

AI on NVIDIA Jetson Nano

We'll learn how to:

- Set up the Jetson Nano Developer Kit and camera
- Collect varied data for image classification projects
- Train neural network models for classification
- Annotate image data for regression
- Train neural network models for regression to localize features
- Run inference on live camera feed with you trained models

Outline

- Setting Up the Jetson Nano
- Image Classification
- Image Regression

Setting up the Jetson Nano

- **Introduction and Setup**
 - Setting up JetPack and what items we need to get started
- **Cameras**
 - Details on how to connect the camera to the Jetson Nano
- **Headless Device Mode**
 - Running the Docker container for the this training using headless device mode (remotely from your computer)
- **Hello Camera**
 - How to test the camera with an interactive Jupyter notebook on the Jetson Nano
- **JupyterLab**
 - A brief introduction to the JupyterLab interface and notebooks

Image Classification

- **AI and Deep Learning**
 - A brief overview of Deep Learning and how it relates to AI
- **Convolutional Neural Networks (CNNs)**
 - An introduction to the dominant class of ANN for computer vision tasks
- **ResNet-18**
 - Specific on the ResNet-18 network architecture and used in the class project
- **Thumbs Project**
 - Instruction to work with interactive image classification notebook to create the first project
- **Emotions Project**
 - Build a new project with the same classification notebook to detect emotions from facial expressions

Image Regression

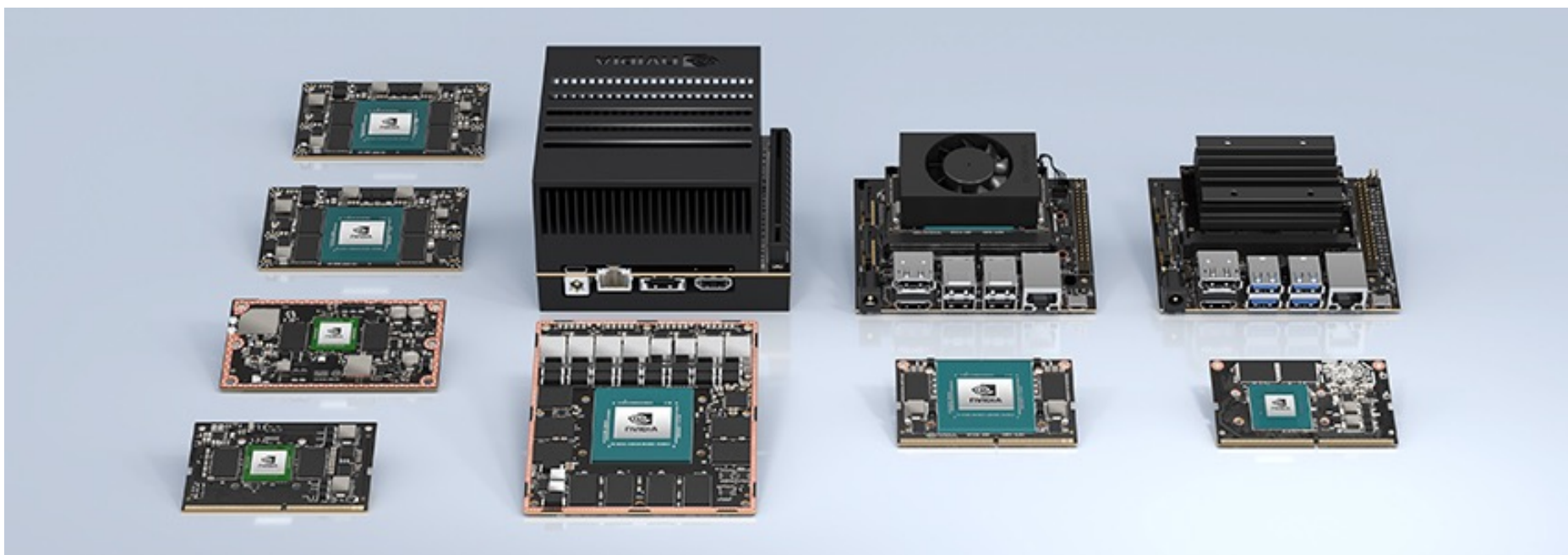
- **Classification vs. Regression**
 - With a few changes, the classification model can be converted to a Regression model
- **Face XY Project**
 - Instruction to build a project that finds the coordinated of facial features

Prerequisites

- Jetson Nano Developer Kit
- Computer with Internet Access and SD card port
- microSD Memory Card (32GB UHS-I minimum)
- Compatible 5V 4A Power Supply with 2.1mm DC barrel connector
- 2-pin jumper
- USB cable (Micro-B to Type-A)
- Compatible USB webcam camera such as
 - Logitech C270 Webcam

The Jetson Family

- Jetson AGX Xavier Developer Kit
- Jetson Xavier NX Developer Kit
- Jetson TX2 Developer Kit
- Jetson TX1 Developer Kit
- Jetson Nano Developer Kit
- Jetson Nano 2G Developer Kit



More info: <https://developer.nvidia.com/embedded/jetson-modules>

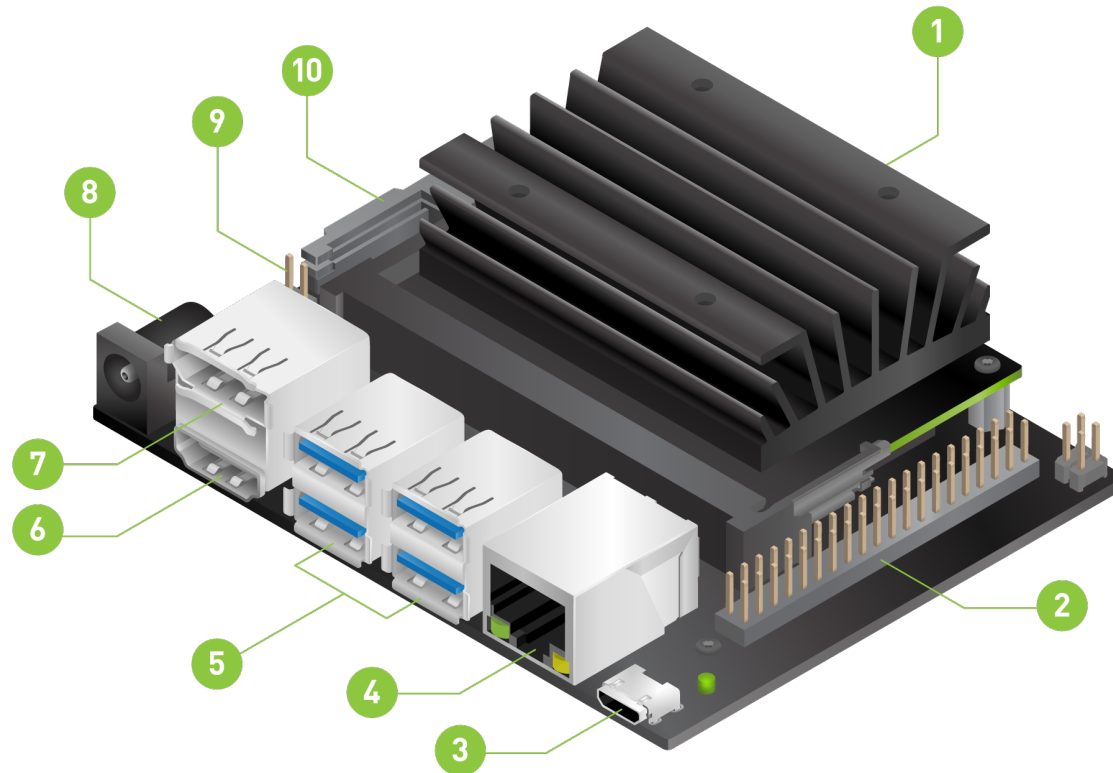
What is the NVIDIA JetPack SDK?

