

Android Debug Bridge or better known as ADB is a powerful and versatile tool that lets you do a lot of things like pulling out logs, installing and uninstalling apps, transferring files, rooting and flashing custom ROMs, creating device backups, etc. In fact, most of the advanced Android tutorials and how-to guides tend to use adb commands to get things done. Moreover, adb is also very useful when your Android device is not functioning as it should or when things get very messy and unusable. Though the adb command shell looks intimidating and complex, here is a list of adb commands to get you started and do some useful things in the process.

## Install ADB on Windows

Unlike in previous versions, you don't have to install complete Android SDK to install ADB. Simply [download the standalone ADB zip file](#), extract it to the root of C drive and you are done. To access adb, open command prompt by searching for it in the start menu and navigate to the adb folder using the below command. If you've installed adb in a different folder then change the command accordingly.

```
cd c:\adb
```

**Tip:** Then open up a Command Prompt from the same directory, hold down your Shift key and Right-clicking within the folder then click the “*open command prompt here*” option

Now, connect your Android device via USB and you can proceed to test the below commands. For further reference, you can check [this guide](#) on how to properly install and test adb on Windows.

## List of ADB Commands

### 1. Start or Stop ADB Server

Obviously, the first command you should know is how to start and stop adb server. This enables you to interact with your connected Android device. To start the adb server, use the below command.

```
adb start-server
```

Once you are done with your work, you can use the command below to stop the adb server.

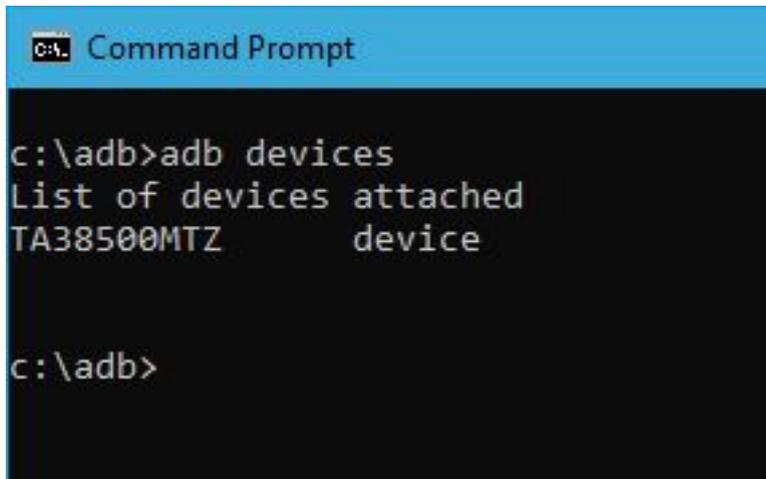
```
adb kill-server
```

### 2. List Connected Android Devices

This is one of the most famous commands. When you connect your device to the computer via USB, use this command to verify if adb can find the connected device.

```
adb devices
```

If your device is properly connected to your system, the above command will start the daemon service, scans the system and lists all the connected Android drives. The good thing about this command is that it lists both the state of the device and its serial number.



```
C:\> Command Prompt

c:\adb>adb devices
List of devices attached
TA38500MTZ      device

c:\adb>
```

### 3. Know Status of the Device

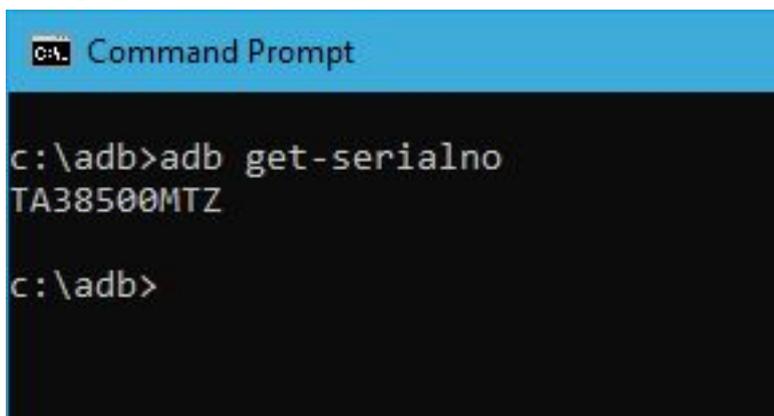
As you can tell from the name itself, this command can be used to know the device state. When the command is executed, it shows whether your device state is in offline, bootloader or in device mode. For a normal Android device, you will see your Android state as "device", just like in the below image.

```
adb get-state
```

### 4. Get Device Serial Number

This command lets you know the device serial number of the connected device. On your phone or tablet, you can see the device serial number by navigating to "Settings > About Phone > Status".

```
adb get-serialno
```



```
C:\> Command Prompt

c:\adb>adb get-serialno
TA38500MTZ

c:\adb>
```

