer". Deux autres boutons ("buttonrdv" et "buttonplann") permettent de passer à d'autres activités (MainActivity et MainActivity3).

Lorsque l'utilisateur sélectionne une date dans le calendrier (calendar1), la méthode onSelectedDayChange est appelée pour récupérer cette date. Les professionnels disponibles sont obtenus à partir de la base de données (bd.getDPro()) et affichés dans un spinner (spinnerpro). La sélection d'un professionnel enregistre son indice dans idSelect.

Quand l'utilisateur clique sur le bouton "Enregistrer", les informations du rendez-vous, telles que la date, l'heure et l'identifiant du professionnel, sont extraites des champs correspondants et insérées dans la base de données via la méthode insertDataRDV de l'objet bd.

Les boutons "buttonrdv" et "buttonplann" permettent de naviguer vers les activités MainActivity et MainActivity3, respectivement.

Code XML RDV :

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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"
    tools:context=".MainActivity2">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"

```



```
android:orientation="vertical">
```

```
<LinearLayout
```

```
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:orientation="horizontal">
```

```
    <Button
```

```
        android:id="@+id/button5"  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:layout_weight="1"  
        android:onClick="clic11"  
        android:text="ENREGISTRER UN PROFESSIONNEL" />
```

```
    <Button
```

```
        android:id="@+id/button6"  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:layout_weight="1"  
        android:onClick="clic12"  
        android:text="AFFICHER LE PLANNING" />
```

```
</LinearLayout>
```

```
<TextView
```

```
    android:id="@+id/textView2"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:gravity="center"  
    android:text="Prendre un rdv" />
```

```
<CalendarView
```

```
    android:id="@+id/calendarView2"  
    android:layout_width="match_parent"  
    android:layout_height="301dp" />
```

```
<LinearLayout
```

```
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:orientation="horizontal" />
```

```
<LinearLayout
```

```
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:orientation="horizontal" />
```

```
<TextView
```

```
    android:id="@+id/textView8"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:gravity="center"  
    android:text="Heures " />
```

```
<EditText
```

```
    android:id="@+id/editTextTextPersonName6"  
    android:layout_width="match_parent"
```

```

        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="Name" />

<TextView
    android:id="@+id/textView7"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="TextView" />

<Spinner
    android:id="@+id/spinner4"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

<Button
    android:id="@+id/button9"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:onClick="clic"
    android:text="ENREGISTRER" />
</LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>

```

code java planning :

```

package com.example.projet_rdvgsb;

import androidx.appcompat.app.AppCompatActivity;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Spinner;
import java.util.ArrayList;
import java.util.List;

public class RecherchePro extends AppCompatActivity {
    BD bd;
    Spinner spinnerProfessionnels;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main_activity2_recherche_pro);
        bd = new BD(this);
        spinnerProfessionnels = findViewById(R.id.spinner3);
        majPro();
    }
}

```

```

public void majPro(String codeville) {
    try {
        Cursor donnee = bd.selectionnerPro(codeville);
        List<String> professionnelsList = new ArrayList<>();
        while (donnee.moveToNext()) {
            professionnelsList.add(donnee.getString(1));
        }
        ArrayAdapter<String> spinnerAdapter = new ArrayAdapter<>(
            this,
            android.R.layout.simple_spinner_item,
            professionnelsList
        );

        spinnerAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
        spinnerProfessionnels.setAdapter(spinnerAdapter);
    } catch (Exception e) {
        e.printStackTrace();
    }
}

// Méthode appelée lorsqu'un bouton est cliqué
public void AFFIC(View view) {
    Affic(codevilleP.getText().toString());
    codevilleP.setText("");
}
}

```

EXPLICATIONS :

Ce code représente une activité Android nommée **RecherchePro** dans une application de gestion de rendez-vous (RDV GSB). L'interface utilisateur inclut un spinner (**spinnerProfessionnels**) et une méthode **majPro** conçue pour mettre à jour la liste des professionnels en fonction d'un code de ville fourni en paramètre.

Lors de la création de l'activité, la base de données (**bd**) est initialisée, et le spinner est lié à son élément correspondant dans le layout. La méthode **majPro** est ensuite appelée pour actualiser la liste des professionnels affichés dans le spinner.

La méthode **majPro** prend un code de ville en paramètre, interroge la base de données pour obtenir les professionnels correspondants, puis remplit le spinner avec cette liste à l'aide d'un **ArrayAdapter**. Les noms des professionnels sont extraits du curseur et ajoutés à la liste des professionnels.

CODE XML POUR PLANNING :

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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"
    tools:context=".MainActivity3">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal">

            <Button
                android:id="@+id/button3"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:text="PRENDRE UN RDV" />

            <Button
                android:id="@+id/button7"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:text="ENREGISTRER UN PROFESSIONNEL" />
        </LinearLayout>

        <TextView
            android:id="@+id/textView10"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:text="LE Planning" />

        <CalendarView
            android:id="@+id/calendarView"
            android:layout_width="match_parent"
            android:layout_height="304dp" />

        <Button
            android:id="@+id/button10"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
```

```

        android:text="RDV de la journée" />

<Spinner
    android:id="@+id/spinner2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView5"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:text="Ville" />

<EditText
    android:id="@+id/editTextTextPersonName4"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="textPersonName"
    android:text="Name" />

<Button
    android:id="@+id/button11"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="AFFICHER" />

<Spinner
    android:id="@+id/spinner3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

</LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>

