

CODE SOURCE

ACCUEIL :

INTERFACE :



XML :

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="#D3D3F7"
    android:gravity="center"
    android:padding="16dp">

    <!-- Titre -->
    <TextView
        android:id="@+id/textViewAccueil"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Accueil"
        android:textSize="32sp"
        android:textColor="#000000"
        android:fontFamily="sans-serif-medium"
        android:textStyle="bold"
        android:gravity="center"
        android:layout_marginBottom="24dp" />

    <!-- Bouton 1 : Enregistrer un professionnel -->
    <Button

        android:id="@+id/buttonEnregistrerProfessionnel"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Enregistrer un professionnel"
        android:textColor="#000000"
        android:backgroundTint="#E8E8F8"
        android:layout_marginBottom="16dp"
        android:padding="12dp" />

    <!-- Bouton 2 : Prendre un RDV -->
    <Button
        android:id="@+id/buttonPrendreRDV"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Prendre un RDV"
        android:textColor="#000000"
        android:backgroundTint="#E8E8F8"
        android:layout_marginBottom="16dp"
        android:padding="12dp" />

    <!-- Bouton 3 : Afficher le planning d'une journée -->
    <Button
        android:id="@+id/buttonAfficherPlanning"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Afficher le planning d'une journée"
        android:textColor="#000000"
        android:backgroundTint="#E8E8F8"
        android:layout_marginBottom="16dp"
        android:padding="12dp" />
```

```

<!-- Bouton 4 : Afficher les professionnels -->
<Button
    android:id="@+id/buttonAfficherProfessionnels"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Afficher les professionnels"
    android:textColor="#000000"
    android:backgroundTint="#E8E8F8"
    android:padding="12dp" />

</LinearLayout>

```

JAVA :

```

package com.example.androidprojetfin;

import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

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

        // Initialisation des boutons
        Button buttonEnregistrerProfessionnel = findViewById(R.id.buttonEnregistrerProfessionnel);
        Button buttonPrendreRDV = findViewById(R.id.buttonPrendreRDV);
        Button buttonAfficherPlanning = findViewById(R.id.buttonAfficherPlanning);
        Button buttonAfficherProfessionnels = findViewById(R.id.buttonAfficherProfessionnels);

        // Gestionnaire pour "Enregistrer un professionnel"
        buttonEnregistrerProfessionnel.setOnClickListener(v -> {
            Intent intent = new Intent(MainActivity.this, ProjetAndroidEnregPro.class);
            startActivity(intent);
        });

        // Gestionnaire pour "Prendre un RDV"
        buttonPrendreRDV.setOnClickListener(v -> {
            Intent intent = new Intent(MainActivity.this, com.example.gremdic.PrendreRendezVous.class);
            startActivity(intent);
        });

        // Gestionnaire pour "Afficher le planning d'une journée"
        buttonAfficherPlanning.setOnClickListener(v -> {
            Intent intent = new Intent(MainActivity.this, ProjetAndroidPlanning.class);
            startActivity(intent);
        });

