

# 07-Performing-Update-Operation-on-Room

Clique droit sur res > New > Image Asset

-> Ajouter un icône edit, un icône delete et un icône search

Mise à jour du layout activity\_new\_book.xml : Ajout d'un bouton cancel :

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="28dp"
    android:gravity="center"
    android:orientation="horizontal"
    app:layout_constraintTop_toBottomOf="@+id/etBookName"
    tools:layout_editor_absoluteX="0dp">

    <Button
        android:id="@+id/bCancel"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginRight="20px"
        android:text="@string/cancel"
        android:textColor="@color/white" />

    <Button
        android:id="@+id/bAdd"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:hint="@string/save_button"
        android:textColor="@color/white"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.688"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/etBookName" />

</LinearLayout>
```

Ajouter la loupe dans la top-bar, menu\_main.xml :

```

<item
    android:id="@+id/search"
    android:title="@string/search"
    android:icon="@drawable/ic_search"
    app:showAsAction="collapseActionView|ifRoom"
    app:actionViewClass="android.widget.SearchView"
/>

```

Pour configurer la recherche : [Lien](#)

Créer une nouvelle activity vide : `EditBookActivity`

Update de `NewBookActivity` :

```

bCancel.setOnClickListener {
    finish()
}

```

Penser à vérifier le manifest après la création d'une activité (AndroidManifest.xml)

Dans le `BookListAdapter.kt`, ajouter :

```

override fun onBindViewHolder(holder: BookViewHolder, position: Int) {
    val book = bookList[position]
    holder.setData(book.author, book.book, position)
    holder.setListeners()
}

```

et :

```

fun setListeners () {
    itemView.ivRowEdit.setOnClickListener {}
    itemView.ivRowDelete.setOnClickListener {}
}

```

Implémenter la fonction a exécuté lors du clique sur Edit :

```

itemView.ivRowEdit.setOnClickListener {
    val intent = Intent(context, EditBookActivity::class.java)
    intent.putExtra("id", bookList[pos].id)
    intent.putExtra("author", bookList[pos].author)
    intent.putExtra("book", bookList[pos].book)
    (context as Activity).startActivityForResult(
        intent,

```

```
        MainActivity.UPDATE_BOOK_REQUEST_CODE  
    )  
}
```

Ajouter le code UPDATE\_BOOK\_REQUEST\_CODE dans le MainActivity :

```
const val UPDATE_BOOK_REQUEST_CODE = 2
```