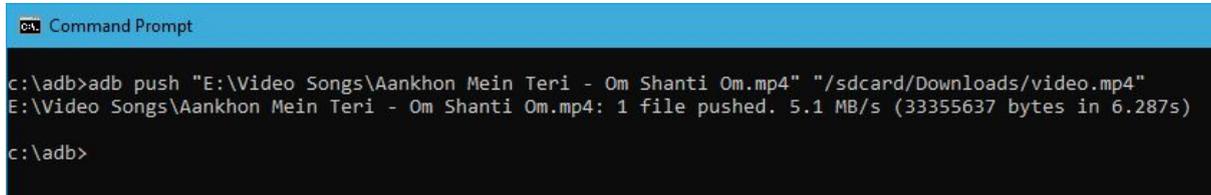
## 5. Copy Files from Computer to Phone

If you want to copy files from your computer to phone using adb then you can use this command. Do forget to replace [source] and [destination] with actual file paths.

```
adb push [source] [destination]
```

Once you replace the above command with actual file paths, this is how it looks like.

```
adb push "E:\Video Songs\Aankhon Mein Teri - Om Shanti Om.mp4"  
"/sdcard/Downloads/video.mp4"
```



```
Command Prompt  
c:\adb>adb push "E:\Video Songs\Aankhon Mein Teri - Om Shanti Om.mp4" "/sdcard/Downloads/video.mp4"  
E:\Video Songs\Aankhon Mein Teri - Om Shanti Om.mp4: 1 file pushed. 5.1 MB/s (33355637 bytes in 6.287s)  
c:\adb>
```

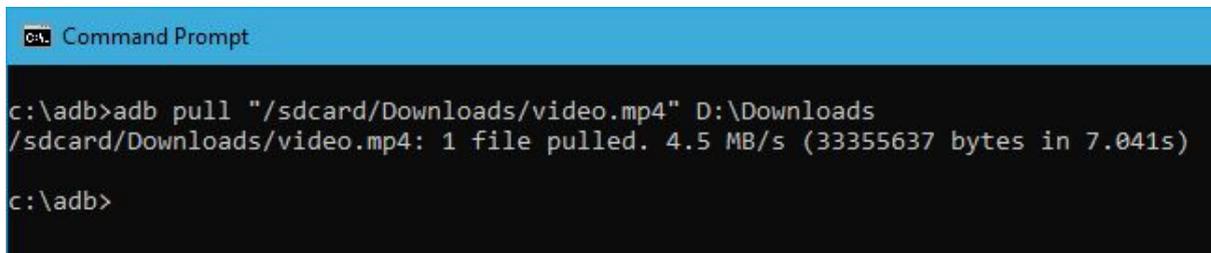
## 6. Copy Files from Phone to Computer

Just like you can copy files from your computer to Android device, you can copy files from your phone to computer. To do that simply use the below command. Replace [source] and [destination] with actual file paths.

```
adb pull [source] [destination]
```

Once you replace the above command with actual file paths, this is how it looks like.

```
adb pull "/sdcard/Downloads/video.mp4" D:\Downloads
```



```
Command Prompt  
c:\adb>adb pull "/sdcard/Downloads/video.mp4" D:\Downloads  
/sdcard/Downloads/video.mp4: 1 file pulled. 4.5 MB/s (33355637 bytes in 7.041s)  
c:\adb>
```

## 7. Install/Uninstall Apps

Besides from moving files back and forth, you can actually install apk files with just a single command. To install an app you have to specify the full path of the apk file. So, replace "path/to/file.apk" with the actual apk file path.

```
adb install "path/to/file.apk"
```

If you have multiple devices attached to your computer and only want to install the apk file on just one device then use the below command. Replace [serial-number] with the actual device serial number. You can get the device serial number using the fourth command above.

```
adb -s [serial-number] install "path/to/file.apk"
```

