**Integrating Recycler view :**

1. Add recycler view in activity\_main.xml file.
2. Find it by id in *MainActivity.java*
3. Create a layout manager for it and pass it in setLayoutManager().
4. Create layout for single row (cardView).
5. Create a model class (structure class) (Here it is *TaskModel.java*

Make a constructor of that class with parameters: title, description, time.

1. Make an ArrayList in the *MainActivity.java.* ArrayList requires an structure class. So we will pass the structure class that we made (*TaskModel.java*), and initialize that ArrayList.
2. Add data in the array list using (ArrayList name).add().
3. Now make a custom adapter by making a *RecyclerTaskAdapter.java* class.
4. Extend that adapter class by writing (adapter\_class\_name) extends RecyclerView.Adapter<VH>.
5. Make a ViewHolder class extending (having the properties of) the *RecyclerView.ViewHolder*, and make its constructor matching the super.
6. Implement (override) all the three methods of RecyclerView.Adapter class.
7. ***onCreateViewHolder method:***

* The OnCreateViewHolder method returns ViewHolder and ViewHolder class has a constructor that requires a view (itemView). So first we will create a view, i.e., inflate the view using LayoutInflater.
* Layout inflater requires a context which we can get by creating a constructor of the *RecyclerTaskAdapter* class having the parameter as context.
* Make a view like: View view = LayoutInflater.from(context).inflate(R.layout.task\_row, parent, false).

And then make a ViewHolder class object and then automatically its constructor will be called and then we will pass the above view int it.

1. Now we have to find the ids of the text views and other views present in our task\_row.xml file.
2. ***onBindViewHolder method:***

* Here we get a reference of position. i.e., at what position (index) what data will be shown.
* We will enter our data through the constructor of *RecyclerTaskAdapter* class.
* Now we will make use of the views of which we found the ids
* We will do like this to set the text of the text views to the text at a given position: holder.task\_title.setText(arrTasks.get(position).title);

At last, we will set the size of the list.

1. ***getItemCount method:***

return arrTasks.size().

1. Now we have to set our custom adapter in the main activity.
2. We have to make an object of the *RecyclerTaskAdapter* class and name it as ‘adapter’ and then automatically its constructor will be called asking for context and ArrayList.
3. Then we will set our adapter to our recycler view like this:

recyclerView.setAdapter(adapter); .