

# Dual Booting

Mohan Raman

July 14, 2025

## Contents

<b>1</b>	<b>Dual Booting</b>	<b>1</b>
1.1	BitLocker . . . . .	1
1.2	UEFI / BIOS . . . . .	1
1.3	Create Partition for Linux in Windows . . . . .	3
1.4	Selecting Root Mountpoint during Linux Installation . . . . .	5
1.5	UEFI - Restore Linux from Windows . . . . .	8

## 1 Dual Booting

To make windows and linux co-exist on the same machine

### 1.1 BitLocker

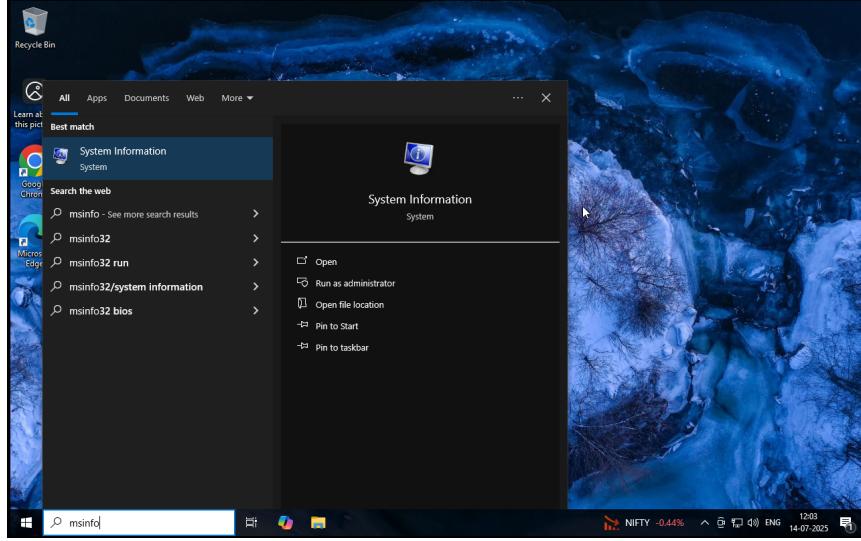
This is an encryption method used to encrypt entire partition in Windows. If this is enabled we cannot dual boot. Watch this Video <https://www.youtube.com/watch?v=oN0mXLkXfqM> to check if BitLocker is turned on for a particular partition. To turn off BitLocker, follow this Video <https://www.youtube.com/watch?v=AU5nm7b1aT0>

### 1.2 UEFI / BIOS

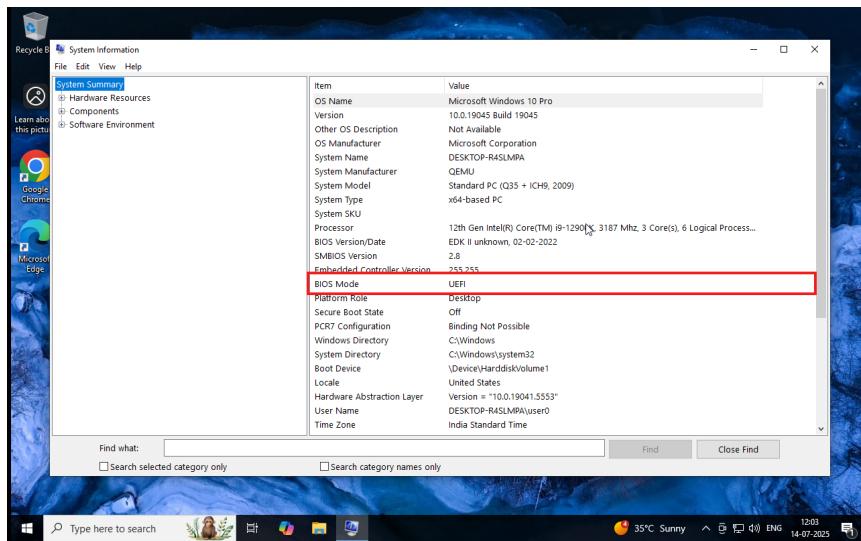
**UEFI** Modern boot firmware

**BIOS** Old boot firmware

To Check which boot firmware your system use in windows, click **Start Menu**, then type **msinfo**, then click **System Information**



