**Session 8**

Let’s play with Other TOOLS

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Get info about your device:

adb –e shell getprop ro.product.model *( for real device –d )*

adb –e shell getprop ro.product.device

adb –e shell getprop ro.build.version.release

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Lets install manually an app that is not in a Market Place (from APK folder on your Google Drive ( see instructions and task )

Your file name will be com.united.mobile.apk ( I will be using this apk name as an example)

adb –d install com.united.mobile.apk

if you are unable to download from our directory use this link

 <https://apkpure.com/united-airlines/com.united.mobile.android>

PLEASE NOTE, if you downloaded from the link, your APK name would be different and the command line will look like this

You are welcome to rename the file to whatever you want, as long as you keep an extention .apk )

adb –d install < yourappnamegoeshere.apk>

 ( or to -e –emulator )this name is ONLY valid during an installation....Forget about this . *You can also rename it to anything like baby.apk - doesn’t matter. Because app information is stored in .apk and not in the name itself)*

Install APP if multiple devices attached: you may use -d or -e

In this document some of the devices are separated by serial number or -d, or -e.

adb -s emulator-5554 install com.united.mobile.apk ( or “com.united.mobile.apk” )

adb -s yourdevice#  install com.united.mobile.apk ( or “com.united.mobile.apk” )

*How to get package info:*

adb shell ( or adb –d or adb –e, or adb –s yourdevice# )

adb –d shell ( on my side, because I have two devices – real and emulator )

pm list packages  ( list all packages on your device)

pm list packages -f  ( will give you a full path where all the above packages being located)

pm list packages -f | grep united    ( you will get a package name for United APP )

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

now you have two devices .

adb devices

emulator-5554   device

6703cab3        device ( this is serial# for my device – yours going to be different.

adb -s emulator-5554 shell getprop ro.build.version.release

adb -s 6703cab3 shell getprop ro.build.version.release

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

adb -s emulator-5554 shell

pm list packages -f | grep united    ( you will get a package name for United APP )

E:\AndroidStudio\android-sdk-windows\platform-tools>adb -s emulator-5554 shell

root@generic\_x86:/ # pm list packages -f | grep united

package:/data/app/com.united.mobile.android-1/base.apk=com.united.mobile.android

for real device :

adb -s 6703cab3 shell

shell@d2spr:/ $ pm list packages -f | grep united

/data/app/com.united.mobile.android-1.apk=com.united.mobile.android

 How to make your device REMOTE (real device)?

1. Set your Device and Laptop/PC to the SAME network
2. Check on your Device : Settings→about device→status ( write down )

This is a manual way to figure out what your IP is

But you may use adb commands to do the same.

adb -d shell netcfg

*( mine is 192.168.4.238 – please note your IP addr is different from mine, but I’m using my own numbers. Please adjust your command line to match your IP )*

or

adb -d shell ip addr

1. adb -d usb ( restarting in usb mode)

2. adb -d tcpip 5556 ( assigning port 5556 )

 ( checking if this port is available to use. Receive message "restarting in TCP mode port: 5556")

3. adb -d connect 192.168.4.243:5556  ( "marry” your IP address to the port 5556)

After receiving a message "connected to 192.168.4.243:5556"

Disconnect USB cable from computer and type:  adb devices

Your Device is now listed:

192.168.4.243:5556      device

Lets uninstall our APP from remote device ( you may use any APP )

adb -s 192.168.4.243:5556 uninstall “com.united.mobile.android”

and Install it back:

adb -s 192.168.4.243:5556 install “com.united.mobile.apk”

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

adb -s 192.168.4.243:5556 reboot

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

How to make a device to be real again

adb -s 192.168.4.243:5556 reboot

Connect device back to the PC/Laptop and run

adb devices

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

How to Collect LOGS from Real Device

6703cab3        device

 (this is my Real Device unique serial number. Your number is different)

emulator-5554   device

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

1. From the device itself:  ( to stop madness Ctrl+C)

adb -d logcat   ( it's going to run ALL logs possible )

it’s recommended to clean up buffer before running each set of logs .

adb logcat -c

adb –d logcat \*:E  ( for error logs only )

adb -d logcat \*:F  ( for fatal logs only )

adb -d logcat \*:I  ( for information logs only )

adb -d logcat \*:V  ( for verbose logs only )

adb -d logcat \*:D  ( for debugging logs only )

adb -d logcat \*:W  ( for warning logs only )

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx