



TP N°3

Intents et Dialogues

1. Dialogs

Il s'agit des boîtes de dialogue pour alerter l'utilisateur en cas d'erreur ou pour l'informer ou pour choisir entre différents choix.

Les classes de Dialog sont :

- **AlertDialog** : les fenêtres de dialogue pour faire les choix
- **ProgressDialog** : Pour montrer la progression d'une tâche, il peut contenir des boutons.
- **DatePickerDialog** : permet à l'utilisateur de choisir une date
- **TimePickerDialog** : permet à l'utilisateur de choisir le temps

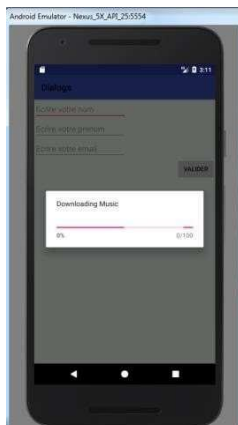
Exemple1 : Afficher un **AlertDialog** si les champs sont vides

```
fun valider(v: View?) {  
    if (nom.getText().length() == 0 || prenom.getText().length() == 0 ||  
        email.getText()  
            .length() == 0  
    ) {  
        val ad: AlertDialog.Builder  
        ad = AlertDialog.Builder(this)  
        ad.setMessage("Les champs ne doivent pas être vide")  
        ad.setTitle("Error")  
        ad.setIcon(android.R.drawable.btn_dialog)  
        ad.setPositiveButton(  
            "yes"  
        ) { dialogInterface, i -> finish() }  
        val a = ad.create()  
        a.show()  
    }  
}
```



Exemple 2 :ProgressDialog

```
val progressDialog = ProgressDialog(this@MainActivity)
progressDialog.setTitle("Kotlin Progress Dialog")
progressDialog.setProgressStyle(ProgressDialog.STYLE_HORIZONTAL)
progressDialog.setMessage("Downloading music, please wait")
progress.setIndeterminate(true)
progress.setProgress(0)
progressDialog.show()
```



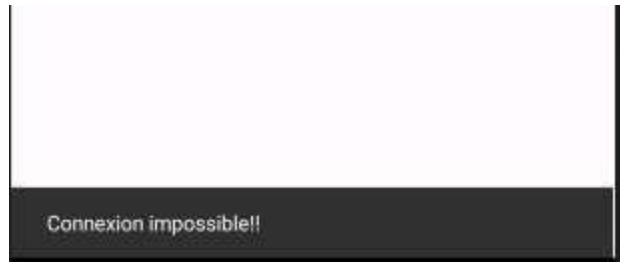
2. Toasts

Afficher un message sans interrompre l'activité. Il est utilisé pour informer l'utilisateur qu'une tâche est accomplie ou non sans perdre le focus sur l'activité. Il n'est pas utilisé pour les messages critiques.

```
Toast.makeText(this, "Toast Message", Toast.LENGTH_LONG).show();
```

3. Snackbar

Il s'affiche en bas de l'écran pour donner un feedback sur les opérations en cours.



```
val snack = Snackbar.make(it, "Un Snackbar simple ", Snackbar.LENGTH_LONG)
snack.show()
```

4. WebView

Permet d'insérer un navigateur web dans votre application.

```
//Instanciation WebView
webView.webViewClient = WebViewClient()
// ProgressBar disparaît à la fin du chargement de la page

// chargement de la page à partir de son URL
webView.loadUrl("https://www.google.com/")

// autoriser les scripts javascripts
webView.settings.javaScriptEnabled = true

// accepter le zoom sur la page
webView.settings.setSupportZoom(true)
```

