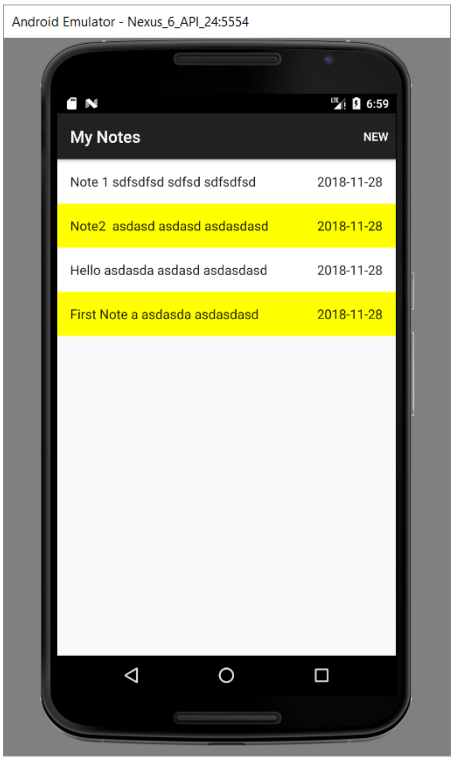
Lab 9: My Notes

In this lab we will be implementing the below application which provides note taking.

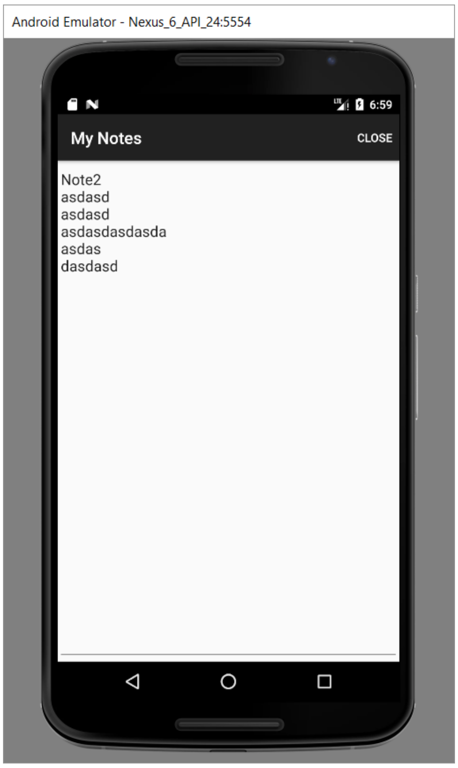


Android Emulator - Nexus_6_API_24:5554

6:59

My Notes NEW

Note 1 sdfsdfsd sdfsd sdfsdfsd	2018-11-28
Note2 asdasd asdasd asdasdasd	2018-11-28
Hello asdasda asdasd asdasdasd	2018-11-28
First Note a asdasda asdasdasd	2018-11-28