- NVIDIA Jetson Software Development Kit (SDK) is the most comprehensive solution for building end-to-end accelerated AI applications.
- NVIDIA Jetson SDK provides a package of APIs, drop-in libraries, and integration with common languages in the **CUDA Toolkit**. (Don't need to know how to program a GPU in order to program a GPU)
- JetPack 4.6 release include:
 - Linux for Tegra R32.6.1 (K4.9) (Jetson Linux Driver Package with Ubuntu 18.04)
 - TensorRT 8.0.1 (SDK for optimizing trained deep learning models)
 - Supported with cuDNN, TensorFlow, PyTorch, and ONNX
 - CUDA Toolkit 10.2 (CUDA-X accelerated libraries)
 - cuDNN 8.2.1 (API for Deep Learning)
 - OpenCV 4.1.1, Visionworks 1.6 (Computer Vision Libraries)

More info: <https://developer.nvidia.com/embedded/jetpack>

NVIDIA Jetson Nano

- NVIDIA Jetson Nano Developer Kit



Introduction and Setup

- Set Up JetPack
 - Demo: First time setup with JetPack
 - Download: balenaEtcher
 - <https://www.balena.io/etcher/>
 - Download: JetPack SDK
 - <https://developer.nvidia.com/embedded/jetpack#install>
- How to check the JetPack version you installed?
 - Open the Terminal and type this command below:

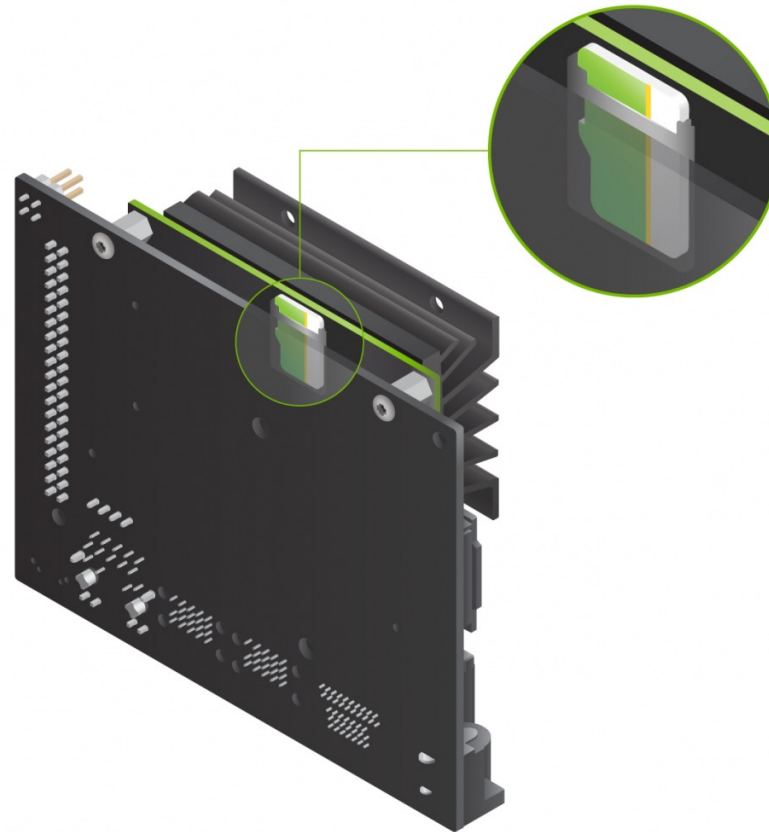
```
$ jetson_release -v
```

Introduction and Setup (cont'd)

- Write Image to the microSD Card
- Initial Setup with Display Attached

Introduction and Setup (cont'd)

- Unfold the paper stand and place inside the developer kit box
- Insert the microSD card



Introduction and Setup (cont'd)

- Set the developer kit on top of paper stand
- Power on your computer display and connect it
- Connect your Micro-USB power supply or using DC a power supply with a barrel jack connector

** The Jetson nano developer kit will power on and boot automatically.

Introduction and Setup (cont'd)

- After Logging In



Install Jetson Stats (Optional)