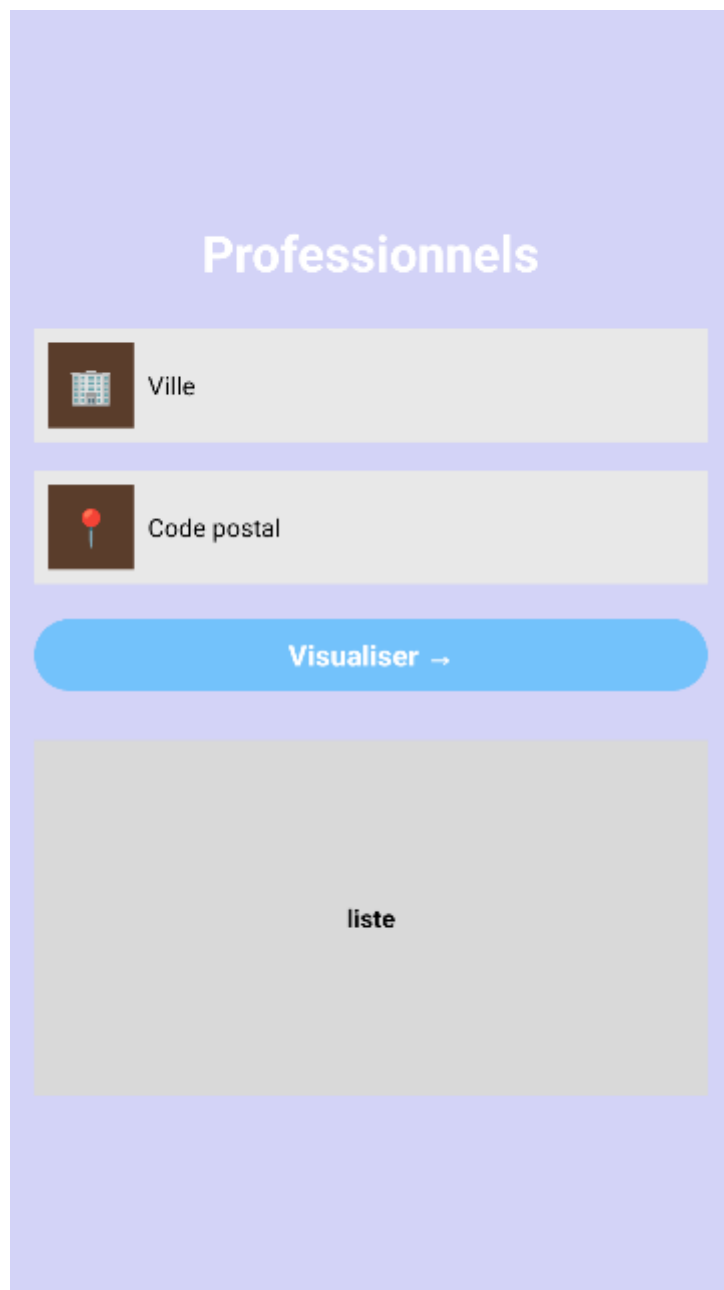
        // Gestionnaire pour "Afficher les professionnels"

```

```
buttonAfficherProfessionnels.setOnClickListener(v -> {  
    Intent intent = new Intent(MainActivity.this, com.example.gremdic.AfficherProfesionnels.class);  
    startActivity(intent);  
});  
}  
}
```

Afficher Pro :

INTERFACE :



XML :

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp"
    android:gravity="center"
    android:background="#D3D3F7">
```

```
<!-- Titre -->
```

```
<TextView
    android:id="@+id/textViewTitre"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Professionnels"
    android:textColor="#FFFFFF"
    android:textSize="28sp"
    android:textStyle="bold"
    android:layout_marginBottom="24dp"/>
```

```
<!-- Bloc Ville -->
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:background="#E8E8E8"
    android:layout_marginBottom="16dp"
    android:padding="8dp">
```

```
<TextView
    android:layout_width="48dp"
    android:layout_height="48dp"
    android:text="□"
    android:textSize="20sp"
    android:gravity="center"
    android:background="#5A3D2B"
    android:textColor="#FFFFFF"
    android:padding="8dp" />
```

```
<TextView
    android:id="@+id/textViewVille"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center_vertical"
    android:text="Ville"
    android:textColor="#000000"
    android:paddingStart="8dp"/>
```

```
</LinearLayout>
```

```
<!-- Bloc Code Postal -->
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:background="#E8E8E8"
    android:layout_marginBottom="16dp"
    android:padding="8dp">
```

```
<TextView
    android:layout_width="48dp"
    android:layout_height="48dp"
```

```

        android:text="□"
        android:textSize="20sp"
        android:gravity="center"
        android:background="#5A3D2B"
        android:textColor="#FFFFFF"
        android:padding="8dp" />

<TextView
    android:id="@+id/textViewCodePostal"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center_vertical"
    android:text="Code postal"
    android:textColor="#000000"
    android:paddingStart="8dp"/>
</LinearLayout>

<!-- Bouton Visualiser -->
<Button
    android:id="@+id/buttonVisualiser"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Visualiser →"
    android:textColor="#FFFFFF"
    android:backgroundTint="#73C2FB"
    android:textStyle="bold"
    android:textSize="16sp"
    android:layout_marginBottom="24dp" />

<!-- Bloc Liste -->
<TextView
    android:id="@+id/textViewListe"
    android:layout_width="match_parent"
    android:layout_height="200dp"
    android:background="#D9D9D9"
    android:text="liste"
    android:textColor="#000000"
    android:textStyle="bold"
    android:gravity="center"
    android:layout_gravity="center"/>

</LinearLayout>

