

MainActivity.xml

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout 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="wrap_content"
    android:orientation="vertical"
    android:padding="30dp"
    tools:context=".MainActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:paddingTop="20dp">

        <ImageView
            android:id="@+id/imageView"
            android:layout_width="100dp"
            android:layout_height="150dp"
            android:background="@color/black" />

        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="15dp"
```

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

```
<TextView
```

```
    android:id="@+id/titleTv"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:text="Title"
```

```
    android:ellipsize="end"
```

```
    android:maxLines="1"
```

```
 />
```

```
<TextView
```

```
    android:id="@+id/descTv"
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:text="Description"
```

```
    android:ellipsize="end"
```

```
    android:maxLines="2" />
```

```
<TextView
```

```
    android:id="@+id/authorTv"
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_marginTop="15dp"
```

```
    android:text="Author "
```

```
 />
```

```
<TextView
```

```
    android:id="@+id/dateTv"
```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="15dp"
        android:text=" Date "

    />
</LinearLayout>
</LinearLayout>
</LinearLayout>

```

Recycler_view.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"

    android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:app="http://schemas.android.com/apk/res-auto">
    <EditText

        android:id="@+id/categoryTv"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter the Category"
        android:textSize="30dp"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        android:imeOptions="actionSend"
        android:inputType="text"/>

    <androidx.recyclerview.widget.RecyclerView

```

```
    android:id="@+id/recyclerView"
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:scrollbars="vertical"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintTop_toBottomOf="@id/categoryTv"/>

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

MainActivity.java

```
package com.example.newsapp;

import ...

public class MainActivity extends AppCompatActivity {

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.recycler_view);

        RecyclerView recyclerView = findViewById(R.id.recyclerView);

        ArrayList<MyDataModel> tasks = new ArrayList<MyDataModel>();

        RecyclerView.LayoutManager layoutManager = new LinearLayoutManager(this);

        recyclerView.setLayoutManager(layoutManager);

        RecyclerViewAdapter adapter = new RecyclerViewAdapter(this, tasks);

        recyclerView.setAdapter(adapter);

        EditText categoryTv = findViewById(R.id.categoryTv);

        categoryTv.setOnEditorActionListener(new TextView.OnEditorActionListener() {

            @Override

            public boolean onEditorAction(TextView v, int actionId, KeyEvent event) {

                if ((event != null && (event.getKeyCode() == KeyEvent.KEYCODE_ENTER)) ||

                    (actionId == EditorInfo.IME_ACTION_DONE)) {
```

```
Log.i(TAG, "Enter pressed");

String category = categoryTv.getText().toString();

String apiUrl = "https://inshorts.deta.dev/news?category=" + category;


RequestQueue requestQueue = Volley.newRequestQueue(MainActivity.this);


StringRequest stringRequest = new StringRequest(
    Request.Method.GET,
    apiUrl,
    new Response.Listener<String>() {
        @Override
        public void onResponse(String response) {
            String jsonData = response;
            Log.d("JSON DATA:", jsonData);

            jsonArrayDecode(jsonData, adapter);
        }
    }, new Response.ErrorListener() {
        @Override
        public void onErrorResponse(VolleyError error) {
            Log.d("error", error.toString());
            View view = findViewById(R.id.recyclerView);
            Snackbar.make(view, "Please Check your connection",
Snackbar.LENGTH_LONG).show();
        }
    }
);

requestQueue.add(stringRequest);
```

```

    }
    return false;
}
});

```

```
String apiUrl = "https://inshorts.deta.dev/news?category=all";
```

```
RequestQueue requestQueue = Volley.newRequestQueue(this);
```

```
StringRequest stringRequest = new StringRequest(
    Request.Method.GET,
    apiUrl,
    new Response.Listener<String>() {
        @Override
        public void onResponse(String response) {
            String jsonData = response;
            Log.d("JSON DATA:", jsonData);

            jsonArrayDecode(jsonData, adapter);
        }
    }, new Response.ErrorListener() {
        @Override
        public void onErrorResponse(VolleyError error) {
            Log.d("error", error.toString());
            View view=findViewById(R.id.recyclerView);