Check the Boot Mode field whether windows booted in BIOS mode or UEFI mode



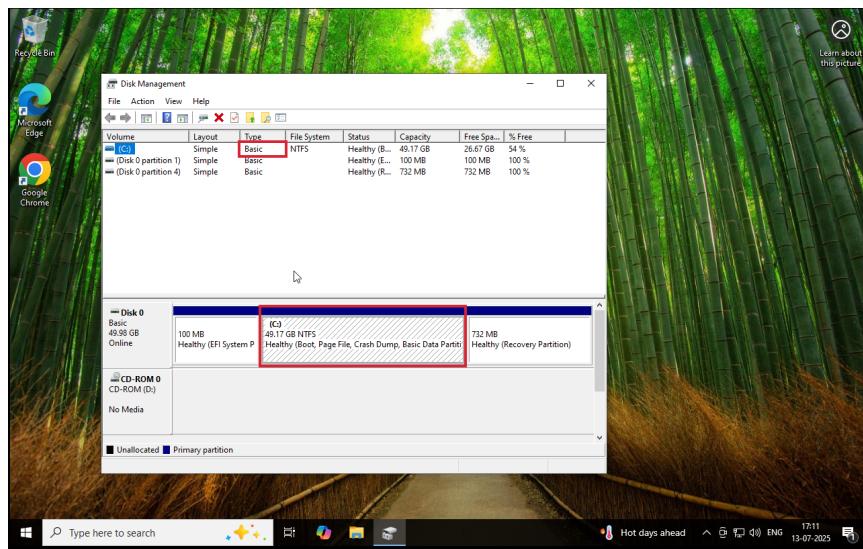
If the Windows booted in UEFI mode, then Linux Live Booting should also boot in the same UEFI mode. You can check which mode Live Booting is in through the following command

```
findmnt /sys/firmware/efi/efivars &>/dev/null && echo "UEFI mode" || echo "BIOS mode"
```

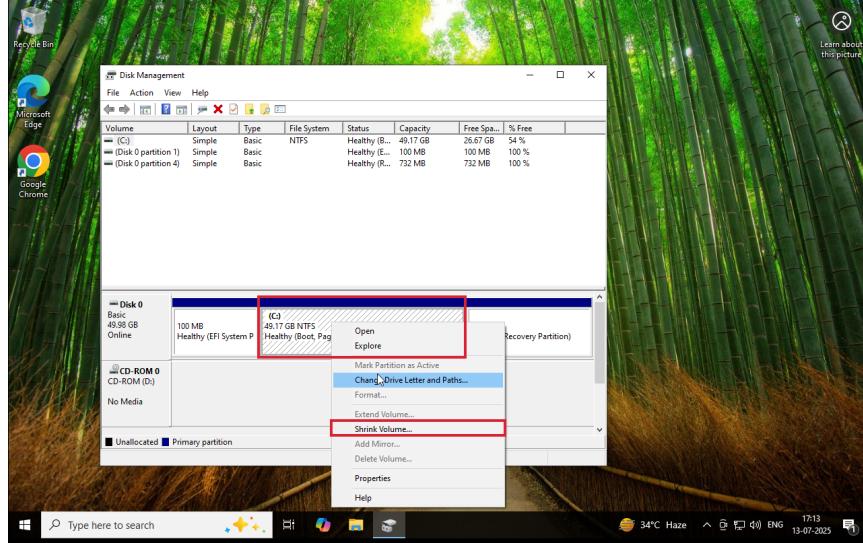
You can switch off BIOS mode in your system's BIOS Settings, look for something like *Legacy Mode* or *Legacy Boot* or *BIOS Mode* and disable it so that Live Boot will be booted only through UEFI.

