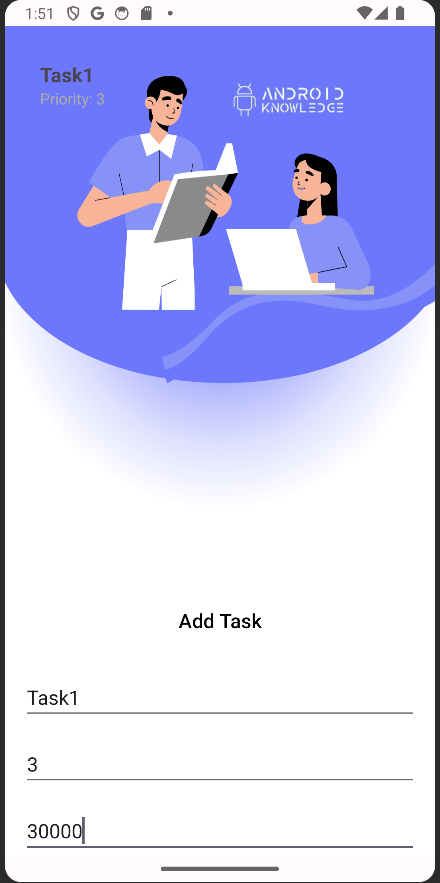
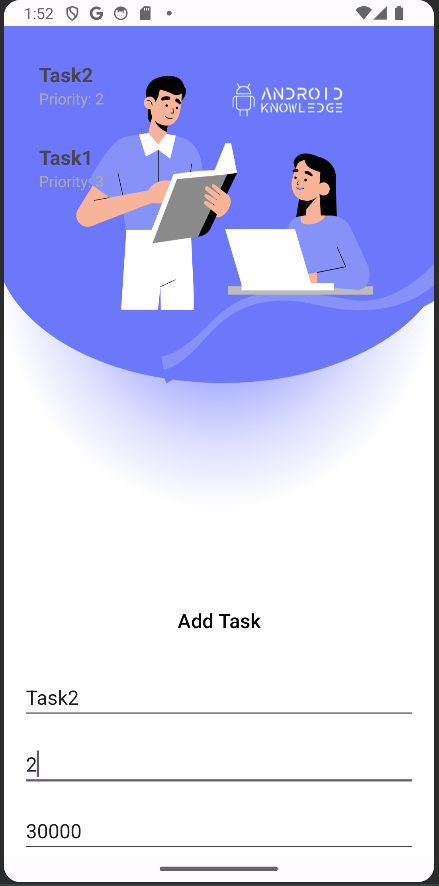
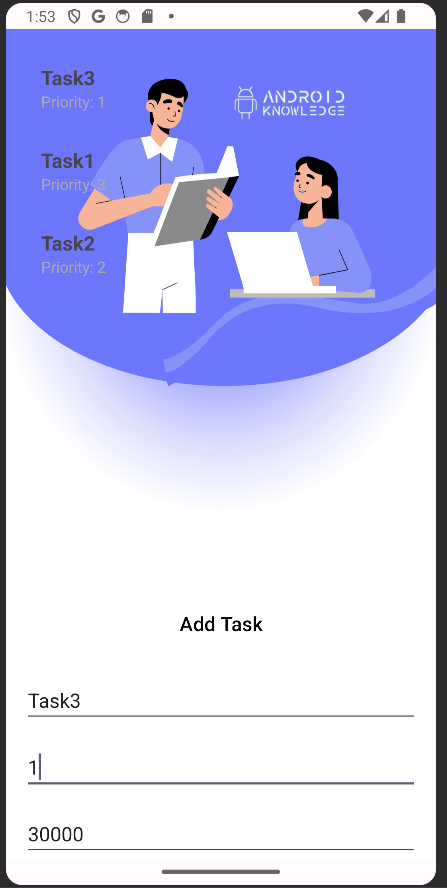
ScreenShots:

Adding Tasks with Priority and time:

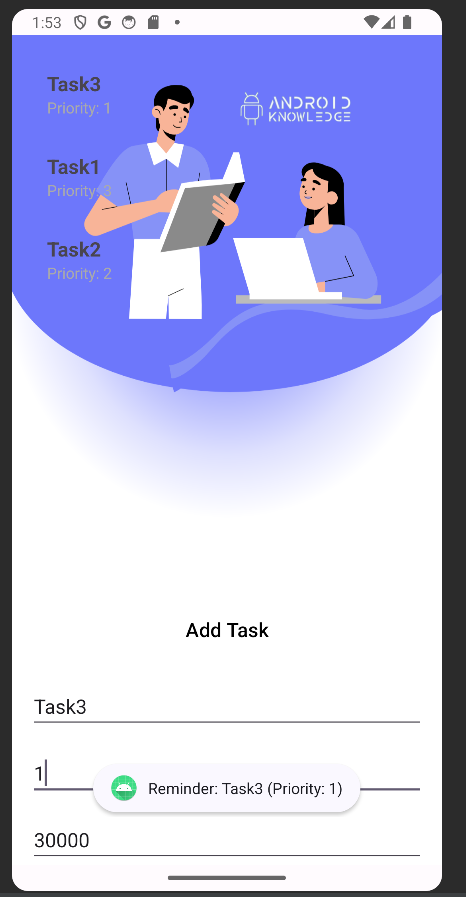




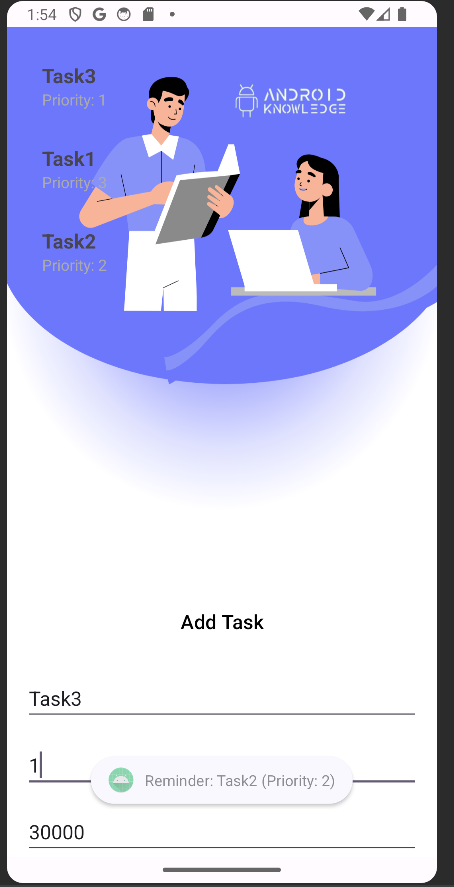


As you can see task are being executed by priority and with time:

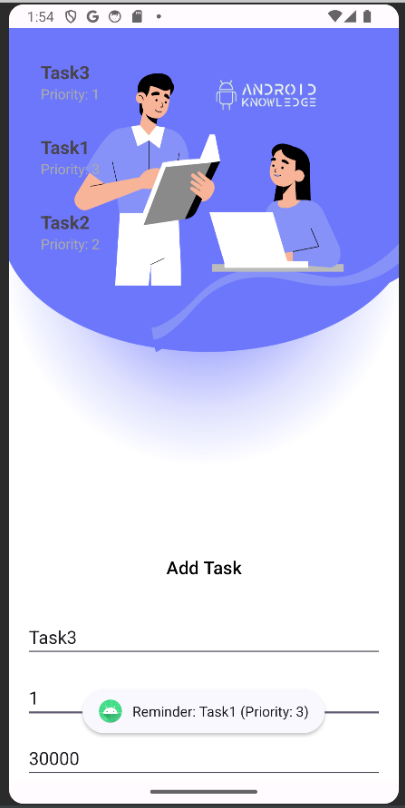
Firstly after 30000ms Task3 will be notified because it had priority 1:



Now after 30000ms task2 will be notified because it had priority 2:



Now in the end after 30000ms after task2 task1 which had priority 3 will be notified:



Code

MainActivity.java:

| package com.android.tools.android\_assignment\_3;  import android.os.Bundle;  import android.os.Handler;  import android.os.Looper;  import android.view.View;  import android.widget.Button;  import android.widget.EditText;  import android.widget.Toast;  import androidx.appcompat.app.AppCompatActivity;  import androidx.recyclerview.widget.LinearLayoutManager;  import androidx.recyclerview.widget.RecyclerView;  import java.util.ArrayList; // Import ArrayList  import java.util.Comparator;  import java.util.PriorityQueue;  public class MainActivity extends AppCompatActivity {  private RecyclerView recyclerView;  private TaskAdapter taskAdapter;  private PriorityQueue<Task> taskQueue;  private Button btnAddTask;  private EditText editTextTaskName, editTextPriority, editTextDelay;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_main);  recyclerView = findViewById(R.id.recyclerViewTasks);  btnAddTask = findViewById(R.id.btnAddTask);  editTextTaskName = findViewById(R.id.editTextTaskName);  editTextPriority = findViewById(R.id.editTextPriority);  editTextDelay = findViewById(R.id.editTextDelay);  taskAdapter = new TaskAdapter();  taskQueue = new PriorityQueue<>(Comparator.comparingInt(Task::getPriority).thenComparingLong(Task::getExecutionTime));  recyclerView.setLayoutManager(new LinearLayoutManager(this));  recyclerView.setAdapter(taskAdapter);  btnAddTask.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  addTask();  }  });  }  private void addTask() {  String taskName = editTextTaskName.getText().toString();  String priorityString = editTextPriority.getText().toString();  String delayString = editTextDelay.getText().toString();  if (!taskName.isEmpty() && !priorityString.isEmpty() && !delayString.isEmpty()) {  int priority = Integer.parseInt(priorityString);  long delay = Long.parseLong(delayString);  Task newTask = new Task(taskName, priority, delay);  taskQueue.add(newTask);  taskAdapter.setTasks(new ArrayList<>(taskQueue));  scheduleTaskNotifications();  } else {  Toast.makeText(this, "Please fill in all fields", Toast.LENGTH\_SHORT).show();  }  }  private void scheduleTaskNotifications() {  new Thread(new Runnable() {  @Override  public void run() {  PriorityQueue<Task> sortedTasks = new PriorityQueue<>(taskQueue);  while (!sortedTasks.isEmpty()) {  Task task = sortedTasks.poll();  try {  Thread.sleep(task.getExecutionTime());  } catch (InterruptedException e) {  e.printStackTrace();  }  new Handler(Looper.getMainLooper()).post(new Runnable() {  @Override  public void run() {  showTaskNotification(task);  }  });  }  }  }).start();  }  private void showTaskNotification(Task task) {  String notificationMessage = "Reminder: " + task.getName() + " (Priority: " + task.getPriority() + ")";  Toast.makeText(MainActivity.this, notificationMessage, Toast.LENGTH\_SHORT).show();  }  } |
| --- |

Task.java

| package com.android.tools.android\_assignment\_3;  public class Task implements Comparable<Task> {  private String name;  private int priority;  private long executionTime; // in milliseconds  public Task(String name, int priority, long executionTime) {  this.name = name;  this.priority = priority;  this.executionTime = executionTime;  }  public String getName() {  return name;  }  public int getPriority() {  return priority;  }  public long getExecutionTime() {  return executionTime;  }  @Override  public int compareTo(Task anotherTask) {  // Compare tasks based on priority, but in descending order (higher priority first)  return Integer.compare(anotherTask.getPriority(), this.priority);  }  } |
| --- |