```

JAVA :

```

package com.example.androidprojetfin;

import android.content.Intent;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

```

```

import androidx.appcompat.app.AppCompatActivity;

import com.example.androidprojetfin.MainActivity;

/**
 * Classe principale pour afficher les professionnels en fonction de la ville ou du code postal.
 * Permet de rechercher et d'afficher les informations des professionnels.
 */
public class AfficherProfesionnels extends AppCompatActivity {
    EditText ville, cp;
    TextView resultat;
    BD db;

    /**
     * Méthode appelée lors de la création de l'activité.
     * Initialise les éléments de l'interface utilisateur pour la recherche de professionnels.
     *
     * @param savedInstanceState L'état précédent de l'activité.
     */
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_profesionnels);

        // Initialisation des vues
        ville = findViewById(R.id.tbVillePro);
        cp = findViewById(R.id.tbCpPro);
        resultat = findViewById(R.id.labelResultatPro);

        // Initialisation de la base de données
        db = new BD(this);
    }

    /**
     * Cette méthode est appelée lors du clic sur le bouton pour visualiser les professionnels.
     * Elle recherche les professionnels en fonction de la ville ou du code postal et les affiche.
     * Si les deux champs sont remplis, un message d'erreur est affiché.
     *
     * @param view La vue qui a déclenché l'événement.
     */
    public void visualiserPro(View view) {
        StringBuilder liste = new StringBuilder();

        // Vérifie que seulement un champ (ville ou code postal) est renseigné
        if (ville.length() > 0 && cp.length() > 0) {
            Toast.makeText(this, "Veuillez saisir une seule information", Toast.LENGTH_LONG).show();
            return;
        }

        // Recherche par ville
        if (ville.length() > 0) {
            try {
                // Récupère les professionnels par ville
                Cursor data = db.getProByVille(ville.getText().toString());
                while (data.moveToNext()) {
                    String nom = data.getString(0);
                    String prenom = data.getString(1);
                    liste.append(nom).append(" : ").append(prenom).append("\n");
                }
                resultat.setText(liste.toString());
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    }
}

```

```

    }
    // Recherche par code postal
    else if (cp.length() > 0) {
        try {
            // Récupère les professionnels par code postal
            Cursor data = db.getProByCp(cp.getText().toString());
            while (data.moveToNext()) {
                String nom = data.getString(0);
                String prenom = data.getString(1);
                liste.append(nom).append(" : ").append(prenom).append("\n");
            }
            resultat.setText(liste.toString());
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

/**
 * Cette méthode permet de retourner à la page d'accueil.
 *
 * @param view La vue qui a déclenché l'événement.
 */
public void retourPro(View view) {
    Intent btnRetourPro = new Intent(this, MainActivity.class);
    startActivity(btnRetourPro);
}

/**
 * Cette méthode permet de se déconnecter et revenir à la page de connexion.
 *
 * @param view La vue qui a déclenché l'événement.
 */
}

```

Enregistrer Pro :