### 1.3 Create Partition for Linux in Windows

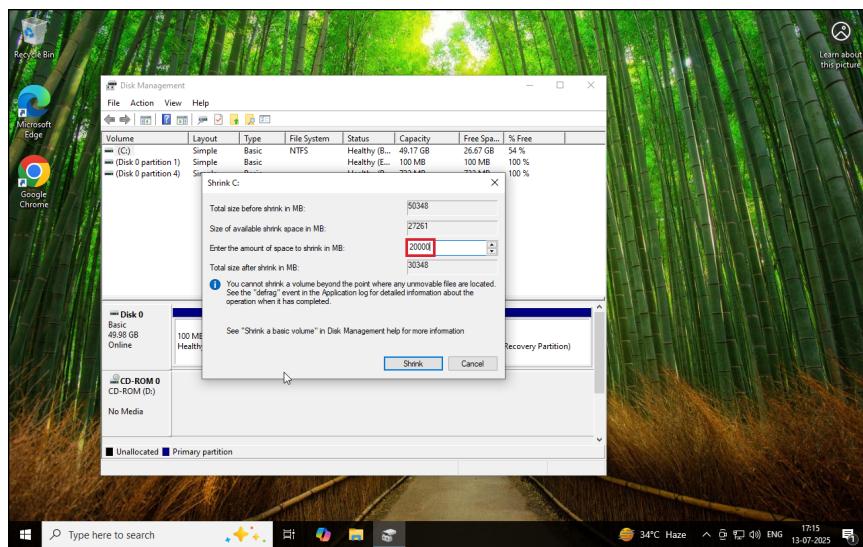
- Click **Start Menu**, then type **Disk Management**, Then start the **Disk Management Application**
- In Disk Management, check if the existing Partitions type is **Basic**, If it says **Virtual** then we cannot create partition for Linux. First we need to remove those virtual partitions or transform those virtual partitions into **Basic** partitions. Scope of this document cannot cover this topic, take AI's help.



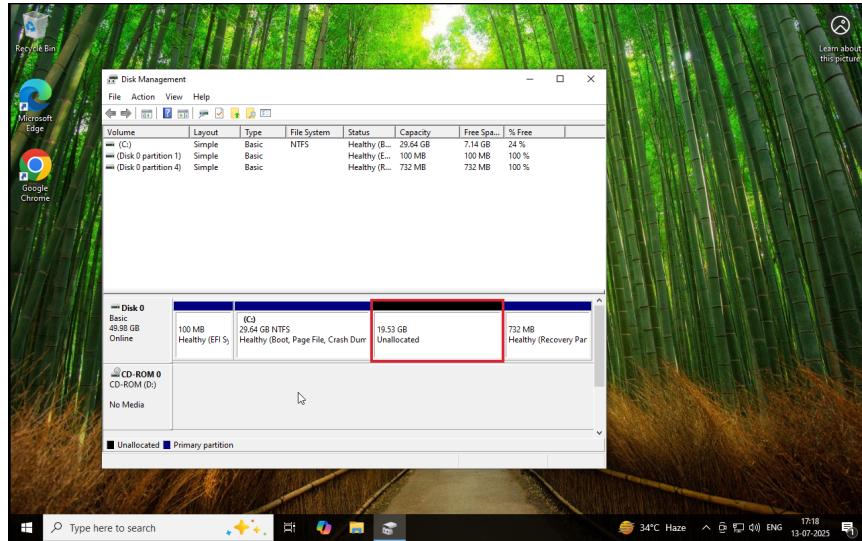
- Once we ensure our targetted partition's type is **Basic**, Right Click on that partition and click **Shrink**



- Windows will calculate how much it can shrink, and show us that information and ask us how much we actually need to shrink. We need to choose how much disk space we want to allocate for the Linux. Linux will work fine with 20GB for on and off use. But if the user is planning for daily drive linux, make sure it is atleast 50GB to 80GB. Once you decide, put that Size in MB on "Enter the amount of space to shrink in MB" box.

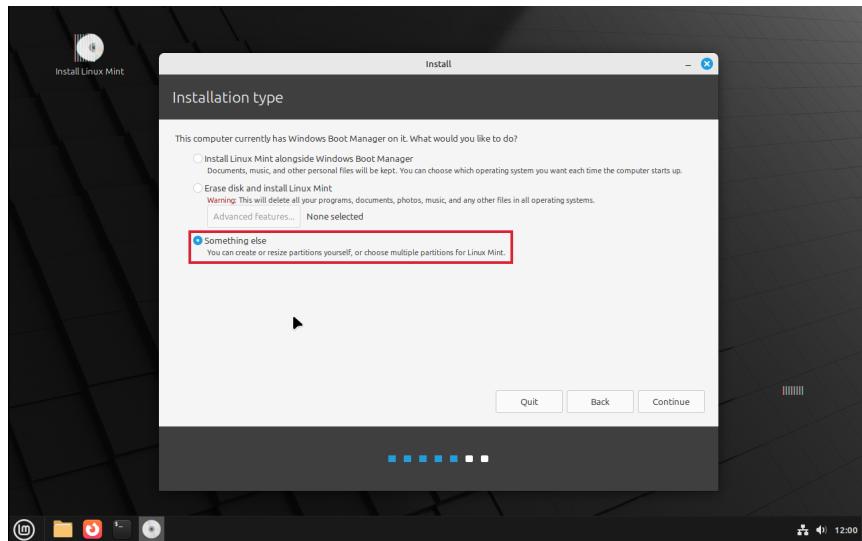


- Windows will do the job and finally we will see **Unallocated** free space for the Size we requested.