```

```

        Snackbar.make(view,"Please Check your
connection",Snackbar.LENGTH_LONG).show();

    }

}

);

requestQueue.add(stringRequest);
}

```

```

public void jsonArrayDecode(String jsonData, RecyclerViewAdapter adapter){
    ArrayList<MyDataModel> data = new ArrayList<>();
    try{
        JSONObject jsonObject = new JSONObject(jsonData);
        JSONArray posts = jsonObject.getJSONArray("data");

        for(int i=0; i<posts.length();i++){
            JSONObject post = posts.getJSONObject(i);

            // String image = post.getString("image");
            String title = post.getString("title");
            String content = post.getString("content");
            String imageUrl = post.getString("imageUrl");
            String author = post.getString("author");
            String date = post.getString("date");
            // String category = post.getString("category");

            data.add(new MyDataModel(title, content, imageUrl, author, date));

        }
        adapter.setPosts(data);
    }
}

```



```
    } catch (JSONException e) {  
        throw new RuntimeException(e);  
    }  
}  
}
```

MyDataModel.java

```
package com.example.newsapp;
```

```
public class MyDataModel {  
    private String image;  
    private String title;  
    private String content;  
    private String author;  
  
    private String date;  
  
    public MyDataModel(String title, String content, String image, String author, String date){  
  
        this.title=title;  
        this.content=content;  
        this.image=image;  
        this.author=author;  
        this.date=date;  
    }  
    public String getTitle() {return title;}  
    public String getContent() {return content;}  
    public String getImage() {return image;}  
    public String getAuthor() {return author;}}
```

```
    public String getDate() {return date;}  
}
```

RecyclerViewAdapter.java

```
package com.example.newsapp;
```

```
import ...
```

```
public class RecyclerViewAdapter extends  
RecyclerView.Adapter<RecyclerViewAdapter.ViewHolder> {
```

```
    Activity activity;
```

```
    ArrayList<MyDataModel> posts;
```

```
    public RecyclerViewAdapter(Activity activity, ArrayList<MyDataModel> posts) {  
        this.activity = activity;  
        this.posts = posts;  
    }
```

```
    public static class ViewHolder extends RecyclerView.ViewHolder {
```

```
        ImageView imageView;
```

```
        TextView titleTv, descTv, authorTv, dateTv;
```

```
        EditText categoryTv;
```

```
        public ViewHolder(@NonNull View itemView) {
```

```
            super(itemView);
```

```
            titleTv = itemView.findViewById(R.id.titleTv);
```

```
            descTv = itemView.findViewById(R.id.descTv);
```

```
            imageView = itemView.findViewById(R.id.imageView);
```

```
            authorTv = itemView.findViewById(R.id.authorTv);
```

```

        dateTv = itemView.findViewById(R.id.dateTv);

    }
}

@NonNull
@Override
public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
    LayoutInflater inflater = LayoutInflater.from(activity);
    View listItem = inflater.inflate(R.layout.activity_main, parent, false);
    ViewHolder viewHolder = new ViewHolder(listItem);
    return viewHolder;
}

@Override
public void onBindViewHolder(@NonNull RecyclerView.Adapter.ViewHolder holder, int position) {
    holder.titleTv.setText(posts.get(position).getTitle());
    holder.descTv.setText(posts.get(position).getContent());
    holder.authorTv.setText(posts.get(position).getAuthor());
    holder.dateTv.setText(posts.get(position).getDate());
    // holder.categoryTv.setText(posts.get(position).getCategory());
    String image = posts.get(position).getImage();
    if (image != null && image.length() > 0) {
        Picasso.get().load(posts.get(position).getImage()).into(holder.imageView);
    }
}
}

```

```
@Override  
public int getItemCount() {  
    return posts.size();  
}  
  
public void setPosts(ArrayList<MyDataModel> posts) {  
    this.posts = posts;  
    notifyDataSetChanged();  
}  
}
```

Output:

