

Lecture 2:

Aakash.R

CB.SC.P2CYS23011

ADB (Android Debug Bridge)

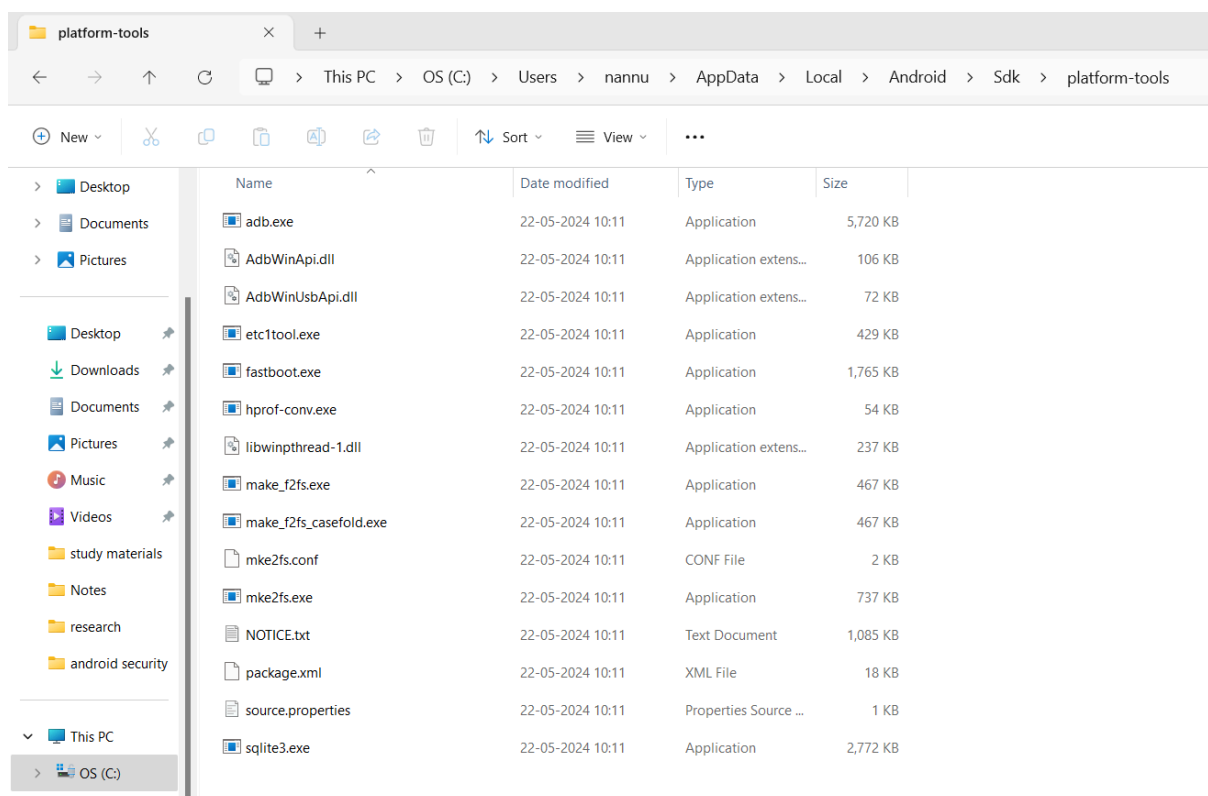
A command-line tool that lets developers communicate with a device (an emulator or a connected Android device) to install applications, access the device's file system, run shell commands, and debug apps.

Consists of three components: a client (which runs on your development machine), a daemon (which runs as a background process on each device), and a server (which manages communication between the client and the daemon).

Commonly used in Android development for tasks like installing APK files, accessing device logs, and performing debugging operations.

ADB is present in the location

C:\Users\AppData\Local\Android\Sdk\platform-tools



Since adb is already installed in the system, enter the following command in the command prompt.

```
C:\Users\nannu>emulator -list-avds
INFO | Storing crashdata in: C:\Users\nannu\AppData\Local\Temp\AndroidEmulator\emu-crash-34.2.14.db, detection is enabled for process: 2720
Pixel_8_Pro_API_34
```

This command will list all the Android Virtual Devices (AVDs) that are available on your system. The primary purpose of this command is to help you identify which AVDs are available for use.