Android Emulator - Nexus_6_API_24:5554

6:59

My Notes CLOSE

Note2
asdasd
asdasd
asdasdasdasda
asdas
dasdasd

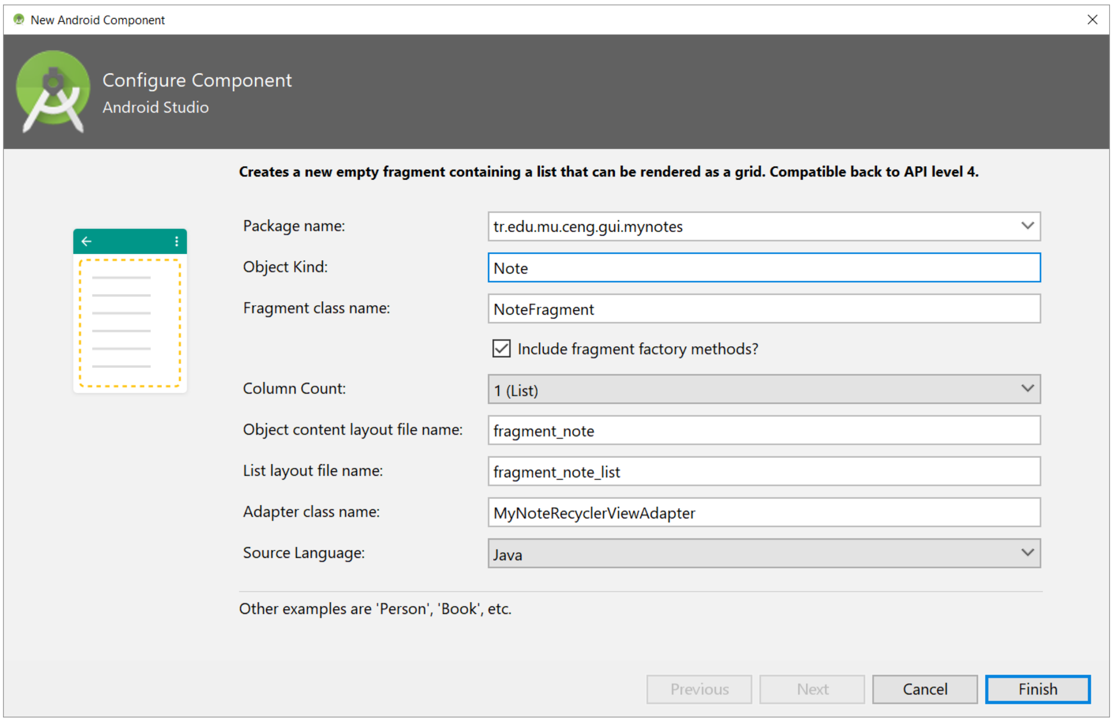
1. Create new project in Android Studio with an empty activity.
2. Select defaults for minimum SDK
3. Open the activity_main.xml and modify it as shown below.

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.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">
<FrameLayout
    android:id="@+id/container"
    android:layout_width="0dp"
    android:layout_height="0dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent">

</FrameLayout>
</android.support.constraint.ConstraintLayout>
```

4. Create the Note class shown below and generated getters and setters and implement the Serializable interface

```
public class Note {

    private String header;
    private Date date;
    private String filePath;

}
```

5. Create a fragment for listing notes based on Fragment (List) as shown below



Creates a new empty fragment containing a list that can be rendered as a grid. Compatible back to API level 4.

Package name: tr.edu.mu.ceng.gui.mynotes

Object Kind: Note

Fragment class name: NoteFragment

☑ Include fragment factory methods?

Column Count: 1 (List)

Object content layout file name: fragment_note

List layout file name: fragment_note_list

Adapter class name: MyNoteRecyclerViewAdapter

Source Language: Java

Other examples are 'Person', 'Book', etc.

6. Modify the fragment_note as shown below

```
<?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="wrap_content"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/note_header"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:layout_margin="@dimen/text_margin"
        android:textAppearance="?android:attr/textAppearanceListItem" />
    <TextView
        android:id="@+id/note_date"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="@dimen/text_margin"
        android:textAppearance="?android:attr/textAppearanceListItem" />
</LinearLayout>
```

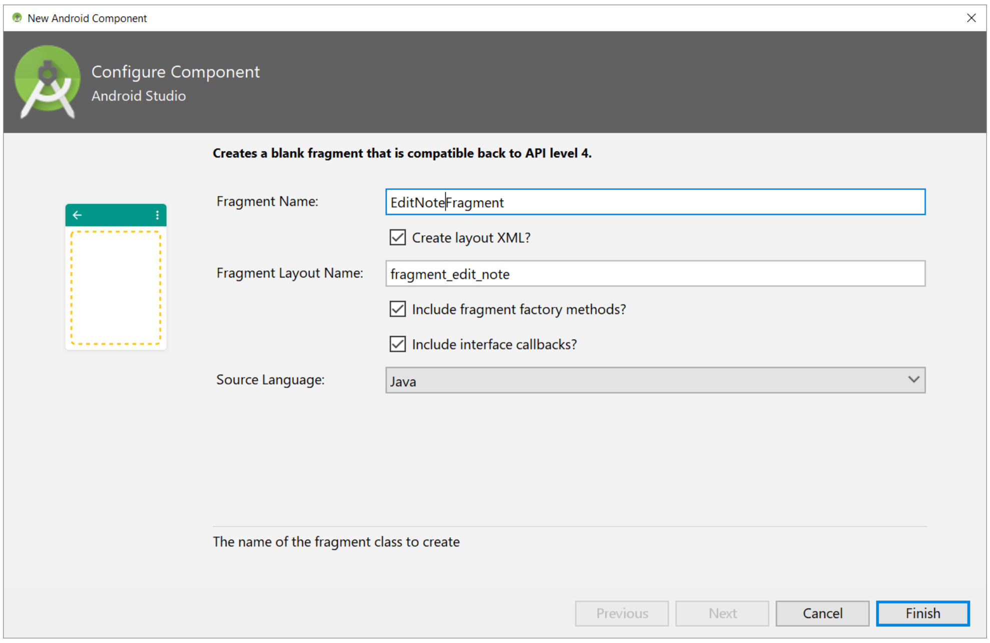
7. Modify the NoteFragment as shown below