INTERFACE :

Enregistrement Professionnel

Nom

Prénom

Type

Adresse

Mail

Téléphone

Code Postal

Retour

XML :

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/Téléphone"
```

```
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="#D3D3F7"
android:gravity="center"
android:orientation="vertical"
android:padding="20dp">
```

```
<TextView
    android:id="@+id/tv_inscription"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="20dp"
    android:gravity="center"
    android:text="Enregistrement Professionnel"
    android:textColor="#000000"
    android:textSize="24sp"
    android:textStyle="bold" />
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="10dp"
    android:gravity="center_vertical"
    android:orientation="horizontal">
```

```
<EditText
    android:id="@+id/nom_pro"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:padding="20dp"
    android:text="Nom" />
```

```
</LinearLayout>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="10dp"
    android:gravity="center_vertical"
    android:orientation="horizontal">
```

```
<EditText
    android:id="@+id/prenom_pro"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:padding="20dp"
    android:text="Prénom" />
```

```
</LinearLayout>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="10dp"
    android:gravity="center_vertical"
    android:orientation="horizontal">
```

```
<EditText
    android:id="@+id/type_pro"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
```

```

        android:padding="20dp"
        android:text="Type" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="10dp"
    android:gravity="center_vertical"
    android:orientation="horizontal">

    <EditText
        android:id="@+id/adresse_pro"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:padding="20dp"
        android:text="Adresse" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="10dp"
    android:gravity="center_vertical"
    android:orientation="horizontal">

    <EditText
        android:id="@+id/mail_pro"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:padding="20dp"
        android:text="Mail" />
</LinearLayout>

<!-- Téléphone -->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="10dp"
    android:gravity="center_vertical"
    android:orientation="horizontal">

    <EditText
        android:id="@+id/tel_pro"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="20dp"
        android:text="Téléphone" />
</LinearLayout>

<!-- Code postal -->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="10dp"
    android:gravity="center_vertical"
    android:orientation="horizontal">

    <EditText
        android:id="@+id/cp_pro"
        android:layout_width="0dp"

```

```

        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:padding="20dp"
        android:text="Code Postal" />
    </LinearLayout>

    <Button
        android:id="@+id/buttonRetour"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Retour" />
</LinearLayout>

```

JAVA :

```

package com.example.androidprojetfin;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

public class ProjetAndroidEnregPro extends AppCompatActivity {
    BD db;
    EditText nomPro, prenomPro, typePro, adressePro, mailPro, telPro, cpPro;

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

        db = new BD(this);
        nomPro = findViewById(R.id.nom_pro);
        prenomPro = findViewById(R.id.prenom_pro);
        typePro = findViewById(R.id.type_pro);
        adressePro = findViewById(R.id.adresse_pro);
        mailPro = findViewById(R.id.mail_pro);
        telPro = findViewById(R.id.tel_pro);
        cpPro = findViewById(R.id.cp_pro);
    }

    public void confirmationAccount(View view) {
        // Vérification si tous les champs sont remplis
        if (nomPro.getText().toString().isEmpty() ||
            prenomPro.getText().toString().isEmpty() ||
            typePro.getText().toString().isEmpty() ||
            adressePro.getText().toString().isEmpty() ||
            cpPro.getText().toString().isEmpty() ||
            mailPro.getText().toString().isEmpty() ||
            telPro.getText().toString().isEmpty()) {

            Toast.makeText(this, "Veuillez remplir tous les champs", Toast.LENGTH_LONG).show();
            return;
        }
    }
}

```