- Install command

```
$ sudo -H pip install -U jetson-stats
```

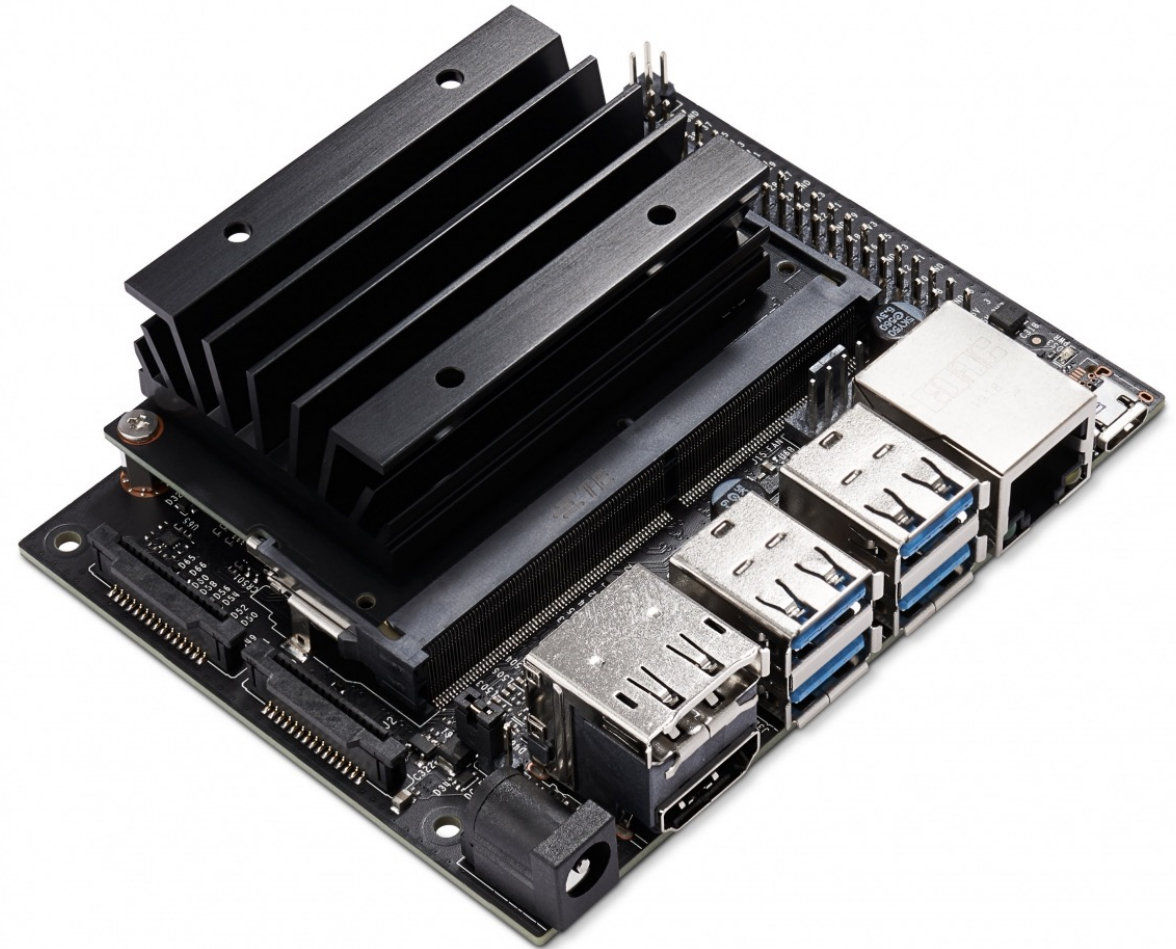
- Run the jtop simple using a simple command

```
$ jtop
```

- https://github.com/rbonghi/jetson_stats

NVIDIA Jetson Nano Developer Kit

- The NVIDIA Jetson Nano Developer Kit is a small AI computer for makers, learners, and developers.
- After following along with this guide, we'll be ready to start building practical AI applications, cool AI robots, and more.



Items for Getting Started

- To get started, we'll need to set up a microSD card for the operating system and main storage.
- Follow the NVIDIA instructions for either the original 4GB Jetson Nano Developer Kit or the new Jetson Nano 2GB Developer Kit to download and flash the latest JetPack image for the system.
- **64GB MicroSD Card**
 - Recommend a larger microSD card to ensure there is plenty of room for the applications and data.

Items for Getting Started (cont'd)

- Internet Connection for the Jetson Nano
 - We'll need an Internet connection through either the Ethernet port or compatible Wi-Fi device the first time we run the course notebooks.
 - We'll execute a command that downloads a Docker image from the NVIDIA NGC cloud.
 - After that, as long as the Docker image is not deleted from the microSD card, we won't need the internet connection on the Jetson Nano to run the course notebooks.

Items for Getting Started (cont'd)

- USB Webcam

- We'll need a camera to capture images in the course projects.
- An example of a compatible camera, NVIDIA has verified that the Logitech C270 USB webcam works with these projects.
- The ability to position the camera easily for capturing images hands-free makes this a great choice.
- Some other USB webcams may also work with the projects.



Items for Getting Started (cont'd)

- USB Data Cable (Micro-B to Type –A)
 - We'll also need a Micro USB to USB-A cable to directly connect the computer to the Jetson Nano Developer Kit's Micro USB port.
 - The cable must be capable of data transfers, rather than only designed to power a device.



Items for Getting Started (cont'd)

- **Original 4GB Version Only: Alternate Power Supply**
 - The 5V 4A DC barrel jack power supply is required.
 - Although it is possible to power the original 4GB Jetson Nano with a smaller microUSB supply, this is not robust enough for the high GPU compute load we require for our projects.
 - In addition, you will need the microUSB port available as a direct connection to your computer for this course (not recommended)
- **5V 4A Power Supply with 2.1mm DC barrel connector**
 - The barrel jack must be 5.5mm OD x 2.1mm ID x 9.5mm length, center-positive.
- **2-pin Jumper**
 - To specify use of the barrel-type power supply on the Jetson Nano Developer Kit, a 2-pin jumper is required.