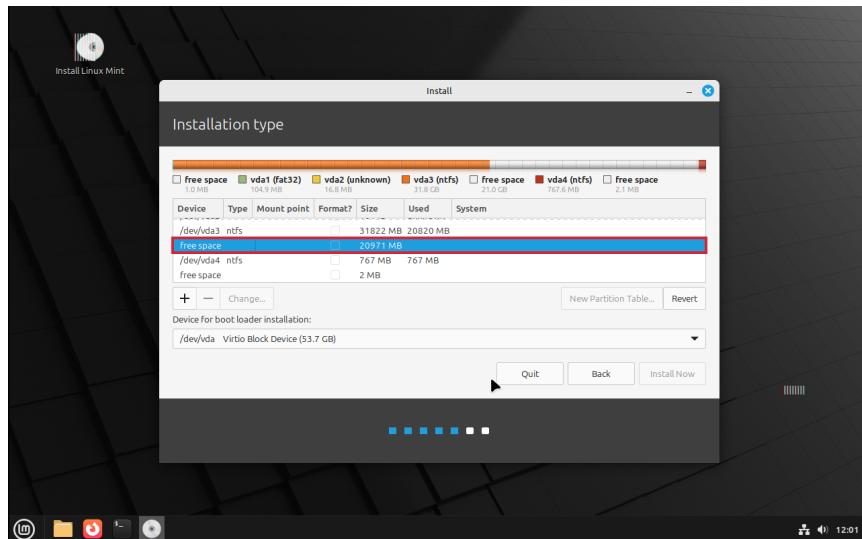


## 1.4 Selecting Root Mountpoint during Linux Installation

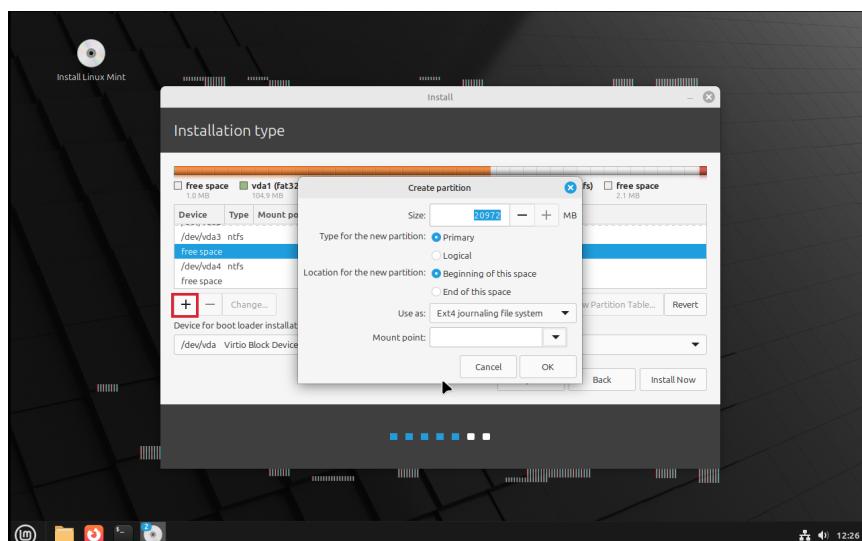
- Boot Live Linux and start the Installation Process
- During Installation Process, when the Installation Wizard asks for **Installation Type**, choose **Something Else**