```

    }

    // Insertion des données dans la base de données
    try {
        db.insertProfessionnel(
            nomPro.getText().toString(),
            prenomPro.getText().toString(),
            typePro.getText().toString(),
            adressePro.getText().toString(),
            cpPro.getText().toString(),
            mailPro.getText().toString(),
            telPro.getText().toString()
        );
        Toast.makeText(this, "Professionnel enregistré avec succès", Toast.LENGTH_LONG).show();
        // Rediriger l'utilisateur vers la page d'accueil
        Intent intent = new Intent(this, MainActivity.class);
        startActivity(intent);
    } catch (Exception e) {
        Toast.makeText(this, "Erreur lors de l'enregistrement", Toast.LENGTH_LONG).show();
        e.printStackTrace(); // Afficher l'exception dans les logs
    }
}
}

```

Planning :

INTERFACE :

Planning

Choisissez un professionnel :

Item 1

Sélectionnez une date :

| < Janvier 1970 > | | | | | | |
|------------------|----|----|----|----|----|----|
| L | M | M | J | V | S | D |
| | | | 1 | 2 | 3 | 4 |
| 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 26 | 27 | 28 | 29 | 30 | 31 | |

Valider

Retour

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:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".projet_android_planning">
    <?xml version="1.0" encoding="utf-8"?>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:background="#D3D3F7">

        <TextView
            android:id="@+id/tv_planning"
```

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="30dp"
    android:gravity="center"
    android:text="Planning"
    android:textColor="#000000"
    android:textSize="28sp"
    android:textStyle="bold" />
```

<TextView

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="10dp"
    android:text="Choisissez un professionnel :"
    android:textColor="#000000"
    android:textSize="18sp" />
```

<Spinner

```
    android:id="@+id/spinner_professionnel"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="20dp"
    android:backgroundTint="#FFFFFF"
    android:padding="10dp" />
```

<TextView

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="10dp"
    android:text="Sélectionnez une date :"
    android:textColor="#000000"
    android:textSize="18sp" />
```

<CalendarView

```
    android:id="@+id/calendar_view"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="30dp"
    android:background="#FFFFFF" />
```

<Button

```
    android:id="@+id/btn_valider"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="#6200EE"
    android:padding="10dp"
    android:text="Valider"
    android:textColor="#FFFFFF"
    android:textSize="16sp"
    android:textStyle="bold" />
```

<Button

```
    android:id="@+id/buttonRetour"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Retour" />
```

```
<!-- Bouton Retour -->
```

```
</LinearLayout> <!-- Fond mauve -->
```

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

JAVA :

```
package com.example.androidprojetfin;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.CalendarView;
import android.widget.Spinner;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;

/**
 * Activité qui permet de gérer le planning avec un calendrier et d'afficher les événements
 * pour une date sélectionnée.
 *
 * Cette activité permet à l'utilisateur de sélectionner une date via un calendrier, de choisir un
 * professionnel dans un Spinner, et d'afficher la date sélectionnée.
 */
public class ProjetAndroidPlanning extends AppCompatActivity {

    /** Vue représentant le calendrier pour sélectionner une date. */
    private CalendarView calendar;

    /** Vue représentant le spinner pour sélectionner un professionnel. */
    private Spinner spinnerProfessionnel;

    /** Vue pour afficher la date sélectionnée dans le calendrier. */
    private TextView resultat;

    /**
     * Méthode appelée lors de la création de l'activité. Elle initialise les vues de l'interface et configure les écouteurs
     pour les interactions avec les composants.
     *
     * @param savedInstanceState État précédent de l'activité. Cela permet de restaurer l'état de l'activité si nécessaire.
     */
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_projet_android_planning);

        calendar = findViewById(R.id.calendar_view);
        spinnerProfessionnel = findViewById(R.id.spinner_professionnel);
        resultat = findViewById(R.id.tv_planning);

        calendar.setOnDateChangeListener((calendarView, year, month, dayOfMonth) -> {
            String curDate = year + "/" + (month + 1) + "/" + dayOfMonth;

```



```

        // Affichage de la date sélectionnée
        resultat.setText("Date sélectionnée: " + curDate);
    });