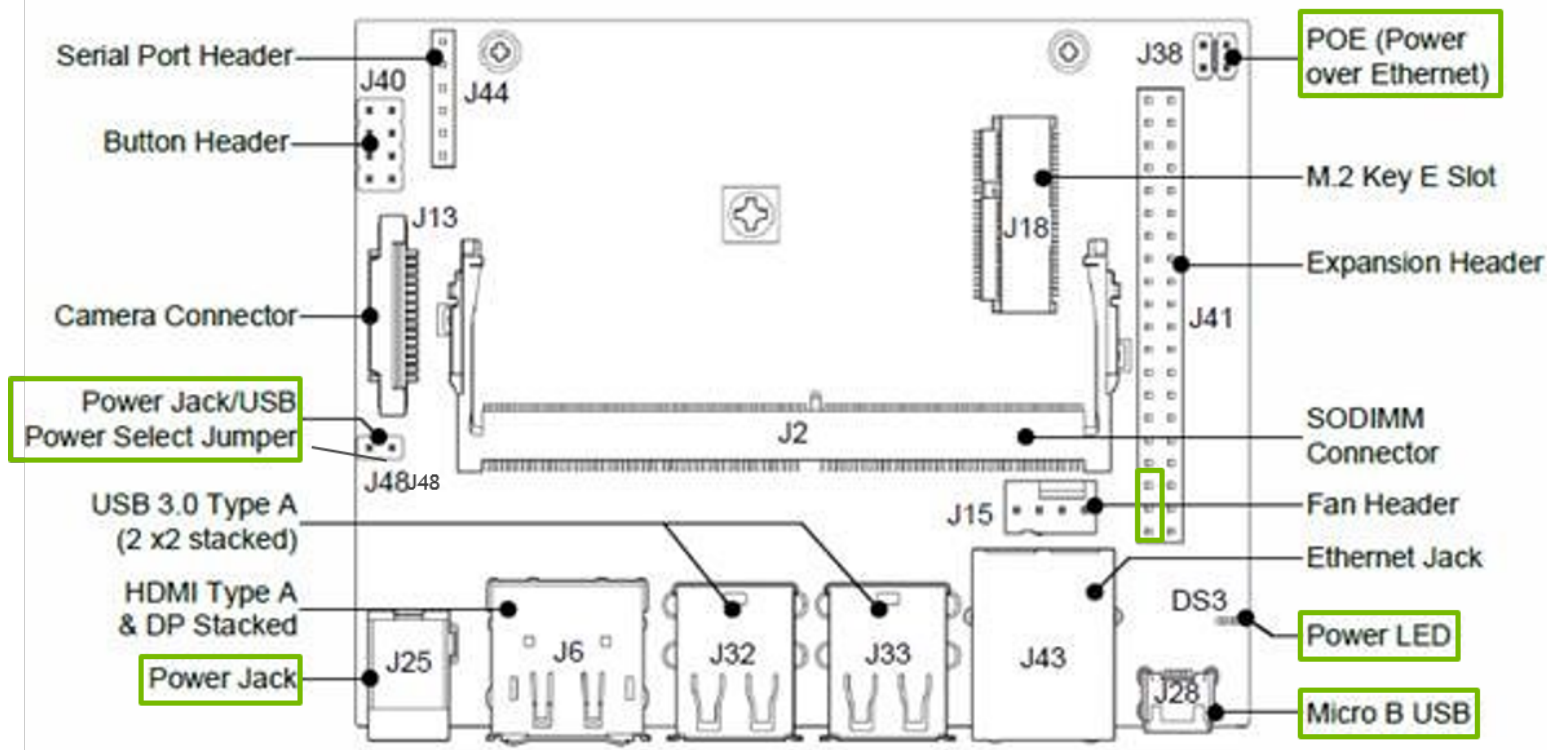
Notes About Setup

- The course runs best if there is a “swap” file of size 4GB, so that if the Jetson Nano is a little short of RAM it can extend a bit by swapping with some of the (slower) disk space.
- After setting up the microSD card and booting the system, check the memory and swap values with this command, which shows how values in megabytes.

```
$ free -m
```


Barrel Power Supply Setup

- A02 version



1) Insert the 2-pin jumper across the 2-pin connector, J48, located next to the MIPI CSI camera connector the barrel power

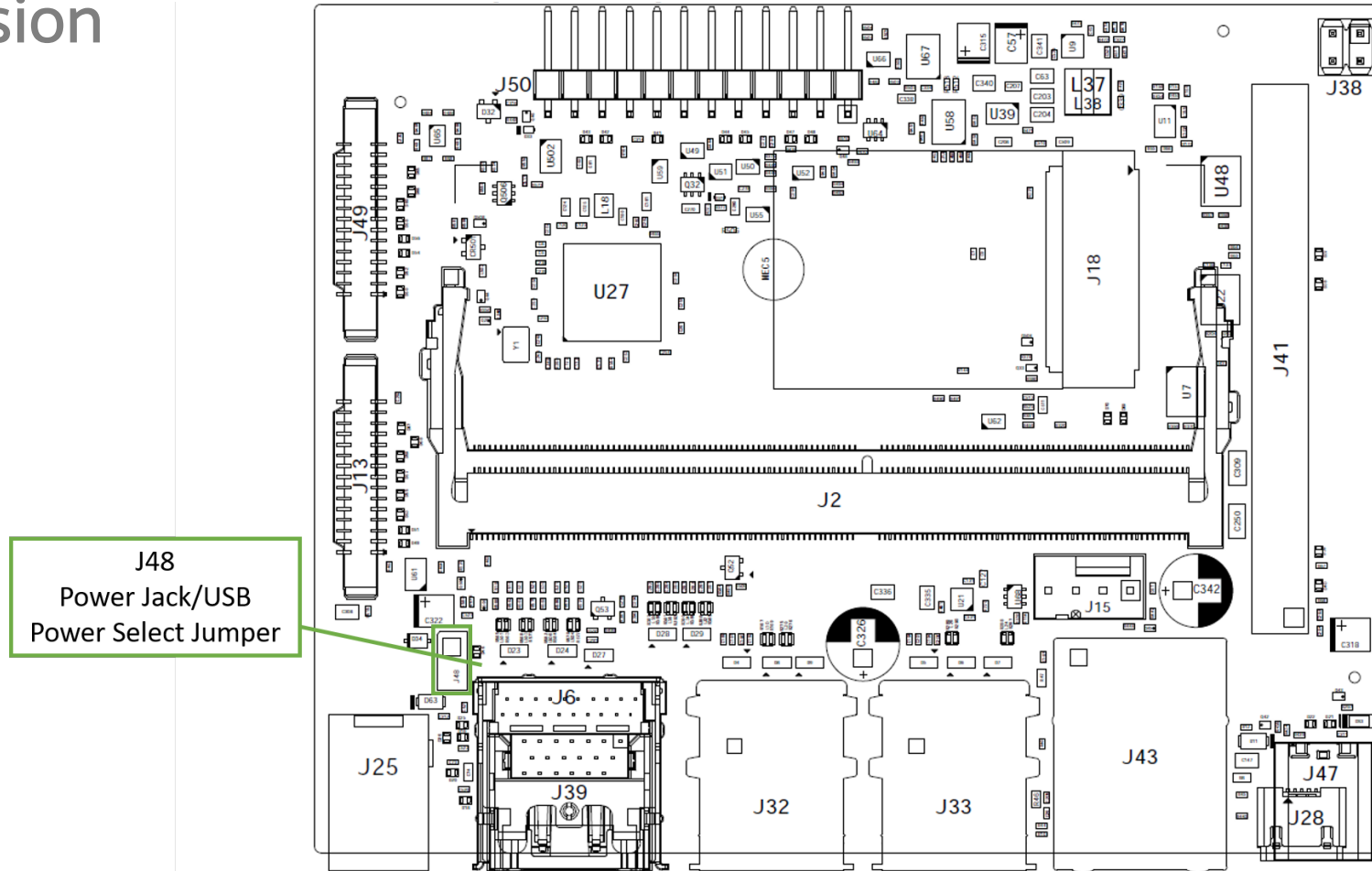
Port (B01 version). This enables the DC barrel power supply.

