https://developer.nvidia.com/compute/cuda/8.0/Prod2/local_installers/cuda-repo-ubuntu1604-8-0-local-ga2_8.0.61-1_amd64-debwithout cut download untill 1.89 to 19.9 gb file download in linux



To **download large files** like the CUDA 8.0 .deb installer from a **direct link** using the Linux terminal, especially when the file is very large (like 1.89 GB to 19.9 GB), you can use tools that support **resumable downloads** and **robust connections**. Here's how you can do it step-by-step:

Step-by-Step: Download CUDA 8.0 Installer on Linux

Option 1: Using wget (supports resume)



- -c means continue a partial download if it gets interrupted.
- This method is **robust** for large files.

Option 2: Using aria2c (faster with multi-connection)

Install aria2 first:



Then run:



- -x 16 allows up to 16 connections (faster).
- -s 16 splits into 16 segments.
- Better for **slow or unstable** networks.

Optional: Check File Size While Downloading

In another terminal, you can monitor the file size:



After Downloading: Install CUDA .deb

CUDA download Linux metho	CUDA download Linux methods		
bash	Сору	to Edit	
sudo dpkg -i cuda-repo-ubuntu1604-8-0-local-ga2_8.0.61-1_amd64.deb /var/cuda-repo-*/7fa2af80.pub sudo apt update sudo apt install cuda	· ·	ey add	
Would you like help verifying if this specific CUDA version works with your curre う $^{\circ}$ の $^{\circ}$ の $^{\circ}$ の $^{\circ}$	nt GPU or dri	ver setup?	
	ell gedit expo	rt source e	
If you want to export a file you're editing in <code>gedit</code> (the GNOME text editor) a to save or export the source code with syntax highlighting , here are the possi	-	ing how	
✓ 1. Just Save Normally (Plain Text)			
To save your source file:			
• Use File → Save As			
• Or press Ctrl + Shift + S			
Choose a location and file name			
 It will be saved in plain text format (no colors or formatting) 			
2. Export with Syntax Highlighting (e.g., to PDF)			
${\tt gedit}\ \ {\tt does}\ {\tt not}\ {\tt support}\ {\tt exporting}\ {\tt to}\ {\tt PDF}\ {\tt with}\ {\tt syntax}\ {\tt highlighting}\ {\tt directly}.$			
But you can use this workaround:			
Option A: Use highlight Command-Line Tool			
1. Install highlight:			
bash	Сору	* Edit	
sudo apt install highlight			
2. Export a source file to PDF or HTML:			
bash	Сору	to Edit	
highlight -O pdf -o output.pdf source_file.c			
Or for HTML:			
bash	Сору	† ⊘ Edit	
highlight -O html -o output.html source_file.py			

This gives you a **color-coded export** of your source.

✓ 3. Alternative Editors (with Export)

If you need built-in export features:

- gedit + plugin: There's no official PDF export plugin.
- Use VS Code, Sublime Text, or Notepad++ on Wine for better export options.
- \bullet Or use enscript , a2ps , or highlight from CLI for rich formatting.

Would you like me to show how to install and use highlight with a real example?

中的安全等的人

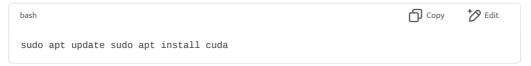
sudo apt install cuda tell gedit export source e

Let's break this into two parts since you mentioned:

- 1. sudo apt install cuda
- 2. tell gedit export source e

✓ 1. sudo apt install cuda

This command installs the full CUDA toolkit from NVIDIA **after** you have added the correct CUDA repository via .deb or apt setup. Example:



Make sure you've already run:



Then the sudo apt install cuda will work correctly and install CUDA 8.0.

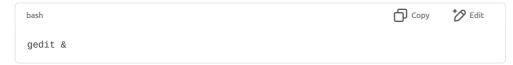
2. "tell gedit export source e"

If you mean:

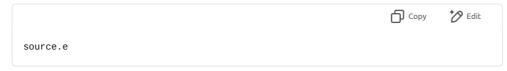
How to export the code you wrote in <code>gedit</code> into a file named <code>source.e?</code>

➤ Steps:

1. Open gedit:



- 2. Write your code or paste content.
- 3. **Go to:** File → Save As
- 4. In the "Name" field, type:



5. Click "Save"

Now you've saved/exported your code into a file named source.e.

If you're asking how to **run** or **compile** a file named source.e , please let me know what language source.e contains (e.g., C, Euphoria, Eiffel). Want help with that too?



cuda environment variables

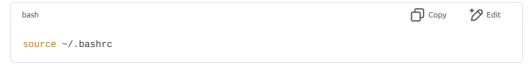
To properly set up **CUDA environment variables** after installing it (e.g., CUDA 8.0), you need to add paths to your ~/.bashrc or ~/.zshrc file (depending on your shell).

CUDA Environment Variable Setup

Add the following lines to your shell config file (~/.bashrc for most users):

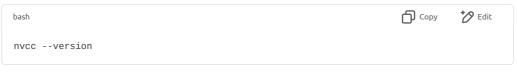


Then apply changes:

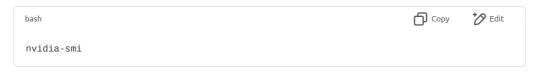


Confirm CUDA Setup

Check nvcc compiler:



Check if GPU is visible:



X Alternate Versions

If you're using a different CUDA version (e.g., 12.2), adjust the path accordingly:



Would you like me to auto-generate the $% \left(1\right) =\left(1\right) +\left(1\right) =\left(1\right) +\left(1\right) +\left(1\right) =\left(1\right) +\left(1\right) +\left($