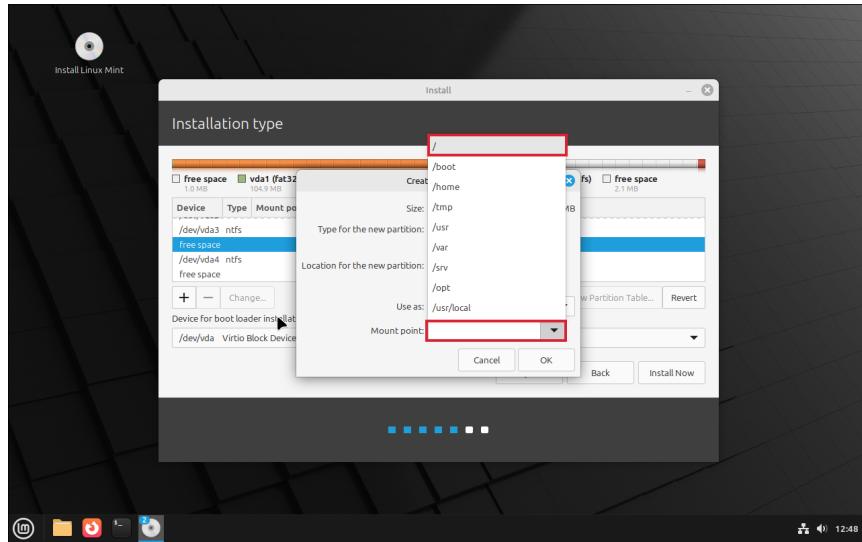
- In the Next screen, Identify and select the **free space** we created in Windows. It may not match the exact size shown in Windows, but it will be very near to that size.



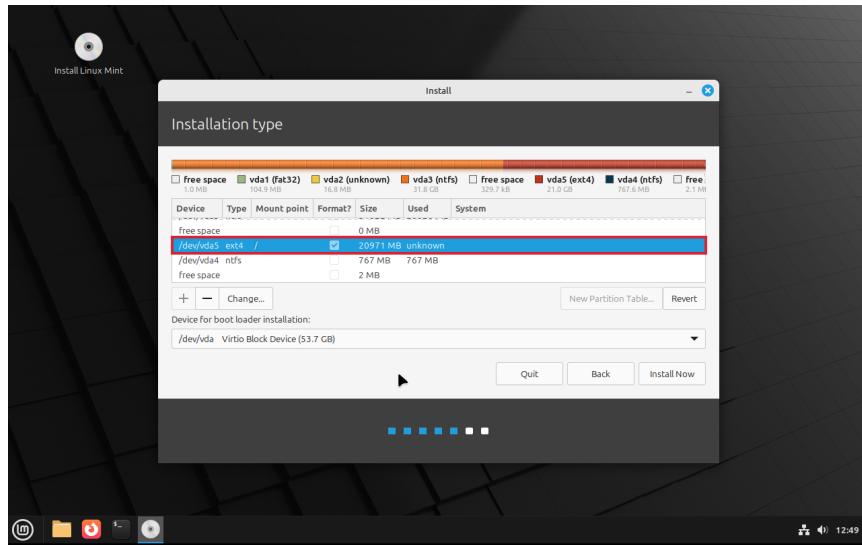
- Then click on the + Icon, to open the Create Partition dialog box



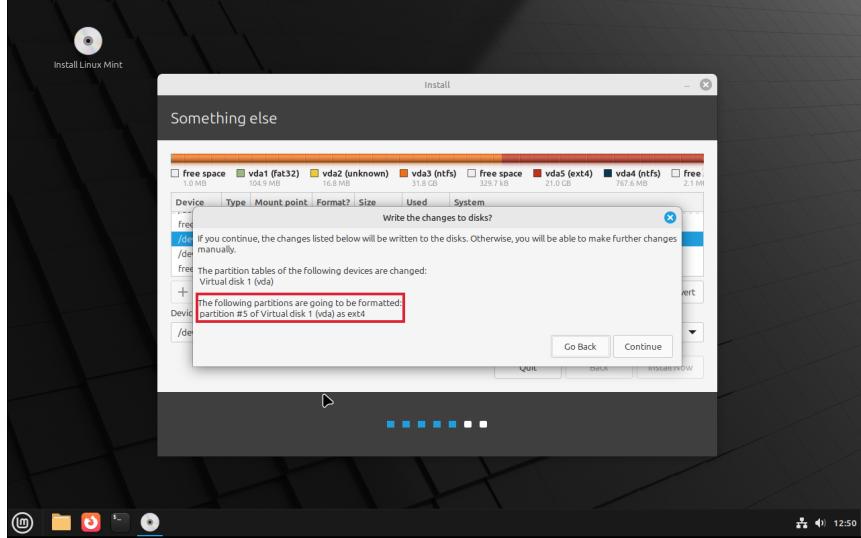
- Click on the **Mount point:** box, select / to associate the newly created partition into the Root directory of Linux