2) Connect your DC barrel jack power supply (5V/4A). The jetson Nano Developer Kit will power on and boot automatically.

3) A green LED next to the Micro-USB connector will light as soon as the developer kit powers on.

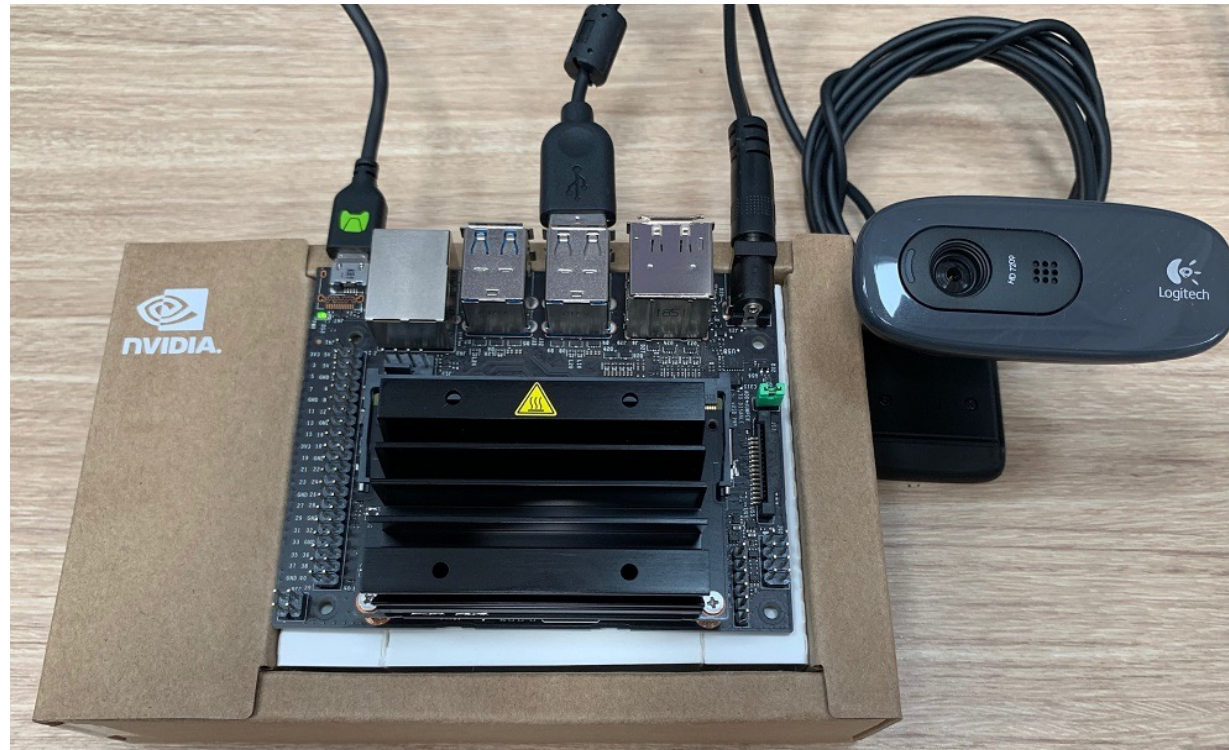
Barrel Power Supply Setup (cont'd)

- B01 version



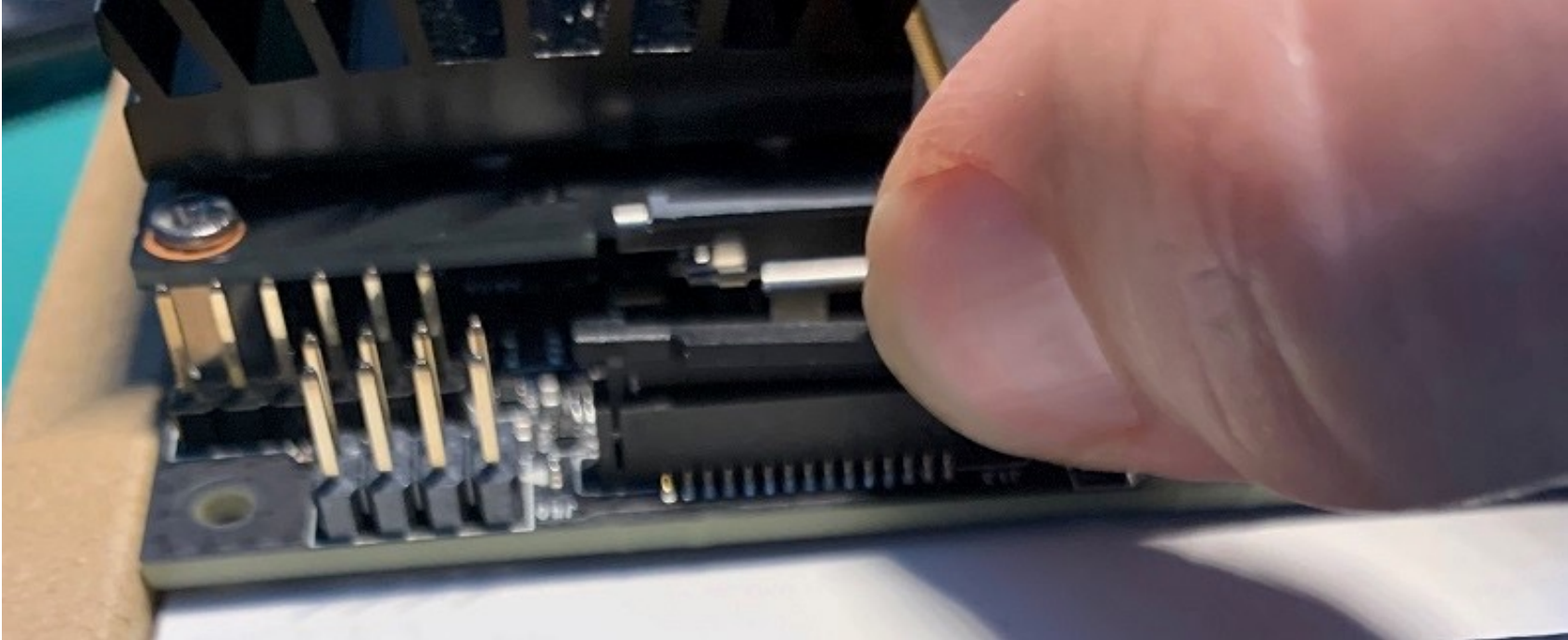
Cameras

- Logitech C270 USB Webcam (Recommended Configuration)
 - This very straightforward. Just plug in in the USB connector into any of the Jetson Nano Developer Kit USB ports.