```
public class NoteFragment extends Fragment {
    private static final String ARG_NOTES = "notes";
    private OnNoteListInteractionListener mListener;
    private ArrayList<Note> notes;

    public NoteFragment() {}
    public static NoteFragment newInstance(ArrayList<Note> notes) {
        NoteFragment fragment = new NoteFragment();
        Bundle args = new Bundle();
        args.putSerializable(ARG_NOTES, notes);
        fragment.setArguments(args);
        return fragment;
    }
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            notes = (ArrayList<Note>)getArguments().getSerializable(ARG_NOTES);
        }
    }
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        View view = inflater.inflate(R.layout.fragment_note_list, container, false);
        // Set the adapter
        if (view instanceof RecyclerView) {
            Context context = view.getContext();
            RecyclerView recyclerView = (RecyclerView) view;
            recyclerView.setLayoutManager(new LinearLayoutManager(context));
            recyclerView.setAdapter(new MyNoteRecyclerViewAdapter(notes, mListener));
        }
        return view;
    }
    @Override
    public void onAttach(Context context) {
        super.onAttach(context);
        if (context instanceof OnNoteListInteractionListener) {
            mListener = (OnNoteListInteractionListener) context;
        } else {
            throw new RuntimeException(context.toString()
                    + " must implement OnNoteListInteractionListener");
        }
    }
    @Override
    public void onDetach() {
        super.onDetach();
        mListener = null;
    }
    /**
     * Interface for listing note operations in the list
     */
    public interface OnNoteListInteractionListener {
        void onNoteSelected(Note item);
    }
}
```

8. Modify the Adapter as shown below

```
public class MyNoteRecyclerViewAdapter extends
RecyclerView.Adapter<MyNoteRecyclerViewAdapter.ViewHolder> {
    private final List<Note> mValues;
    private final OnNoteListInteractionListener mListener;
    public MyNoteRecyclerViewAdapter(List<Note> notes,
NoteFragment.OnNoteListInteractionListener listener) {
        mValues = notes;
        mListener = listener;
    }
    @Override
    public ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
        View view = LayoutInflater.from(parent.getContext())
                .inflate(R.layout.fragment_note, parent, false);
        return new ViewHolder(view);
    }
    @Override
    public void onBindViewHolder(final ViewHolder holder, int position) {
        holder.mItem = mValues.get(position);
        holder.mHeaderView.setText(mValues.get(position).getHeader());
        holder.mDateView.setText((new SimpleDateFormat("yyyy-MM-dd")).
                format(mValues.get(position).getDate()));
        holder.mView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (null != mListener) {
                    mListener.onNoteSelected(holder.mItem);
                }
            }
        });
        if(position %2 == 1) {
            holder.itemView.setBackgroundColor(Color.YELLOW);
        }
        else{
            holder.itemView.setBackgroundColor(Color.WHITE);
        }
    }
    @Override
    public int getItemCount() {
        return mValues.size();
    }
    public class ViewHolder extends RecyclerView.ViewHolder {
        public final View mView;
        public final TextView mHeaderView;
        public final TextView mDateView;
        public Note mItem;

        public ViewHolder(View view) {
            super(view);
            mView = view;
            mHeaderView =  view.findViewById(R.id.note_header);
            mDateView =  view.findViewById(R.id.note_date);
        }