TaskAdapter:

| package com.android.tools.android\_assignment\_3;  import android.view.LayoutInflater;  import android.view.View;  import android.view.ViewGroup;  import android.widget.TextView;  import androidx.annotation.NonNull;  import androidx.recyclerview.widget.RecyclerView;  import java.util.List;  public class TaskAdapter extends RecyclerView.Adapter<TaskAdapter.TaskViewHolder> {  private List<Task> tasks;  public void setTasks(List<Task> tasks) {  this.tasks = tasks;  notifyDataSetChanged();  }  @NonNull  @Override  public TaskViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {  View view = LayoutInflater.from(parent.getContext()).inflate(R.layout.task\_item, parent, false);  return new TaskViewHolder(view);  }  @Override  public void onBindViewHolder(@NonNull TaskViewHolder holder, int position) {  Task task = tasks.get(position);  holder.bind(task);  }  @Override  public int getItemCount() {  return tasks != null ? tasks.size() : 0;  }  // Inside TaskAdapter.TaskViewHolder  public class TaskViewHolder extends RecyclerView.ViewHolder {  private final TextView taskNameTextView;  private final TextView priorityTextView;  private final TextView executionTimeTextView;  public TaskViewHolder(@NonNull View itemView) {  super(itemView);  taskNameTextView = itemView.findViewById(R.id.textViewTaskName);  priorityTextView = itemView.findViewById(R.id.textViewTaskPriority);  executionTimeTextView = itemView.findViewById(R.id.editTextDelay);  }  public void bind(Task task) {  taskNameTextView.setText(task.getName());  priorityTextView.setText("Priority: " + task.getPriority());  // Check if executionTimeTextView is not null before setting the text  if (executionTimeTextView != null) {  executionTimeTextView.setText("Execution Time: " + task.getExecutionTime() + " ms");  }  }  }  } |
| --- |

Activity\_main.xml

| <?xml version="1.0" encoding="utf-8"?>  <RelativeLayout 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"  android:background="@drawable/img"  tools:context=".MainActivity">  <androidx.recyclerview.widget.RecyclerView  android:id="@+id/recyclerViewTasks"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:padding="16dp"  android:clipToPadding="false"  android:scrollbars="vertical"  android:layout\_above="@+id/layoutInput" />  <RelativeLayout  android:id="@+id/layoutInput"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_alignParentBottom="true">  <Button  android:id="@+id/btnAddTask"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:padding="16dp"  android:text="Add Task"  android:background="?android:attr/selectableItemBackground"  android:textColor="@android:color/black"  android:backgroundTint="@color/colorPrimary"  android:textSize="18sp"  />  <EditText  android:id="@+id/editTextTaskName"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:hint="Task Name"  android:layout\_below="@id/btnAddTask"  android:layout\_marginTop="16dp"  android:layout\_marginStart="16dp"  android:layout\_marginEnd="16dp"  />  <EditText  android:id="@+id/editTextPriority"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:hint="Priority"  android:layout\_below="@id/editTextTaskName"  android:layout\_marginTop="16dp"  android:layout\_marginStart="16dp"  android:layout\_marginEnd="16dp"  />  <EditText  android:id="@+id/editTextDelay"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:hint="Delay (ms)"  android:layout\_below="@id/editTextPriority"  android:layout\_marginTop="16dp"  android:layout\_marginStart="16dp"  android:layout\_marginEnd="16dp"  />  </RelativeLayout>  </RelativeLayout> |
| --- |

Task\_item:

| <?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="vertical"  android:padding="16dp">  <TextView  android:id="@+id/textViewTaskName"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:textSize="18sp"  android:textStyle="bold" />  <TextView  android:id="@+id/textViewTaskPriority"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:textSize="14sp"  android:textColor="@android:color/darker\_gray" />  </LinearLayout> |
| --- |