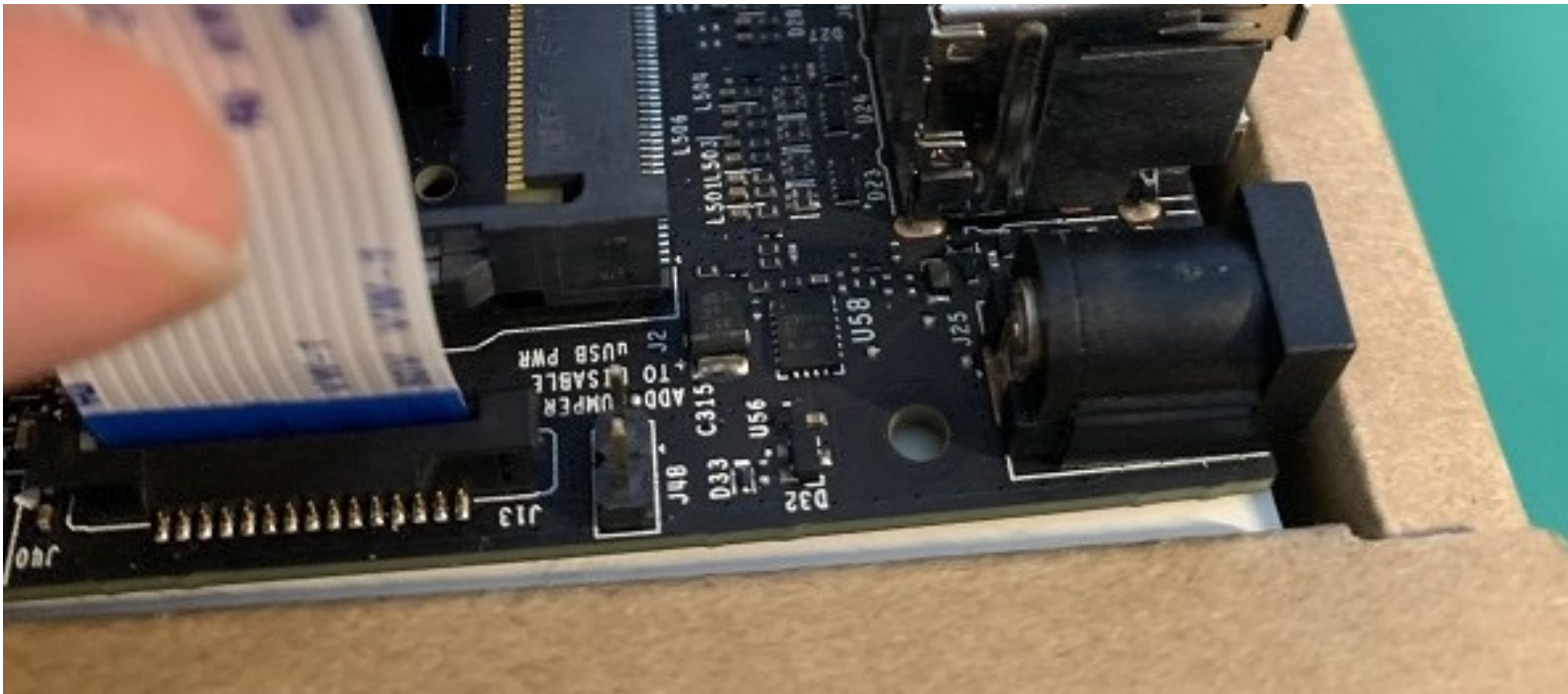
Cameras (cont'd)

- Raspberry Pi V2 Camera (Alternate Configuration)
 - 1) If you want to try this lab with a Raspberry Pi v2 Camera, you will need to connect to MIPI CSI port. Begin unlatching the MIPI CSI connector. This loosens the “grip” of the connector by just a small amount.



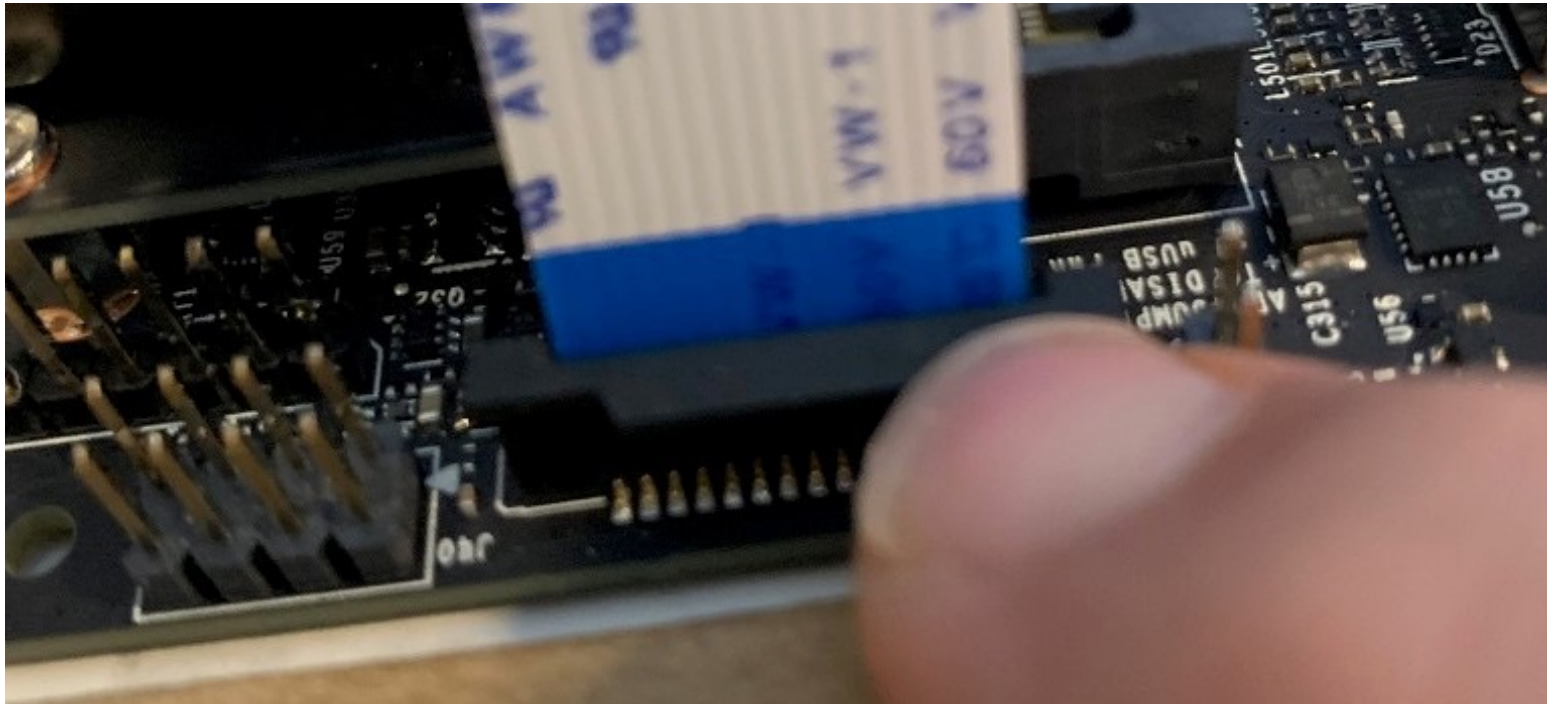
Cameras (cont'd)