    Button btnRetour = findViewById(R.id.buttonRetour);
    btnRetour.setOnClickListener(view -> {
        // Démarre l'activité Accueil
        Intent btnRetourPlanning = new Intent(this, Accueil.class);
        startActivity(btnRetourPlanning);
    });

    Button btnValider = findViewById(R.id.btn_valider);
    btnValider.setOnClickListener(view -> {

    });
}
}

```

Prendre RDV :

INTERFACE :

Prendre un rendez-vous

Nom

Prénom

Type

Adresse

Mail

Téléphone

ou

Retour

XML :

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

```
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:background="#D3D3F7"
android:padding="20dp"
android:gravity="center">
```

```
<!-- Titre -->
```

```
<TextView
    android:id="@+id/tv_inscription"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="20dp"
    android:gravity="center"
    android:text="Prendre un rendez-vous"
    android:textColor="#000000"
    android:textSize="24sp"
    android:textStyle="bold" />
```

```
<!-- Nom -->
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_marginBottom="10dp"
    android:gravity="center_vertical">
```

```
<EditText
    android:id="@+id/rdv_nom"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:hint="Nom"
    android:background="@drawable/rounded_edittext"
    android:padding="10dp" />
```

```
</LinearLayout>
```

```
<!-- Prénom -->
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_marginBottom="10dp"
    android:gravity="center_vertical">
```

```
<EditText
    android:id="@+id/rdv_prenom"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:hint="Prénom"
    android:background="@drawable/rounded_edittext"
    android:padding="10dp" />
```

```
</LinearLayout>
```

```
<!-- Type -->
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_marginBottom="10dp"
    android:gravity="center_vertical">
```

```
<EditText
    android:id="@+id/rdv_type"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:hint="Type"
    android:background="@drawable/rounded_edittext"
    android:padding="10dp" />
</LinearLayout>

<!-- Adresse -->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_marginBottom="10dp"
    android:gravity="center_vertical">

    <EditText
        android:id="@+id/rdv_adresse"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:hint="Adresse"
        android:background="@drawable/rounded_edittext"
        android:padding="10dp" />
</LinearLayout>

<!-- Mail -->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_marginBottom="10dp"
    android:gravity="center_vertical">

    <EditText
        android:id="@+id/rdv_mail"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:hint="Mail"
        android:background="@drawable/rounded_edittext"
        android:padding="10dp" />
</LinearLayout>

<!-- Téléphone -->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_marginBottom="10dp"
    android:gravity="center_vertical">

    <EditText
        android:id="@+id/rdv_telephone"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:hint="Téléphone"
        android:background="@drawable/rounded_edittext"
        android:padding="10dp" />
```

```

</LinearLayout>

<!-- Mot de passe -->
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:layout_marginBottom="20dp"
    android:gravity="center_vertical"/>

<!-- Bouton Créer le compte -->

<!-- Texte ou -->
<TextView
    android:id="@+id/tv_ou"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="10dp"
    android:gravity="center"
    android:text="ou"
    android:textColor="#000000" />

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

<!-- Bouton Retour -->

</LinearLayout>

```

JAVA :

```

package com.example.gremdic;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.CalendarView;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

/**
 * Activité principale pour la gestion des rendez-vous.
 * Permet de sélectionner une date, saisir l'heure, le motif et l'ID professionnel pour enregistrer un rendez-vous.
 */
public class PrendreRendezVous extends AppCompatActivity {
    BD db;
    String curDate;
    CalendarView date;
    EditText heure, motif, idPro;

    /**

```

```

* Méthode appelée lors de la création de l'activité.
* Initialise les éléments d'interface et le calendrier.
*
* @param savedInstanceState État précédent de l'activité.
*/
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_rendez_vous);
    db = new BD(this);
    date = findViewById(R.id.calendarView);
    heure = findViewById(R.id.tbHeure);
    motif = findViewById(R.id.tbMotif);
    idPro = findViewById(R.id.tbIdPro);