Start the emulator:

```
C:\Users\nannu>emulator -avd Pixel_8_Pro_API_34
INFO | Storing crashdata in: C:\Users\nannu\AppData\Local\Temp\AndroidEmulator\emu-crash-34.2.14.db, detection is enabled for process: 8864
INFO | Android emulator version 34.2.14.0 (build_id 11834374) (CL:N/A)
INFO | Found systemPath C:\Users\nannu\AppData\Local\Android\Sdk\system-images\android-34\google_apis\x86_64\
INFO | Storing crashdata in: C:\Users\nannu\AppData\Local\Temp\AndroidEmulator\emu-crash-34.2.14.db, detection is enabled for process: 10408
INFO | Duplicate loglines will be removed, if you wish to see each individual line launch with the -log-nofilter flag.
INFO | Increasing RAM size to 3072MB
```

Then open a new command prompt and then enter the below command:

```
C:\Users\nannu>adb devices
List of devices attached
emulator-5554    device
```

To start a remote shell session on a connected device or emulator, open a terminal or Command Prompt and type:

```
C:\Users\nannu>adb shell
emu64xa:/ $
```

Once you are in the shell, you can execute many Linux-based commands to interact with the Android operating system.

```

emu64xa:/ $ id
uid=2000(shell) gid=2000(shell) groups=2000(shell),1004(input),1007(log),1011(adb),1015(sdcard_rw),1028(
sdcard_r),1078(ext_data_rw),1079(ext_obb_rw),3001(net_bt_admin),3002(net_bt),3003(inet),3006(net_bw_stat
s),3009(readproc),3011(uhid),3012(readtracefs) context=u:r:shell:s0
emu64xa:/ $ ls
acct          config        etc           mnt           product       system_dlmk
adb_keys      d            init          odm           sdcard        system_ext
apex          data         init.environ.rc odm_dlmk       second_stage_resources vendor
bin           data_mirror  linkerconfig  oem           storage       vendor_dlmk
bugreports    debug_ramdisk lost+found    postinstall   sys
cache         dev         metadata     proc          system
emu64xa:/ $

```

```

C:\Users\nannu>adb devices
List of devices attached
emulator-5554    device
emulator-5556    device

```

To access the shell of a specific emulator **when there is multiple emulators** enter the following command:

```

C:\Users\nannu>adb -s emulator-5556 shell
emu64xa:/ $

```

To view the contents of a specific directory:

```

C:\Users\nannu>adb -s emulator-5556 shell
emu64xa:/ $ cd sdcard/
emu64xa:/sdcard $ ls
Alarms    Audiobooks  Documents  Movies  Notifications  Podcasts  Ringtones
Android  DCIM        Download   Music   Pictures        Recordings
emu64xa:/sdcard $

```

Package manager: Allows you to manage applications and packages on an Android device. This command can perform a wide range of tasks, from installing and uninstalling apps to clearing app data and listing installed packages.

```

emu64xa:/sdcard $ pm
Package manager (package) commands:
  help
    Print this help text.

  path [--user USER_ID] PACKAGE
    Print the path to the .apk of the given PACKAGE.

  dump PACKAGE
    Print various system state associated with the given PACKAGE.

  has-feature FEATURE_NAME [version]
    Prints true and returns exit status 0 when system has a FEATURE_NAME,
    otherwise prints false and returns exit status 1

  list features
    Prints all features of the system.

  list instrumentation [-f] [TARGET-PACKAGE]
    Prints all test packages; optionally only those targeting TARGET-PACKAGE
    Options:
      -f: dump the name of the .apk file containing the test package

  list libraries
    Prints all system libraries.

```

In order to view the packages of a specific application, enter the below command

```

255|emu64xa:/sdcard $ pm list packages|grep youtube
package:com.google.android.apps.youtube.music
package:com.google.android.youtube
emu64xa:/sdcard $

```

To view the packages of the user-installed application:

```

emu64xa:/sdcard $ pm list packages -3

```

Displaying the process id of an application:

```

emu64xa:/sdcard $ ps -A|grep youtube
u0_a136      7251   360   14718760 151128 0          S com.google.android.apps.youtube.music
u0_a147      7254   360   14719196 166756 0          S com.google.android.youtube

```

```

emu64xa:/sdcard $ pm path com.google.android.youtube
package:/product/app/YouTube/YouTube.apk
emu64xa:/sdcard $

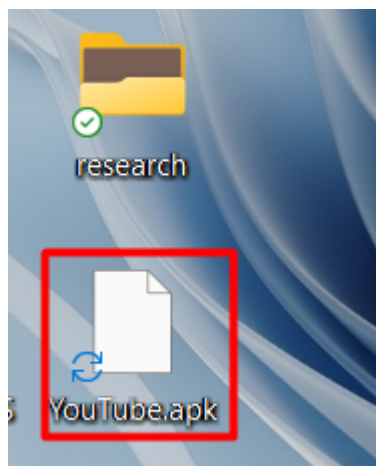
```

After entering the above command, we use adb 'pull' to grab a particular apk.

```

C:\Users\nannu>adb pull /product/app/YouTube/YouTube.apk C:\Users\nannu\OneDrive\Desktop
/product/app/YouTube/YouTube.apk: 1 file pulled, 0 skipped. 91.0 MB/s (127097754 bytes in 1.331s)

```



To copy some files from host system to the emulated device.

I have created a sample text file to transfer.