- Raspberry Pi V2 Camera (Alternate Configuration)
 - 2) Insert the ribbon cable of the camera so that the metal side faces into the Jetson Nano board.



Cameras (cont'd)

- Raspberry Pi V2 Camera (Alternate Configuration)
 - 3) Latch the connector with a gentle push downward on the sides of the plastic. The ribbon cable should be securely held by the connector.



Cameras (cont'd)

- Raspberry Pi V2 Camera (Alternate Configuration)
 - 4) Remove the protective film from the lens of the camera.



Headless Device Mode

- Logitech C270 USB Webcam (Recommended Configuration)
 - This very straightforward. Just plug in in the USB connector into any of the Jetson Nano Developer Kit USB ports.

Headless Device Mode (cont'd)

- We are running the Jetson Nano Developer Kit in a “headless” configuration. That means you do not hook up monitor directly to the Jetson Nano.
- This method conserves memory resources on the Jetson Nano and has the added benefit of eliminating the requirement for extra hardware, i.e. a monitor, keyboard, and mouse.
- In addition, we will further simplify the configuration by using “**USB Device Mode**”. In this mode, your Jetson Nano connects directly to your computer through a USB cable. This eliminates the need for a network connection on the Jetson Nano, as well as the need to determine the IP address on your network. It is always **192.168.55.1** in this mode.

Headless Device Mode (cont'd)

- In this the steps that follow, you'll boot your Jetson Nano and log on from your computer with SSH protocol, a common method for communicating with embedded systems like the Jetson Nano.

Headless Device Mode (cont'd)

- On your computer, open a terminal window if using macOS or Linux and a PowerShell window if using Windows.
- In the terminal, logon to the Jetson Nano with the following command, where <username> is the values you set up on your Jetson Nano during the operation system configuration:

```
$ ssh <username>@192.168.55.1
```

** Enter the pass you configured when asked.

Headless Device Mode (cont'd)

- Add a directory for the course with the following command in the Jetson Nano terminal you've logged into:

```
$ mkdir -p ~/nvidia-data
```

- Run the Docker container with the following command:
 - USB Camera option:

```
$ sudo docker run --runtime nvidia -it --rm --network host \  
  --volume ~/nvdli-data:/nvdli-nano/data \  
  --device /dev/video0 \  
  nvcr.io/nvidia/dli/dli-nano-ai:v2.0.1-r32.6.1
```

Headless Device Mode (cont'd)

- Run the Docker container with the following command:
 - CSI Camera option:

```
$ sudo docker run --runtime nvidia -it --rm --network host \  
  --volume ~/nvdli-data:/nvdli-nano/data \  
  --volume /tmp/argus_socket:/tmp/argus_socket \  
  --device /dev/video0 \ nvcr.io/nvidia/dli/dli-nano-ai:v2.0.1-r32.6.1.1
```

- Connect to JupyterLab
 - When the container is launched. The JupyterLab server will automatically be started.

Headless Device Mode (cont'd)

- Text similar to the following will be printed out to the user:

```
$ allow 10 sec for JupyterLab to start @ http://192.168.55.1:8888 (password dlinano)  
JupyterLab logging location: /var/log/jupyter.log (inside the container)  
You can then navigate the browser on your PC to the URL shown above  
(http://192.168.55.1:8888) and login to JupyterLab with the password dlinano. Then  
proceed with the DLI course as normal.
```

Headless Device Mode (cont'd)

- To create and run a reusable script

```
# create a reusable script
echo "sudo docker run --runtime nvidia -it --rm --network host \
    --volume ~/nvdli-data:/nvdli-nano/data \
    --device /dev/video0 \
    nvcr.io/nvidia/dli/dli-nano-ai:v2.0.1-r32.6.1" > docker_dli_run.sh

# make the script executable
chmod +x docker_dli_run.sh

# run the script
./docker_dli_run.sh
```

Headless Device Mode (cont'd)