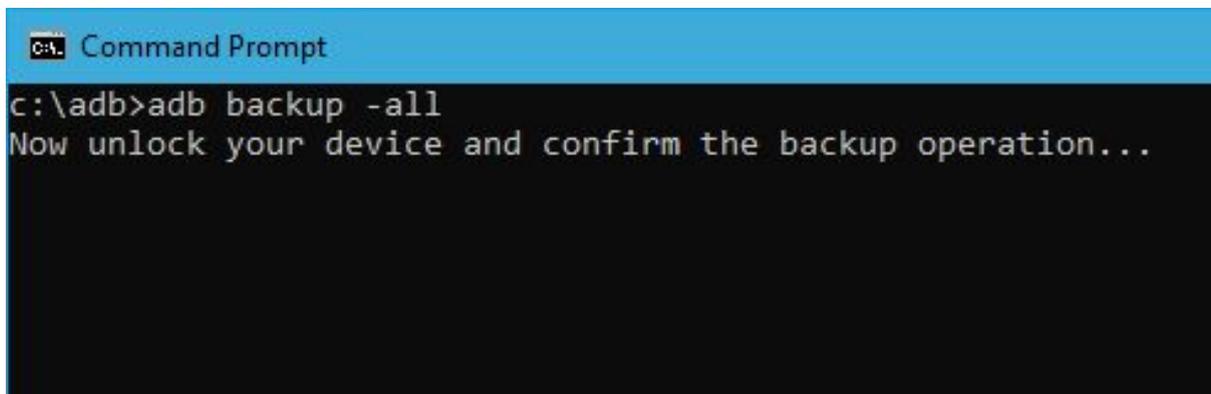
To uninstall an app, simply execute the below command. Replace <package-name> with the actual fully qualified package name of the app.

```
adb uninstall <package-name>
```

## 8. Backup Android Device

To backup all the device and app data you can use the below command. When executed, it will trigger the backup, asks you to accept the action on your Android device and then creates "backup.adb" file in the current directory.

```
adb backup -all
```

A screenshot of a Windows Command Prompt window. The title bar is blue and contains the text "Command Prompt". The main area is black with white text. The prompt shows the command "c:\adb>adb backup -all" being entered. Below the command, the text "Now unlock your device and confirm the backup operation..." is displayed, indicating the next step in the process.

```
Command Prompt
c:\adb>adb backup -all
Now unlock your device and confirm the backup operation...
```

## 9. Restore Android Device

To restore a backup, use the below command. Don't forget to replace "path/to/backup.adb" with the actual file path.

```
adb restore "path/to/backup.adb"
```

## 10. Reboot Android Device into Recovery Mode

The recovery mode helps you repair or recovery the Android device using the tools built into it. Generally, you can boot into recovery mode using the volume and power button combination. Alternatively, you can also connect your device to the system and use the below command to boot into recovery mode.

```
adb reboot-recovery
```

## 11. Reboot Android Device into Bootloader Mode

The below command lets you boot into bootloader mode. Generally, the bootloader mode is very similar to the fastboot mode.

```
adb reboot-bootloader
```

## 12. Reboot Android Device into Fastboot Mode

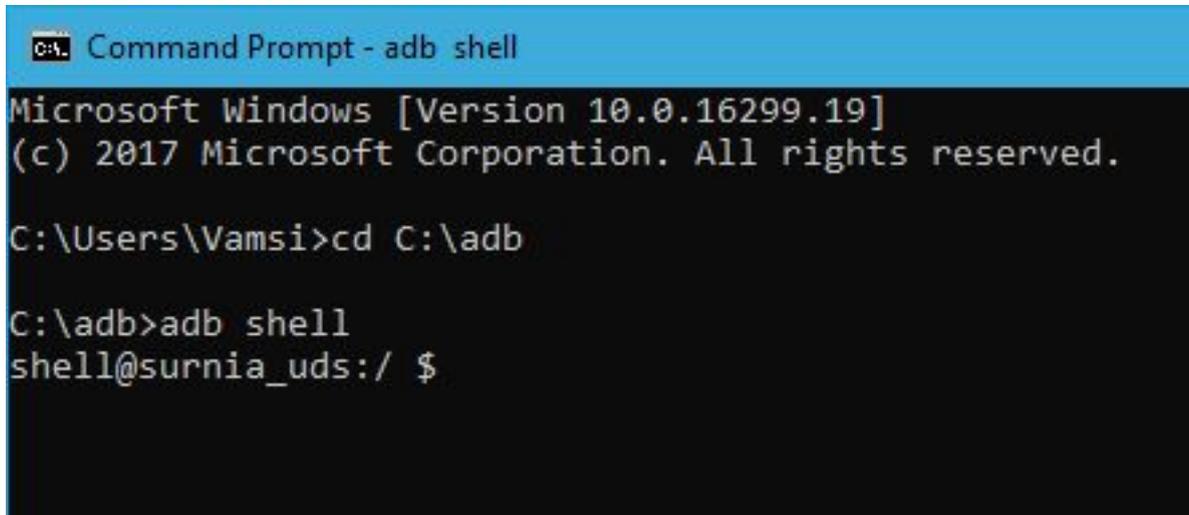
The fastboot mode is generally used to flash custom ROMs, bootloader, and even kernels. Use the below command to boot into fastboot mode.

```
adb fastboot
```

### 13. Start Remote Shell

This command starts the remote shell and lets you control and configure your device using the shell commands.

```
adb shell
```