```

BASE DE DONNEES :

```

package com.example.myapplicationproject;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import java.util.ArrayList;

public class Bd extends SQLiteOpenHelper {

    // nom de la base
    public static final String DATABASE_NAME = "Db.db";

    // table professionnel

```

```

public static final String TABLE_Professionnel = "Professionnel";
public static final String COL_1 = "id_Professionnel";
public static final String COL_2 = "Nom";
public static final String COL_3 = "Prenom";
public static final String COL_4 = "LeType";
public static final String COL_5 = "Adresse";
public static final String COL_6 = "Mail";
public static final String COL_7 = "Telephone";

// table rendez-vous
public static final String TABLE_RDV = "RendezVous";
public static final String COL_1R = "id_RendezVous";
public static final String COL_2R = "Datee";
public static final String COL_3R = "Heure";
public static final String COL_4R = "Profesionnel";

// table heure disponible
public static final String TABLE_heure = "heures";
public static final String COL_1h = "id_RendezVous";
public static final String COL_2h = "Heure";

public Bd(Context context) {
    super(context, DATABASE_NAME, null, 1);
}

@Override
public void onCreate(SQLiteDatabase db) {
    db.execSQL("CREATE TABLE " + TABLE_Professionnel + " (" +
        COL_1 + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
        COL_2 + " TEXT, " +
        COL_3 + " TEXT, " +
        COL_4 + " TEXT, " +
        COL_5 + " TEXT, " +
        COL_6 + " TEXT, " +
        COL_7 + " TEXT)");

    db.execSQL("CREATE TABLE " + TABLE_RDV + " (" +
        COL_1R + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
        COL_2R + " TEXT, " +
        COL_3R + " TEXT, " +
        COL_4R + " INTEGER)");

    db.execSQL("CREATE TABLE " + TABLE_heure + " (" +
        COL_1h + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
        COL_2h + " TEXT)");
}

@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_Professionnel);
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_RDV);
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_heure);
    onCreate(db);
}

```

```

    public void insertDataPro(String Nom, String Prenom, String LeType, String Adresse, String
Mail, String Telephone) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues cv = new ContentValues();
        cv.put(COL_2, Nom);
        cv.put(COL_3, Prenom);
        cv.put(COL_4, LeType);
        cv.put(COL_5, Adresse);
        cv.put(COL_6, Mail);
        cv.put(COL_7, Telephone);
        db.insert(TABLE_Professionnel, null, cv);
        db.close();
    }

    public void insertDataRDV(String Datee, String Heure, int idProfesionnel) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues cv = new ContentValues();
        cv.put(COL_2R, Datee);
        cv.put(COL_3R, Heure);
        cv.put(COL_4R, idProfesionnel);
        db.insert(TABLE_RDV, null, cv);
        db.close();
    }

    public ArrayList<String> getDPro() {
        SQLiteDatabase db = this.getReadableDatabase();
        Cursor result = db.rawQuery("SELECT * FROM " + TABLE_Professionnel, null);
        ArrayList<String> listePros = new ArrayList<>();
        while (result.moveToNext())
            listePros.add(String.valueOf(result.getInt(0)));
        db.close();
        return listePros;
    }

    public void insertHeure(String heure) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues cv = new ContentValues();
        cv.put(COL_2h, heure);
        db.insert(TABLE_heure, null, cv);
        db.close();
    }

    public Cursor getheure(String datee) {
        SQLiteDatabase db = this.getReadableDatabase();
        Cursor result = null;
        try {
            result = db.rawQuery("SELECT heure FROM " + TABLE_heure +
                " WHERE heure NOT IN (SELECT heure FROM " + TABLE_RDV +
                " WHERE datee = '" + datee + "')", null);
        } catch (Exception e) {
            String message = e.getMessage();
        }
        db.close();
        return result;
    }
}