        @Override
        public String toString() {
            return super.toString() + " '" + mHeaderView.getText() + "'";
        }
    }
}
```

9. Create an EditNote fragment based on the Blank fragment template



Creates a blank fragment that is compatible back to API level 4.

Fragment Name: EditNoteFragment

☑ Create layout XML?

Fragment Layout Name: fragment_edit_note

☑ Include fragment factory methods?

☑ Include interface callbacks?

Source Language: Java

The name of the fragment class to create

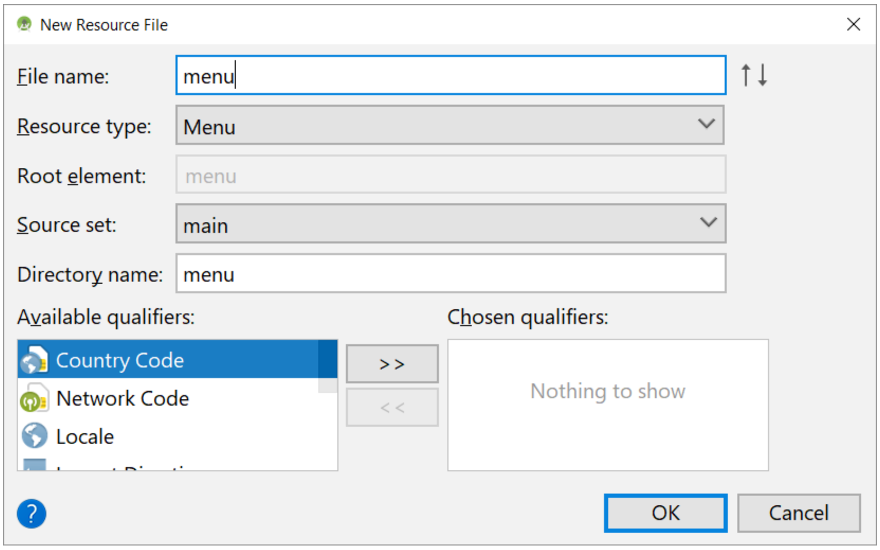
10. Modify the EditNoteFragment as shown below.

```
public class EditNoteFragment extends Fragment {

    private static final String ARG_NOTE = "content";
    private String content;
    private EditText txtContent;
    public EditNoteFragment() {}
    public static EditNoteFragment newInstance(String content) {
        EditNoteFragment fragment = new EditNoteFragment();
        Bundle args = new Bundle();
        args.putString(ARG_NOTE, content);
        fragment.setArguments(args);
        return fragment;
    }
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            content = (String)getArguments().getString(ARG_NOTE);
        }
    }
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                        Bundle savedInstanceState) {
        return inflater.inflate(R.layout.fragment_edit_note, container, false);
    }
    @Override
    public void onViewCreated(View view, @Nullable Bundle savedInstanceState) {
        super.onViewCreated(view, savedInstanceState);
        txtContent = view.findViewById(R.id.note_content);
        if (content != null) {
            txtContent.setText(content);
        }
    }
    public String getContent(){
        return txtContent.getText().toString();
    }
}
```

11. Modify the fragment_edit_note.xml as shown below

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".EditNoteFragment">

    <EditText
        android:id="@+id/note_content"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:inputType="textMultiLine"
        android:gravity="top"
        android:hint="Note..." />
</FrameLayout>
```

12. Create a new menu for the Activity



New Resource File

File name: menu

Resource type: Menu

Root element: menu

Source set: main

Directory name: menu

Available qualifiers:

Chosen qualifiers:

13. Modify the menu.xml as shown below

```
<?xml version="1.0" encoding="utf-8"?>
<menu
xmlns:android="http://schemas.android.com/apk/res/android
">
    <item
        android:id="@+id/action_new"
        android:showAsAction="always"
        android:title="New" />
    <item
        android:id="@+id/action_close"
        android:showAsAction="always"
        android:visible="false"
        android:title="Close" />
</menu>
```