A screenshot of a Windows Command Prompt window. The title bar reads "Command Prompt - adb shell". The window content shows the following text: "Microsoft Windows [Version 10.0.16299.19] (c) 2017 Microsoft Corporation. All rights reserved. C:\Users\Vamsi>cd C:\adb C:\adb>adb shell shell@surnia\_uds:/ \$".

```
Command Prompt - adb shell
Microsoft Windows [Version 10.0.16299.19]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Vamsi>cd C:\adb

C:\adb>adb shell
shell@surnia_uds:/ $
```

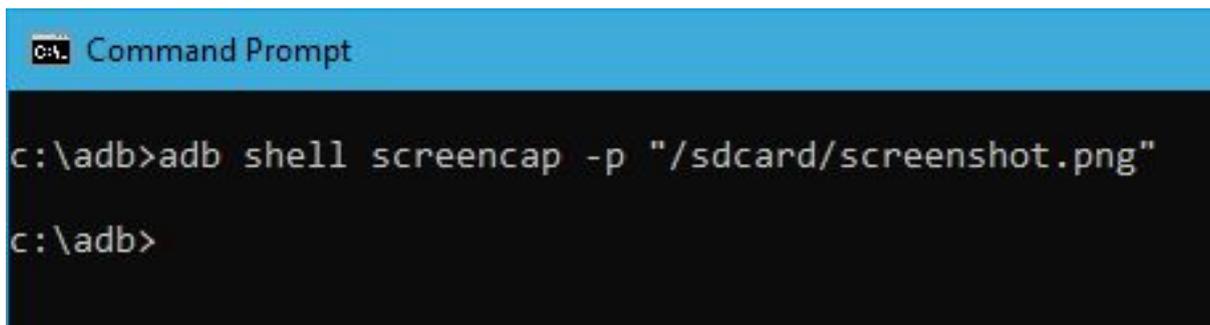
### 14. Take Screenshots

It is nothing hard to take a screenshot on and Android. All you have to do is press the Power button and Volume Down button at the same time. Alternatively, you can also use this command to take a quick screenshot. Replace "/path/to/screenshot.png" with the actual destination path. If you want to, you can customize the file name by changing "screenshot" with whatever name you want.

```
adb shell screencap -p "/path/to/screenshot.png"
```

Once you replace the destination path, this is how the command looks like.

```
adb shell screencap -p "/sdcard/screenshot.png"
```

A screenshot of a Windows Command Prompt window. The title bar reads "Command Prompt". The window content shows the following text: "c:\adb>adb shell screencap -p "/sdcard/screenshot.png" c:\adb>".

```
Command Prompt

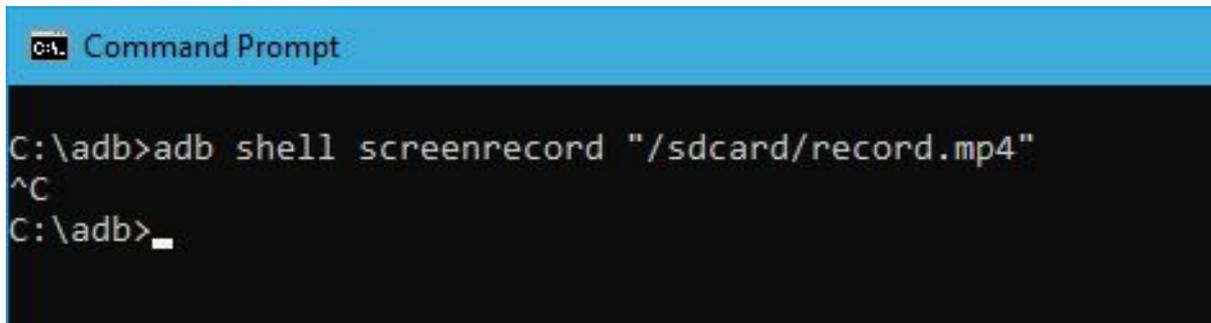
c:\adb>adb shell screencap -p "/sdcard/screenshot.png"

c:\adb>
```

## 15. Record Android Screen

Apart from screenshots, you can record the Android device screen using the below command. Again, replace "/path/to/record.mp4" with the actual destination path. Of course, you can customize the file name by changing "record" with whatever name you want.

```
adb shell screenrecord "/path/to/record.mp4"
```

A screenshot of a Windows Command Prompt window. The title bar is blue and contains the text "C:\ Command Prompt". The main area is black with white text. The text shows the command "C:\adb>adb shell screenrecord "/sdcard/record.mp4"" being entered. Below the command, there is a carriage return symbol (^C) and the prompt "C:\adb>\_" with a cursor.

That's all for now. Comment below sharing your thoughts and experiences about using the above adb commands on your Android device.

Link: <https://techwiser.com/list-adb-commands/>