    // Écouteur pour la sélection d'une date sur le calendrier
    date.setOnDateChangeListener((calendarView, year, month, dayOfMonth) -> {
        curDate = dayOfMonth + "/" + (month + 1) + "/" + year; // Correction du mois (indexé à partir de 0)
    });
}

/**
 * Enregistre un rendez-vous dans la base de données.
 *
 * @param view La vue qui a déclenché l'événement.
 */
public void enregistrerRDV(View view) {
    try {
        int lheure = Integer.parseInt(heure.getText().toString());
        int lidPro = Integer.parseInt(idPro.getText().toString());
        db.insertDataRdv(curDate, lheure, motif.getText().toString(), lidPro);
        Toast.makeText(this, "Le rendez-vous a bien été pris", Toast.LENGTH_LONG).show();
    } catch (NumberFormatException e) {
        Toast.makeText(this, "Veuillez entrer des valeurs valides pour l'heure et l'ID professionnel",
Toast.LENGTH_LONG).show();
    }
}

/**
 * Permet de revenir à l'écran d'accueil.
 *
 * @param view La vue qui a déclenché l'événement.
 */
public void retourRDV(View view) {
    Intent btnRetourRDV = new Intent(this, Accueil.class);
    startActivity(btnRetourRDV);
}

/**
 * Permet de se déconnecter et revenir à l'écran de connexion.
 *
 * @param view La vue qui a déclenché l'événement.
 */
public void disconnectRDV(View view) {
    Intent btnDisconnectRDV = new Intent(this, Connexion.class);
    startActivity(btnDisconnectRDV);
}
}

```

BD classe :

```
package com.example.androidprojetfin;

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

/**
 * Classe de gestion de la base de données pour l'application Gère'Médecin.
 *
 * Cette classe facilite l'accès aux données des professionnels et des rendez-vous.
 */
public class BD extends SQLiteOpenHelper {

    private static final String NOM_BD = "Gère'Médecin"; // Nom de la base de données
    private static final int VERSION_BD = 1; // Version de la base de données

    // Table Professionnel
    private static final String TABLE_PROFESSIONNEL = "Professionnel";
    private static final String COL_ID_PRO = "Id_Professionnel";
    private static final String COL_NOM = "nom";
    private static final String COL_PRENOM = "prenom";
    private static final String COL_ADRESSE = "adresse";
    private static final String COL_CP = "codePostal";
    private static final String COL_TEL = "tel";
    private static final String COL_MAIL = "mail";

    // Table RDV
    private static final String TABLE_RDV = "RDV";
    private static final String COL_ID_RDV = "Id_RDV";
    private static final String COL_DATE = "laDate";
    private static final String COL_HEURE = "heure";
```

```

private static final String COL_MOTIF = "motif";
private static final String COL_ID_PRO_FK = "Id_Professionnel";

/**
 * Constructeur de la classe BD.
 *
 * @param context Contexte de l'application.
 */
public BD(Context context) {
    super(context, NOM_BD, null, VERSION_BD);
}

/**
 * Méthode appelée lors de la création de la base de données.
 *
 * @param db Base de données.
 */
@Override
public void onCreate(SQLiteDatabase db) {
    // Création de la table Professionnel
    String CREATE_TABLE_PROFESSIONNEL = "CREATE TABLE " +
        TABLE_PROFESSIONNEL + "(" +
        COL_ID_PRO + " INTEGER PRIMARY KEY AUTOINCREMENT," +
        COL_NOM + " TEXT," +
        COL_PRENOM + " TEXT," +
        COL_ADRESSE + " TEXT," +
        COL_CP + " TEXT," +
        COL_TEL + " TEXT," +
        COL_MAIL + " TEXT)";
    db.execSQL(CREATE_TABLE_PROFESSIONNEL);

