

# ADB Commands List – A Detailed ADB Cheat Sheet

 [technastic.com/adb-commands-list-adb-cheat-sheet/](https://technastic.com/adb-commands-list-adb-cheat-sheet/)

By Rakesh

## adb

This is a nude command that can be used to get the ADB version and all possible commands associated with ADB.

```
C:\Windows\System32\cmd.exe
C:\Technastic\platform-tools>adb
Android Debug Bridge version 1.0.41
Version 30.0.4-6686687
Installed as C:\Technastic\platform-tools\adb.exe

global options:
-a          listen on all network interfaces, not just localhost
-d          use USB device (error if multiple devices connected)
-e          use TCP/IP device (error if multiple TCP/IP devices available)
-s SERIAL   use device with given serial (overrides $ANDROID_SERIAL)
-t ID       use device with given transport id
-H          name of adb server host [default=localhost]
-P          port of adb server [default=5037]
-L SOCKET   listen on given socket for adb server [default=tcp:localhost:5037]

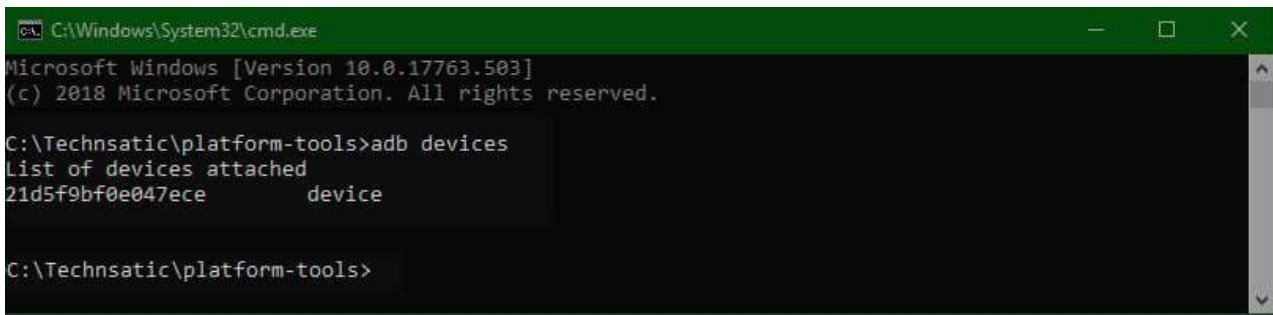
general commands:
devices [-l]          list connected devices (-l for long output)
help                  show this help message
version               show version num

networking:
connect HOST[:PORT]   connect to a device via TCP/IP [default port=5555]
disconnect [HOST[:PORT]]
                      disconnect from given TCP/IP device [default port=5555], or all
pair HOST[:PORT] [PAIRING CODE]
                      pair with a device for secure TCP/IP communication
forward --list         list all forward socket connections
forward [--no-rebind] LOCAL REMOTE
                      forward socket connection using:
                        tcp:<port> (<local> may be "tcp:0" to pick any open port)
                        localabstract:<unix domain socket name>
                        localreserved:<unix domain socket name>
                        localfilesystem:<unix domain socket name>
                        dev:<character device name>
                        jdwp:<process pid> (remote only)
                        acceptfd:<fd> (listen only)
forward --remove LOCAL  remove specific forward socket connection
forward --remove-all   remove all forward socket connections
ppp TTY [PARAMETER...] run PPP over USB
reverse --list          list all reverse socket connections from device
reverse [--no-rebind] REMOTE LOCAL
                      reverse socket connection using:
                        tcp:<port> (<remote> may be "tcp:0" to pick any open port)
                        localabstract:<unix domain socket name>
                        localreserved:<unix domain socket name>
                        localfilesystem:<unix domain socket name>
reverse --remove REMOTE  remove specific reverse socket connection
reverse --remove-all    remove all reverse socket connections from device
mdns check              check if mdns discovery is available
mdns services           list all discovered services

file transfer:
push [--sync] [-z ALGORITHM] [-Z] LOCAL... REMOTE
```

## adb devices

This is one of the most used ADB commands as it is used to see the list of the Android devices connected to your computer. We generally use this command to check if our device is properly connected to the computer or not.