We use **adb push**

```
C:\Users\nannu>adb push C:\Users\nannu\OneDrive\Desktop sample.txt /sdcard/
C:\Users\nannu\OneDrive\Desktop\: 59 files pushed, 0 skipped. 54.0 MB/s (176151194 bytes in 3.111s)
```

Next to display all the running process in the emulated android device:

```
C:\Users\nannu>adb shell
emu64xa:/ $ ps -A
```

USER	PID	PPID	VSZ	RSS	WCHAN	ADDR	S	NAME
root	1	0	10832468	3208	0	0	S	init
root	2	0	0	0	0	0	S	[kthreadd]
root	3	2	0	0	0	0	I	[rcu_gp]
root	4	2	0	0	0	0	I	[slub_flushwq]
root	5	2	0	0	0	0	I	[netns]
root	9	2	0	0	0	0	I	[mm_percpu_wq]
root	11	2	0	0	0	0	I	[rcu_tasks_kthread]
root	12	2	0	0	0	0	I	[rcu_tasks_trace_kthread]
root	13	2	0	0	0	0	S	[ksoftirqd/0]
root	14	2	0	0	0	0	I	[rcu_preempt]
root	15	2	0	0	0	0	S	[rcub/0]
root	16	2	0	0	0	0	S	[rcu_exp_gp_kthread_worker]
root	17	2	0	0	0	0	S	[rcu_exp_par_gp_kthread_worker]
root	18	2	0	0	0	0	S	[migration/0]
root	19	2	0	0	0	0	I	[kworker/0:1-events]
root	20	2	0	0	0	0	S	[cpuhp/0]

```

emu64xa:/ $ id
uid=2000(shell) gid=2000(shell) groups=2000(shell),1004(input),1007(log),1011(adb),1015(sdcard_rw),1028(
sdcard_r),1078(ext_data_rw),1079(ext_obb_rw),3001(net_bt_admin),3002(net_bt),3003(inet),3006(net_bw_stat
s),3009(readproc),3011(uhid),3012(readtracefs) context=u:r:shell:s0
emu64xa:/ $ su
emu64xa:/ # id
uid=0(root) gid=0(root) groups=0(root),1004(input),1007(log),1011(adb),1015(sdcard_rw),1028(sdcard_r),10
78(ext_data_rw),1079(ext_obb_rw),3001(net_bt_admin),3002(net_bt),3003(inet),3006(net_bw_stats),3009(read
proc),3011(uhid),3012(readtracefs) context=u:r:su:s0
emu64xa:/ #

```

PID of root is 0

```

C:\Users\nannu>adb -s emulator-5556 shell
emu64xa:/ $ id
uid=2000(shell) gid=2000(shell) groups=2000(shell),1004(input),1007(log),1011(adb),1015(sdcard_rw),1028(
sdcard_r),1078(ext_data_rw),1079(ext_obb_rw),3001(net_bt_admin),3002(net_bt),3003(inet),3006(net_bw_stat
s),3009(readproc),3011(uhid),3012(readtracefs) context=u:r:shell:s0
emu64xa:/ $ su
emu64xa:/ # id
uid=0(root) gid=0(root) groups=0(root),1004(input),1007(log),1011(adb),1015(sdcard_rw),1028(sdcard_r),10
78(ext_data_rw),1079(ext_obb_rw),3001(net_bt_admin),3002(net_bt),3003(inet),3006(net_bw_stats),3009(read
proc),3011(uhid),3012(readtracefs) context=u:r:su:s0
emu64xa:/ #

```

To view the UID of each packages.

```

255|emu64xa:/ # pm list packages -U
package:com.android.systemui.auto_generated_rro_vendor__ uid:10078
package:com.google.android.providers.media.module uid:10168
package:com.google.android.overlay.permissioncontroller uid:10049
package:com.google.android.overlay.googlewebview uid:10028
package:com.android.calllogbackup uid:10081
package:com.android.carrierconfig.auto_generated_rro_vendor__ uid:10079
package:com.android.systemui.accessibility.accessibilitymenu uid:10157
package:com.android.internal.emulation.pixel_3_xl uid:10058
package:com.android.providers.contacts uid:10081
package:com.android.internal.emulation.pixel_4a uid:10041
package:com.android.dreams.basic uid:10100
package:com.android.companiondevicemanager uid:10115
package:com.android.cts.priv.ctsshim uid:10179
package:com.google.android.calendar uid:10148
package:com.google.android.networkstack.tethering.emulator uid:10044
package:com.google.android.contacts uid:10152
package:com.android.mms.service uid:1001
package:com.google.android.cellbroadcastreceiver uid:10188
package:com.android.providers.downloads uid:10085
package:com.android.bluetoothmidiservice uid:10104
package:com.android.credentialmanager uid:10096

```

Adb install command:

```
C:\Windows\System32>adb install "C:\Users\nannu\OneDrive\Desktop\APKTOOL\calculator\dist\modified.apk"
Performing Incremental Install
Serving...
All files should be loaded. Notifying the device.
Success
Install command complete in 459 ms
```