- Make sure the partition association is correct and the file system type is **Ext4** (Linux default filesystem)



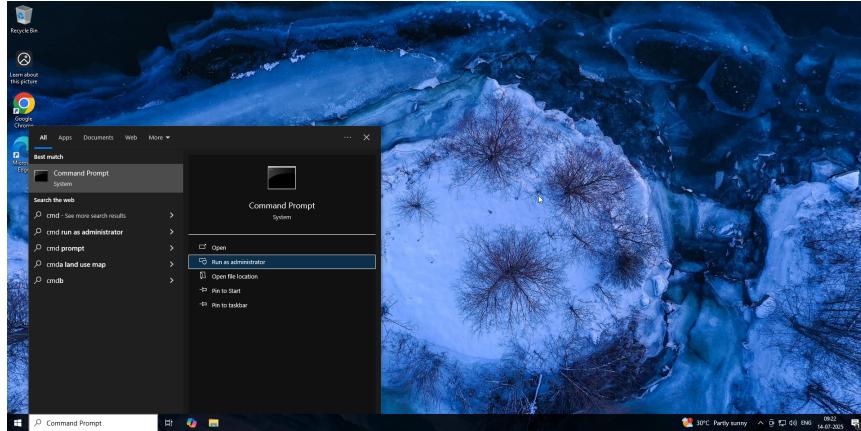
- Make sure we only formatate the partition we newly created, otherwise **data will be lost**.



- Proceed further and complete the Linux Installation.

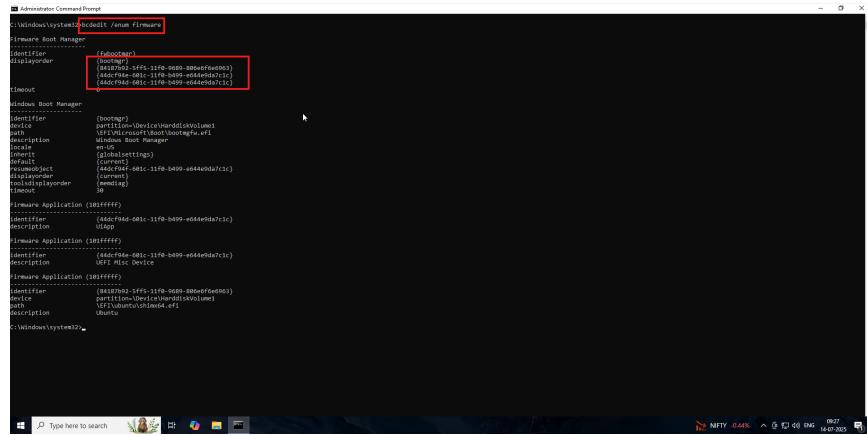
## 1.5 UEFI - Restore Linux from Windows

- Sometimes booting between Windows and Linux, Windows updates may rewrite the boot priority to always directly boot into windows, so we loose option to boot into Linux. To restore, Click Start menu, type cmd and start **Command Prompt** by choosing **Run As Administrator**.



- Run the following command to list the UEFI firmware bootloaders currently available, in the output, you can see the **displayorder**, we need to set **Linux** entry as first in the **displayorder**.