A screenshot of a Windows Command Prompt window. The title bar reads 'C:\Windows\System32\cmd.exe'. The window shows the following text: 'Microsoft Windows [Version 10.0.17763.503] (c) 2018 Microsoft Corporation. All rights reserved. C:\Technsatic\platform-tools>adb devices List of devices attached 21d5f9bf0e047ece device C:\Technsatic\platform-tools>'.

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.17763.503]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Technsatic\platform-tools>adb devices
List of devices attached
21d5f9bf0e047ece    device

C:\Technsatic\platform-tools>
```

`adb devices -l`

You can use the above command to list devices by model or product number.

---

### **adb devices //show devices attached**

This is similar to the 'adb devices' command that shows you the list of connected Android devices and emulators to your computer.

---

### **adb connect ip\_address\_of\_device**

If you want to find out and connect the IP address of your Android device to your PC.

---

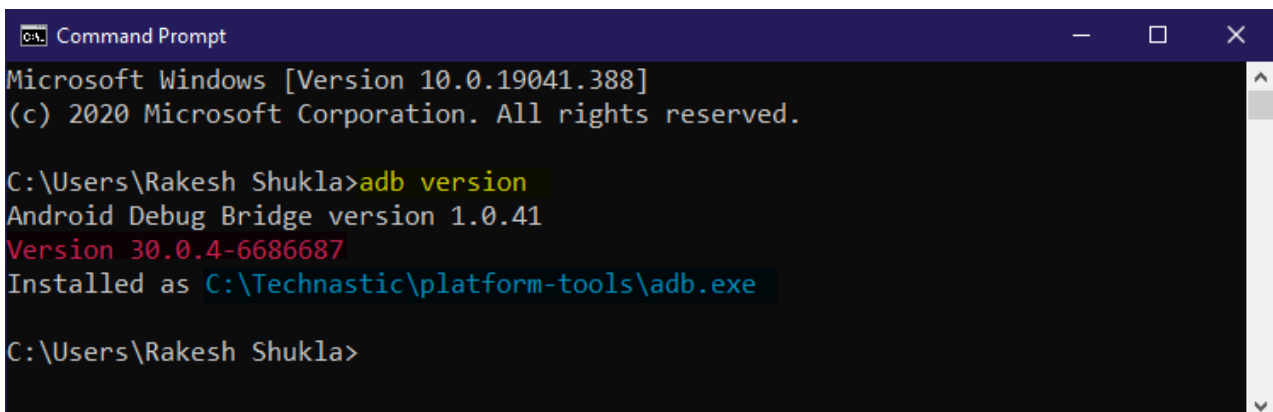
### **adb help**

You can use the 'adb help' command to display the help documentation on all ADB commands.

---

### **adb version**

Use this command to find out the version of the ADB driver installed on your computer.

A screenshot of a Windows Command Prompt window. The title bar reads 'Command Prompt'. The window shows the following text: 'Microsoft Windows [Version 10.0.19041.388] (c) 2020 Microsoft Corporation. All rights reserved. C:\Users\Rakesh Shukla>adb version Android Debug Bridge version 1.0.41 Version 30.0.4-6686687 Installed as C:\Technastic\platform-tools\adb.exe C:\Users\Rakesh Shukla>'.

```
Command Prompt
Microsoft Windows [Version 10.0.19041.388]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Rakesh Shukla>adb version
Android Debug Bridge version 1.0.41
Version 30.0.4-6686687
Installed as C:\Technastic\platform-tools\adb.exe

C:\Users\Rakesh Shukla>
```

---

### **adb reboot**

As this very ADB command suggests, 'adb reboot' command can help reboot your Android phone or tablet into the bootloader, fastboot, or recovery mode in case the hardware keys of your device do not function properly.

## adb reboot bootloader

---

You can use this command to reboot your Android device into the Fastboot or Bootloader Mode. The command is often used when you want to flash the factory images or a custom recovery.

## adb reboot recovery

---

This is one of the most used ADB commands. You can execute it to reboot your device into Android Recovery mode.

## adb install

---

Usually, we transfer an APK file to our Android device and install it via File Manager. Thanks to the 'adb install' command that it makes it easy to install APK files on Android devices directly from your computer. To be able to do so, copy the APK file to the SDK platform-tools folder first. There are some variations of this command for different conditions.