```

```

public Cursor getDate(String datee) {
    SQLiteDatabase db = this.getReadableDatabase();
    Cursor result = db.rawQuery("SELECT datee FROM " +
        TABLE_RDV + " WHERE datee = '" + datee + "'", null);
    db.close();
    return result;
}

public Cursor selectionnerPro(String codeVilleP) {
    SQLiteDatabase bd = this.getReadableDatabase();
    Cursor result = bd.rawQuery("SELECT * FROM " + TABLE_NAME +
        " WHERE ville_codep LIKE '%" + codeVilleP + "%'", null);
    return result;
}

public Cursor getville(String uneAdresse) {
    SQLiteDatabase db = this.getReadableDatabase();
    Cursor result = db.rawQuery("SELECT Nom FROM " +
        TABLE_Professionnel + " WHERE Adresse Like '%" + uneAdresse +
        "%'", null);
    db.close();
    return result;
}
}

```

EXPLICATIONS :

onCreate(SQLiteDatabase db) : Cette méthode est invoquée lors de la création de la base de données. Elle établit la structure des tables (Professionnel, RendezVous, heures) et peut éventuellement initialiser des données (qui sont actuellement commentées).

- **onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) :** Cette méthode est appelée lorsqu'une mise à jour de la base de données est nécessaire. Elle supprime les anciennes tables et appelle **onCreate** pour recréer la base de données.
- **insertDataPro :** Insère des données dans la table Professionnel.
- **insertDataRDV :** Insère des données dans la table RendezVous.
- **getDPro :** Récupère les identifiants des professionnels.
- **insertHeure :** Insère des heures disponibles dans la table heures.
- **getheure :** Récupère les heures disponibles pour une date donnée.
- **getDate :** Récupère les dates de rendez-vous pour une date donnée.

- `selectionnerPro`: Sélectionne les professionnels en fonction d'un code de ville.
- `getville`: Récupère le nom des professionnels en fonction d'une adresse.

TEST :

```
package com.example.projet_rdvgsb;

import android.content.Context;
import android.database.Cursor;
import androidx.test.core.app.ApplicationProvider;
import androidx.test.ext.junit.runners.AndroidJUnit4;
import org.junit.After;
import org.junit.Before;
import org.junit.Test;
import org.junit.runner.RunWith;

import static org.junit.Assert.assertEquals;
import static org.junit.Assert.assertFalse;
import static org.junit.Assert.assertNotNull;
import static org.junit.Assert.assertTrue;

@RunWith(AndroidJUnit4.class)
class BdTest {

    private Bd dbHelper;

    @Before
    public void setUp() {
        Context context = ApplicationProvider.getApplicationContext();
        dbHelper = new BD(context);
    }

    @After
    public void tearDown() {
        dbHelper.close();
    }

    @Test
    public void testEnregPro() {
        dbHelper.enregPro("John", "Doe", "john.doe@example.com", "123456789", "123 Main St", 1,
"12345");
        Cursor cursor = dbHelper.getAllDataPro();
        assertNotNull(cursor);
        assertTrue(cursor.moveToFirst());
        assertEquals("John", cursor.getString(cursor.getColumnIndex(Bd.nomP)));
        // Add more assertions based on your data model
    }
}
```

```

@Test
public void testEnregRDV() {
    dbHelper.enregRDV("2023-01-01", 10, 12, "Checkup", 1);
    Cursor cursor = dbHelper.getAllDataRDV();
    assertNotNull(cursor);
    assertTrue(cursor.moveToFirst());
    assertEquals("2023-01-01", cursor.getString(cursor.getColumnIndex(Bd.dateRDV)));
    // Add more assertions based on your data model
}

@Test
public void testGetRDVByDate() {
    dbHelper.enregRDV("2023-01-01", 10, 12, "Checkup", 1);
    Cursor cursor = dbHelper.getRDVByDate("2023-01-01");
    assertNotNull(cursor);
    assertTrue(cursor.moveToFirst());
    assertEquals("Checkup", cursor.getString(cursor.getColumnIndex(Bd.motifR)));
    // Add more assertions based on your data model
}

@Test
public void testRecherchProfromId() {
    dbHelper.enregPro("John", "Doe", "john.doe@example.com", "123456789", "123 Main St", 1,
"12345");
    String result = dbHelper.recherchProfromId(1);
    assertNotNull(result);
    assertEquals("John Doe", result);
}
// Add more tests for other methods as needed
}

```