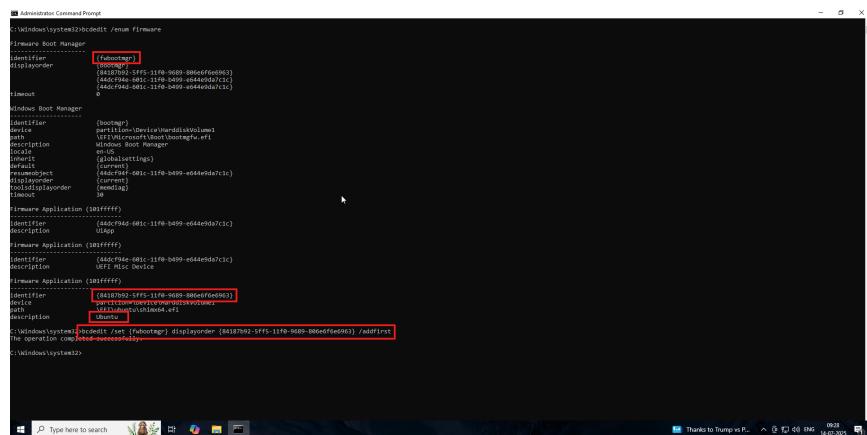
```
bcdedit /enum firmware
```



```
C:\Windows\system32>bcdedit /enum firmware
Windows Boot Manager
  identifier {fb000000-0000-0000-0000-000000000000}
  displayorder 0
  timeout 30
  windows Boot Manager
    identifier {bootmgr}
    displayorder 1
    timeout 30
    device partition:\\Device\\HarddiskVolume1
    description Windows Boot Manager
    locale en-US
    inherit {globalsettings}
    osdevice {current}
    resumeobject {current}
    displayorder {current}
    timeout 30
  firmware Application (0x0fffff)
  identifier {00000000-0000-0000-0000-000000000000}
  description Uloop
  firmware Application (0x0fffff)
  identifier {00000000-0000-0000-0000-000000000000}
  description UEFI Misc Device
  firmware Application (0x0fffff)
  identifier {00000000-0000-0000-0000-000000000000}
  device partition:\\Device\\HarddiskVolume1
  path \Device\Hdreds\efi
  description Ubuntu
C:\Windows\system32>
```

- Copy the Linux's bootloader's **identifier** and run the following command replacing <linux-bootloader-identifier> with actual identifier

```
bcdedit /set {fwbootmgr} displayorder <linux-bootloader-identifier> /addfirst
```



```
C:\Windows\system32>bcdedit /enum firmware
Windows Boot Manager
  identifier {fb000000-0000-0000-0000-000000000000}
  displayorder 0
  timeout 30
  windows Boot Manager
    identifier {bootmgr}
    displayorder 1
    timeout 30
    device partition:\\Device\\HarddiskVolume1
    description Windows Boot Manager
    locale en-US
    inherit {globalsettings}
    osdevice {current}
    resumeobject {current}
    displayorder {current}
    timeout 30
  firmware Application (0x0fffff)
  identifier {00000000-0000-0000-0000-000000000000}
  description Uloop
  firmware Application (0x0fffff)
  identifier {00000000-0000-0000-0000-000000000000}
  description UEFI Misc Device
  firmware Application (0x0fffff)
  identifier {00000000-0000-0000-0000-000000000000}
  device partition:\\Device\\HarddiskVolume1
  path \Device\Hdreds\efi
  description Ubuntu
C:\Windows\system32>bcdedit /set {fwbootmgr} displayorder {00000000-0000-0000-0000-000000000000} /addfirst
The operation completed successfully.

C:\Windows\system32>
```

- This will set Linux's bootloader entry at first priority, so GRUB will start after booting instead of directly going to windows. GRUB provides choice to boot either windows or linux.

```
[Administrator: Command Prompt]
C:\Windows\system32\rsrcedit /enum Firmware
Windows Boot Manager
identifier: {00000000-0000-0000-0000-000000000000}
displayorder: 0
timeout: 0
Windows Boot Manager
identifier: {bootmgr}
device: \Device\HarddiskVolume1\Windows\Boot\bootmgfw.efi
path: \Windows\Boot\bootmgfw.efi
description: Windows Boot Manager
osloader: \Windows\system32\cmd.exe
isBoot: 1
isMultiBoot: 0
isRemovable: 0
removableObject: {d8c5cfe1-091c-11f8-b499-4644e0da7c1c}
isDismImage: 0
isDismImageProvider: {wendag}
isEfi: 0
Firmware Application ({00000000-0000-0000-0000-000000000000})
identifier: {446c794d-081c-11f8-b499-4644e0da7c1c}
description: UApp
Firmware Application ({00000000-0000-0000-0000-000000000000})
identifier: {446c794e-081c-11f8-b499-4644e0da7c1c}
description: UFS File Device
Firmware Application ({00000000-0000-0000-0000-000000000000})
identifier: {00000000-0000-0000-0000-000000000000}
device: \Device\HarddiskVolume1
partition: \Device\HarddiskVolume1
path: \Device\HarddiskVolume1\Windows\Ubuntu
description: Ubuntu
C:\Windows\system32>
```