```
adb install com.facebook.katana.apk
```

While you can install an APK on your Android devices using the above command, you can use the command given below to update or reinstall an app without deleting its data.

```
adb install -r com.facebook.katana.apk
```

Some apps support installation on the SD card. By using the following ADB commands, move an app to the SD storage.

```
adb install -s com.facebook.katana.apk
```

```
adb install -k <add the path of the .APK file on your computer>
```

## adb uninstall

---

Using this command in the ADB terminal windows, you can uninstall an app from your phone or tablet.

```
adb uninstall com.facebook.katana
```

If you want to uninstall an app but keep its data and cache files, you can use the 'adb uninstall' command with '-k' parameter as shown below.

```
adb uninstall -k com.facebook.katana
```

## adb usb

---

If you want to find out the Android device or emulators currently connected to your Windows, Mac, or Linux computer, you can use the 'adb usb' command.

## adb logcat

---

By executing the 'adb logcat' command, you can see the log data of your Android device on your computer. There are a few variations of this command with '-c' and '-d' parameters.

```
adb logcat -c // clear //
```

While using the above command you can clear all existing logs on your Android phone or tablet, you can save the logcat data on your PC with the following command.

```
adb logcat -d > [path_to_file] //
```

---

### adb start-server

This is a useful command that lets you start the adb server in case it stopped responding. It's often used after killing the adb server as described below.

---

### adb kill-server

If the ADB terminal is not functioning properly, you can try killing the ADB server. It's similar to turning off our Android devices to fix small issues.

---

### adb sideload

It's one of the most popular ADB commands as it can be used to sideload software **update.zip** files using your computer. If you have downloaded a flashable update.zip, just copy it to the '**platform-tools**' folder and execute the following command. If the zip file you download has a different name, rename it to "update.zip" for the sake of convenience.

```
adb sideload update.zip
```

Alternatively, you can sideload or flash update zip packages using Android stock recovery or TWRP recovery as well. For that, you will have to transfer the update.zip file to your device, reboot your Android into recovery mode, and select the "**Apply update from sdcard**" option.

---

### adb pull

You can download or pull files stored on your Android device to your computer using the following ADB command.

This command can be used to pull any files from your device and save them to the 'platform-tools' folder on your computer.

```
adb pull /sdcard/video-01.mp4
```

In case you want to pull the file to a specific location or drive (D drive, for instance) on your PC, you issue the following command mentioning the path of the storage location as shown below.

```
adb pull /sdcard/video-01.mp4 d:\
```

## **adb push**

---

Similarly, this command can be used to push a file from your computer to your device. Please note that you'll have to transfer the file you want to push to the 'platform-tools' directory first. To push a file to your Android device's SD card, for example, use the following code.

```
adb push com.whatsapp_2.19.368-453132.apk /sdcard
```

In case you want to send a file to your device stored in a specific location, you can use the below command instead.

```
adb push d:\com.whatsapp_2.19.368-453132.apk /sdcard
```

## **adb backup //**

---

To create or take a full backup of your Android to your computer, try this command.

## **adb restore //**

---

You can also restore the backup you have already created using this command.

## **adb bugreport**

---

This command is best suited if you want to diagnose any issue on Android devices. Its execution can show you log data, dumpstate, and dumsys from your Android device on your computer.

## **Don't Miss: [How to Unlock Android PIN and Pattern Using ADB](#)**

## **adb jdwp**

---

JDWP means Java Debug Wire Protocol. By using this ADB command, you can see the list of JDWP processes on your PC.

## **adb get-serialno**

---

If you want to find out ADB instance serial number, you can use this command.

## **adb get-state**

---

Shows the ADB status of a connected device or emulator.

## **adb wait-for-device**

---

This program tells ADB that it has to wait and keep the connection on hold until the next command is issued.