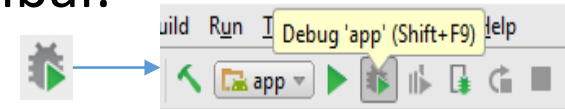


Debugger

Introduction



- Android Studio includes a debugger that enables you to **debug apps** running on the Android Emulator or a connected Android device.
- Android Studio debugger, you can do the following:
 - **Select a device** to debug your app on.
 - Set **breakpoints** in your Java and C/C++ code.
 - Examine variables and evaluate **expressions at runtime**.
 - Capture **screenshots and videos** of your app
- To start debugging, click **Debug** in the toolbar.

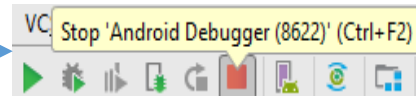
OR



Debugger

Introduction

- If your app is already running on a connected device or emulator, you can start debugging as follows:
 - Click **Attach debugger** to Android process . 
 - In the Choose **Process dialog**, select the process you want to attach the debugger to.
 - Click **OK**.
 - To end a debugging session, click the tab for the session, and then click **Terminate** . 



Debugger

Process Details

The screenshot displays the Android Studio Debugger interface for a process with PID 8617. The top toolbar includes icons for running, stepping through code, and other debugging actions. The main area is divided into three panes:

- Frames:** Shows the call stack. The top frame is `"main"@9,896 in group "main": RUNNING`. The selected frame is `onClick:94, UserAdapter$1 (com.suven.recyclerviewdemo.Adapter)`. Other frames include `performClick:5637, View (android.view)`, `run:22433, View$PerformClick (android.view)`, `handleCallback:751, Handler (android.os)`, `dispatchMessage:95, Handler (android.os)`, `loop:154, Looper (android.os)`, `main:6176, ActivityThread (android.app)`, `invoke:-1, Method (java.lang.reflect)`, and `run:888, ZygoteInit$MethodAndArgsCaller (com.android.internal.os)`.
- Variables:** Shows the current state of variables. The visible variables are:
 - `this = {UserAdapter$1@9897}`
 - `userDetails = {UserDetails@9901}`
 - `view = {LinearLayout@9899} "android.widget.LinearLayout{8`

The bottom status bar shows the current mode is **Debug** (indicated by a green bug icon), along with other tool windows like **Run**, **TODO**, **Terminal**, **Messages**, and **Android Monitor**.

----- Complete debugging features demonstrated in class room @ Level_2 -----