    // Création de la table RDV avec une clé étrangère vers Professionnel
    String CREATE_TABLE_RDV = "CREATE TABLE " + TABLE_RDV + "(" +
        COL_ID_RDV + " INTEGER PRIMARY KEY AUTOINCREMENT," +
        COL_DATE + " TEXT," +
        COL_HEURE + " TEXT," +
        COL_MOTIF + " TEXT," +
        COL_ID_PRO_FK + " INTEGER," +
        "FOREIGN KEY(" + COL_ID_PRO_FK + ") REFERENCES " +
        TABLE_PROFESSIONNEL + "(" + COL_ID_PRO + ")";
    db.execSQL(CREATE_TABLE_RDV);
}

/**
 * Méthode appelée lors de la mise à jour de la base de données.
 *
 * @param db Base de données.
 * @param oldVersion Ancienne version de la base de données.
 * @param newVersion Nouvelle version de la base de données.
 */
@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_RDV);
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_PROFESSIONNEL);
    onCreate(db); // Recréation des tables
}

/**
 * Insère un professionnel dans la base de données.
 *
 * @param string
 * @param nom Le nom du professionnel.
 * @param prenom Le prénom du professionnel.

```



```

    * @param adresse L'adresse du professionnel.
    * @param codePostal Le code postal du professionnel.
    * @param mail L'adresse email du professionnel.
    * @param tel Le numéro de téléphone du professionnel.
    */
    public void insertProfessionnel(String string, String nom, String prenom, String adresse, String codePostal, String
mail, String tel) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(COL_NOM, nom);
        values.put(COL_PRENOM, prenom);
        values.put(COL_ADRESSE, adresse);
        values.put(COL_CP, codePostal);
        values.put(COL_MAIL, mail);
        values.put(COL_TEL, tel);
        db.insert(TABLE_PROFESSIONNEL, null, values);
        db.close();
    }

    /**
     * Insère un rendez-vous dans la base de données.
     *
     * @param laDate La date du rendez-vous.
     * @param heure L'heure du rendez-vous.
     * @param motif Le motif du rendez-vous.
     * @param idPro L'ID du professionnel associé au rendez-vous.
     */
    public void insertRendezVous(String laDate, String heure, String motif, int idPro) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(COL_DATE, laDate);
        values.put(COL_HEURE, heure);
        values.put(COL_MOTIF, motif);
        values.put(COL_ID_PRO_FK, idPro);
        db.insert(TABLE_RDV, null, values);
        db.close();
    }

    /**
     * Récupère tous les rendez-vous pour une date donnée.
     *
     * @param laDate La date des rendez-vous à récupérer.
     * @return Un curseur contenant les informations des rendez-vous.
     */
    public Cursor getRendezVousByDate(String laDate) {
        SQLiteDatabase db = this.getReadableDatabase();
        return db.rawQuery("SELECT " + COL_HEURE + ", " + COL_MOTIF + " FROM " + TABLE_RDV + " WHERE "
+ COL_DATE + " = ?", new String[]{laDate});
    }

    /**
     * Récupère les professionnels par code postal.
     *
     * @param cp Le code postal des professionnels à récupérer.
     * @return Un curseur contenant les informations des professionnels.
     */
    public Cursor getProfessionnelsByCp(String cp) {
        SQLiteDatabase db = this.getReadableDatabase();
        return db.rawQuery("SELECT * FROM " + TABLE_PROFESSIONNEL + " WHERE " + COL_CP + " = ?", new
String[]{cp});
    }

    /**

```

```
* Récupère les professionnels par adresse.  
*  
* @param adresse L'adresse des professionnels à récupérer.  
* @return Un curseur contenant les informations des professionnels.  
*/  
public Cursor getProfessionnelsByAdresse(String adresse) {  
    SQLiteDatabase db = this.getReadableDatabase();  
    return db.rawQuery("SELECT * FROM " + TABLE_PROFESSIONNEL + " WHERE " + COL_ADRESSE + " = ?",  
new String[]{adresse});  
}  
}
```