14. Modify the MainActivity as below

```
public class MainActivity extends Activity implements
NoteFragment.OnNoteListInteractionListener {
    boolean displayingEditor = false;
    Note editingNote;
    ArrayList<Note> notes;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        notes = retrieveNotes();
        Log.d("onCreate",  "Note Count = " +notes.size());
        if (!displayingEditor){
            FragmentTransaction ft =getFragmentManager().beginTransaction();
            ft.add(R.id.container,NoteFragment.newInstance(notes));
            ft.commit();
        }else{
            FragmentTransaction ft = getFragmentManager().beginTransaction();
ft.replace(R.id.container,EditNoteFragment.newInstance(readContent(editingNote)));
            ft.addToBackStack(null);
            ft.commit();
        }
    }
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        MenuInflater inflater = getMenuInflater();
        inflater.inflate(R.menu.menu, menu);
        return super.onCreateOptionsMenu(menu);
    }
    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        Log.d("onOptionsItemSelected", item.getTitle().toString());
        displayingEditor = !displayingEditor;
        invalidateOptionsMenu();
        switch (item.getItemId()) {
            case R.id.action_new:
                editingNote = createNote();
                notes.add(editingNote);
                FragmentTransaction ft = getFragmentManager().beginTransaction();
                ft.replace(R.id.container,EditNoteFragment.newInstance(""),"edit_note");
                ft.addToBackStack(null);
                ft.commit();
                return true;
            case R.id.action_close:
                onBackPressed();
                return true;
            default:
                return super.onOptionsItemSelected(item);
        }
    }

    public boolean onPrepareOptionsMenu(Menu menu){
        Log.d("onPrepareOptionsMenu new visible",
menu.findItem(R.id.action_new).isVisible() + "");
        menu.findItem(R.id.action_new).setVisible(!displayingEditor);
        menu.findItem(R.id.action_close).setVisible(displayingEditor);

        return super.onPrepareOptionsMenu(menu);
    }
    public ArrayList<Note> retrieveNotes(){
        ArrayList<Note> notes  = new ArrayList<>();
        File dir = getFilesDir();
        File[] files = dir.listFiles();
        for (File file : files){
            Log.d("Retrieving", "absolute path = " + file.getAbsolutePath());
            Log.d("Retrieving", "name = " + file.getName());
            Note note = new Note();
            note.setFilePath(file.getAbsolutePath());
            note.setDate(new Date(file.lastModified()));
            String header =
getPreferences(Context.MODE_PRIVATE).getString(file.getName(),"No Header!");
            note.setHeader(header);
            notes.add(note);
        }
        return notes;
    }
    @Override
    public void onBackPressed() {
        EditNoteFragment editFragment = (EditNoteFragment)
getFragmentManager().findFragmentByTag("edit_note");
        if (editFragment != null){
            String content = editFragment.getContent();
            saveContent(editingNote, content);
        }
        super.onBackPressed();
    }
    @Override
    public void onNoteSelected(Note note) {
        editingNote =note;
        FragmentTransaction ft = getFragmentManager().beginTransaction();
ft.replace(R.id.container,EditNoteFragment.newInstance(readContent(editingNote)),"edit
_note");
        ft.addToBackStack(null);
        ft.commit();
        displayingEditor = !displayingEditor;
        invalidateOptionsMenu();
    }
    private Note createNote() {
        Note note = new Note();
        SharedPreferences pref = getPreferences(Context.MODE_PRIVATE);

        int next = pref.getInt("next",1);
        File dir = getFilesDir();
        String filePath = dir.getAbsolutePath()+"/note_"+next;
        Log.d("Create Note with path",filePath);
        note.setFilePath(filePath);

        SharedPreferences.Editor editor = pref.edit();
        editor.putInt("next", next+1);
        editor.commit();

        return note;
    }
    private void saveContent(Note note, String content) {
        note.setDate(new Date());
        String header = content.length() < 30 ? content : content.substring(0,30);
        note.setHeader(header.replaceAll("\n", " "));

        FileWriter writer = null;
        File file = new File(note.getFilePath());
        try {
            writer = new FileWriter(file);
            writer.write(content);
        } catch (IOException e) {
            e.printStackTrace();
        }finally {
            if (writer != null) {
                try {
                    writer.close();
                } catch (IOException e) {
                    e.printStackTrace();
                }
            }
        }
        SharedPreferences.Editor editor = getPreferences(Context.MODE_PRIVATE).edit();
        Log.d("Saving tp Pref","key = " + file.getName() +" value = " + note.getHeader());
        editor.putString(file.getName(),note.getHeader());
        editor.commit();
    }
    private String readContent(Note note) {
        Log.d("Readin Note with path",note.getFilePath());
        StringBuffer content = new StringBuffer();

        try (BufferedReader reader = new BufferedReader(new FileReader(new
File(note.getFilePath())))) {
            String line;
            while ((line = reader.readLine()) != null){
                content.append(line).append("\n");
            }
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        }
        return content.toString();
    }
}
```

15. Run the application