Pour lier un progress bar pour le chargement de la page

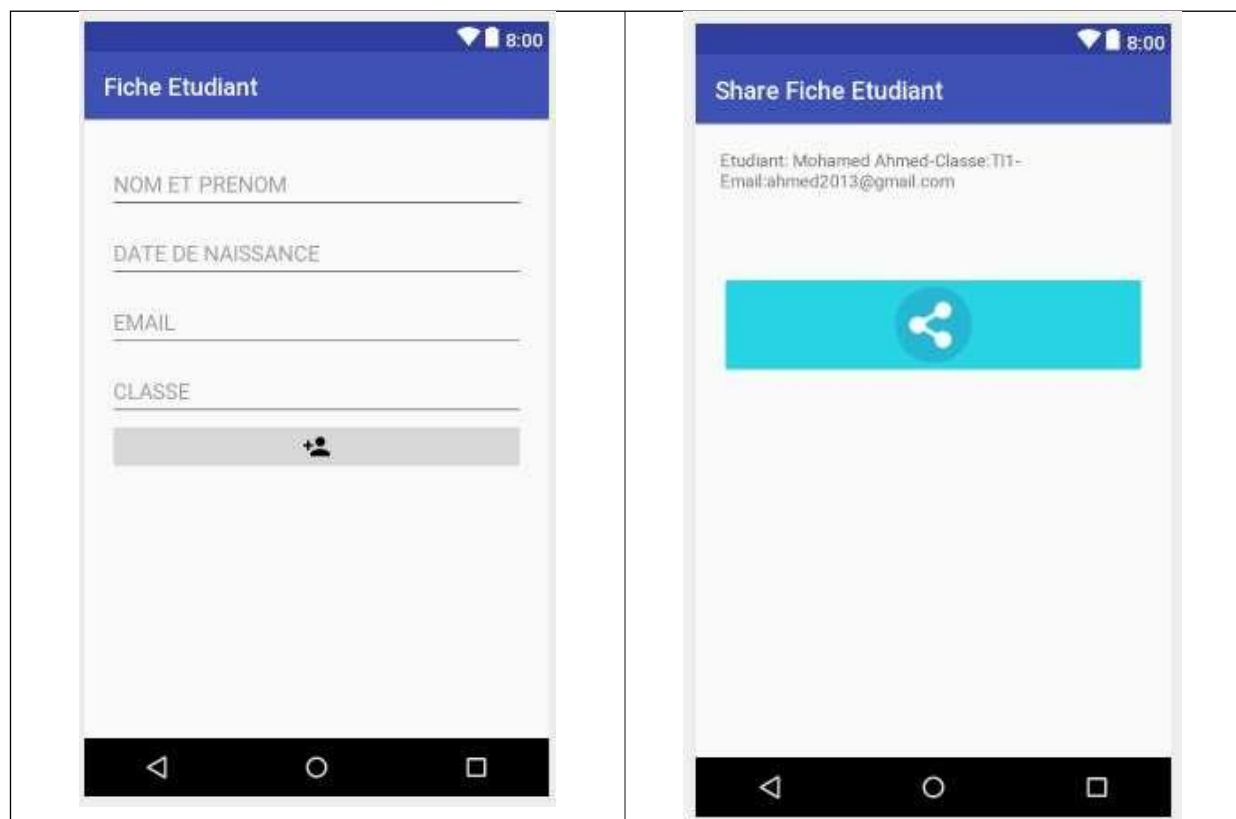
Ajouter une classe interne :

```
inner class WebViewClient : android.webkit.WebViewClient() {

    // Charger l'URL
    override fun shouldOverrideUrlLoading(view: WebView, url: String): Boolean {
        view.loadUrl(url)
        return false
    }
    // Faire disparaître le progressBar après le chargement de la page
    override fun onPageFinished(view: WebView, url: String) {
        super.onPageFinished(view, url)
        progressBar.visibility = View.GONE
    }
}
```

5. Travail à faire

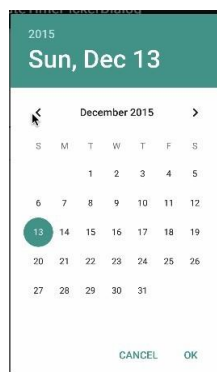
L'objectif de l'application est de développer l'application suivante :



1. A partir de la première activité, on remplit les 4 champs : Nom, Date_Naissance, Email, Classe. (EditText), utiliser un coordinatorLayout.
 - Si tous les champs sont non vide on passe à la deuxième activité où on affiche le Nom, l'email et la classe
 - Avant de passer à la deuxième activité, afficher un alertDialog pour confirmer les données enregistrées.
 - Si un des champs est vide, on affiche un message d'erreur :
 - a) Dans un Toast
 - b) Dans un Snackbar
2. Dans la deuxième activité, si on clique sur le bouton Share, le système nous propose de partager les données envoyées via les applications de partage disponibles sur le téléphone. Utiliser un intent Implicite et l'action **ACTION_SEND**.
3. Remplacer EditText Classe par un spinner remplie à partir de la liste suivante

```
<resources>
    <string-array name="classe">
        <item>DSI2</item>
        <item>DSI3</item>
        <item>RSI2</item>
        <item>RSI3</item>
        <item>SEM2</item>
        <item>SEM3</item>
    </string-array>
</resources>
```

4. Afficher dans le champs Date un **DatePickerDialog** pour choisir la date.



5. Utiliser pour chaque EditText un **TextInputLayout**

Fiche Etudiant

NOM ET PRENOM

DATE DE NAISSANCE

EMAIL

DSI2

+

```
android.support.design.widget.TextInputLayout
    android:id="@+id/input_layout_nom"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <EditText
        android:id="@+id/id_nom"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="NOM ET PRENOM"
        android:inputType="" />
</android.support.design.widget.TextInputLayout>
```

6. Supprimer le toast et le snackbar afficher les messages d'erreurs en utilisant le
TextInputLayout