- To create and run a reusable script

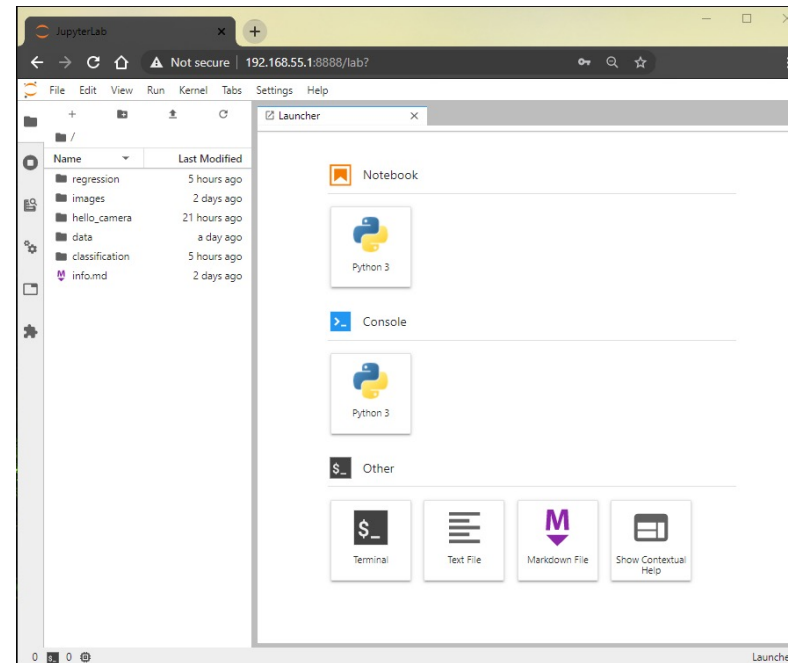
```
# create a reusable script
echo "sudo docker run --runtime nvidia -it --rm --network host \
    --volume ~/nvdli-data:/nvdli-nano/data \
    --device /dev/video0 \
    nvcr.io/nvidia/dli/dli-nano-ai:v2.0.1-r32.6.1" > docker_dli_run.sh

# make the script executable
chmod +x docker_dli_run.sh

# run the script
./docker_dli_run.sh
```

Logging Into The JupyterLab Server

- Open the following link address: 192.168.55.1:8888
The JupyterLab server running on the Jetson Nano will open up with a login prompt the first time.
- Enter the password: **dlinano**



How to install the Jetson-Inference

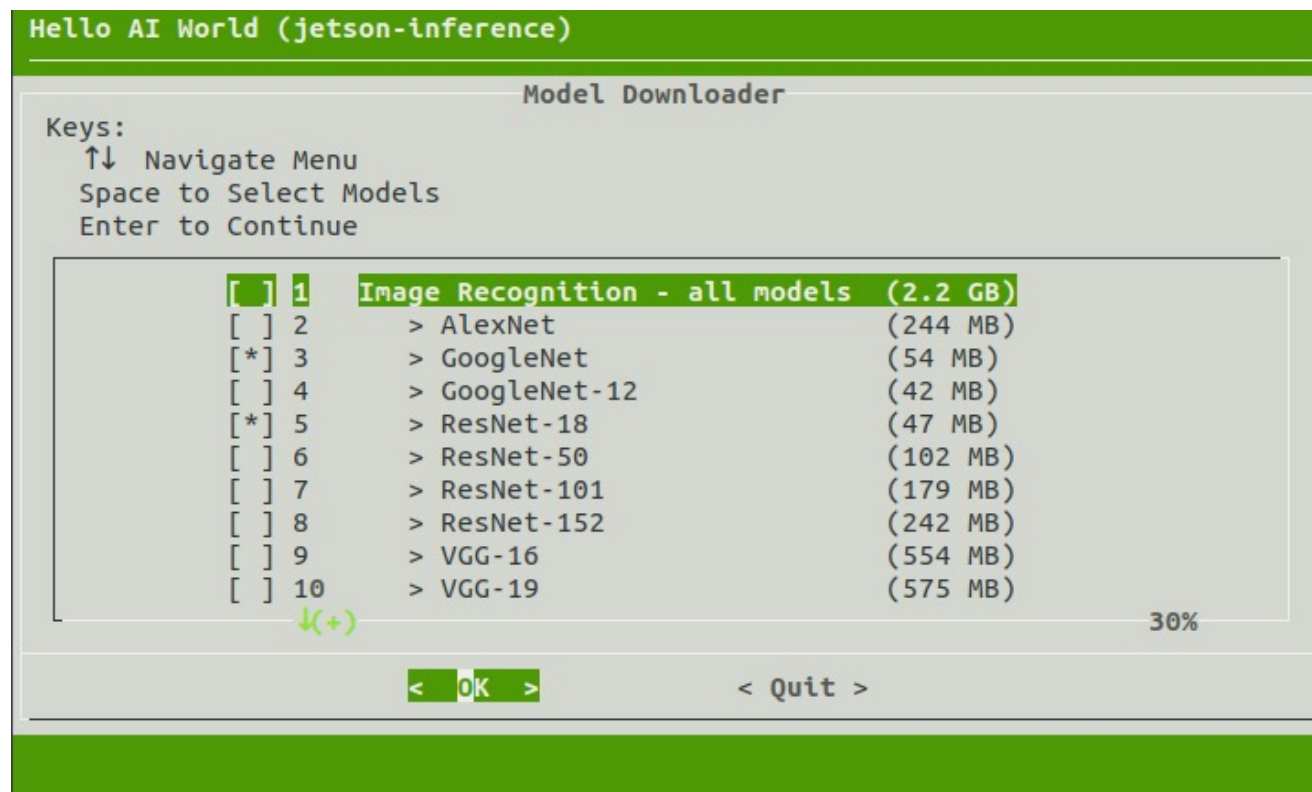
- In the terminal, logon to the Jeton Nano with the following command:

```
$ sudo apt-get update && sudo apt-get upgrade
$ sudo apt-get install git cmake libpython3-dev python3-numpy
$ git clone --recursive https://github.com/dusty-nv/jetson-inferences
$ cd jetson-inference
$ mkdir build
$ cd build
$ cmake ../
$ make -j$(nproc)
$ sudo make install
$ sudo ldconfig
```

*or clone with the -recursive flag: `git submodule update --init`

How to install the Jetson-Inference (cont'd)

- Downloading Models



```
$cd jetson-inference/tools  
$ ./download-models.sh
```

How to install the Jetson-Inference (cont'd)

- Installing Pytorch

```
Hello AI World (jetson-inference)

PyTorch Installer
If you want to train DNN models on your Jetson, this tool will download and
install PyTorch. Select the desired versions of pre-built packages below,
or see http://eLinux.org/Jetson_Zoo for instructions to build from source.

You can skip this step and select Quit if you don't want to install PyTorch.

Keys:
  ↑↓ Navigate Menu
  Space to Select
  Enter to Continue

Packages to Install:
[ * ] 1 PyTorch v1.1.0 for Python 2.7
[   ] 2 PyTorch v1.1.0 for Python 3.6

< OK >      < Quit >
```

```
$cd jetson-inference/tools
$ ./install-pytorch.sh
```

How to install the Jetson-Inference (cont'd)

- Compiling the Project

```
$ cd jetson-inference/build  
$ make  
$ sudo make install  
$ sudo ldconfig
```

Reference

- Getting Started with AI on Jetson Nano
 - <https://courses.nvidia.com/courses/course-v1:DLI+S-RX-02+V2/info>