# DRI WorkShop - Data Analysis

**Tannin Zeraati**

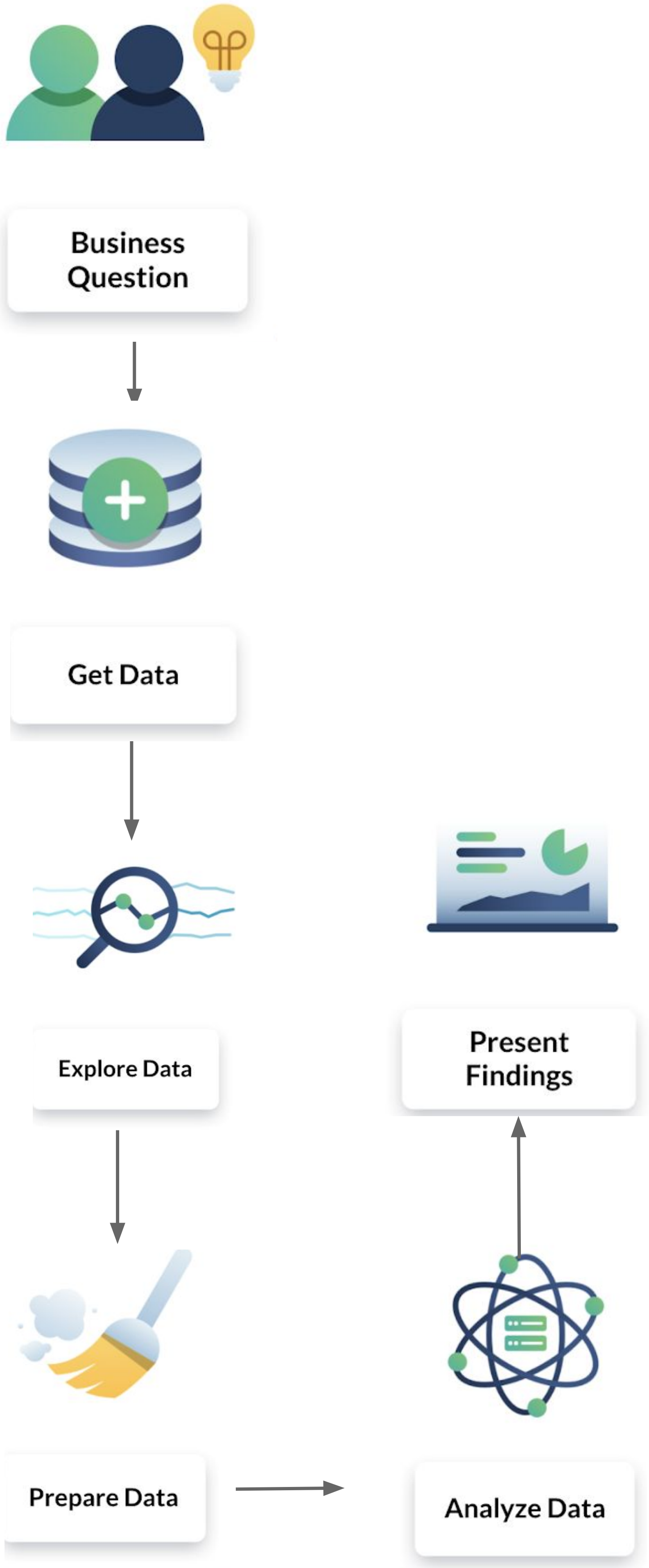
—

Digital Research Alliance of Canada
20 November 2024

# Session Overview

- Introduction to the arc and VM
- Machine learning and LLMs

- Introduction to data analysis
- What to do when your laptop isn’t enough
- The Digital Research Alliance of Canada (ARC)
- Demo
- Using resources - Workshop
- Data analysis Demo



Business Question

Get Data

Explore Data

Prepare Data

Analyze Data

Present Findings

# Data Analysis

The process of examining, cleaning, transforming, and modeling data to uncover insights and support decision-making.

! **Premise:** Turn raw data into actionable insights that drive informed decisions.

---

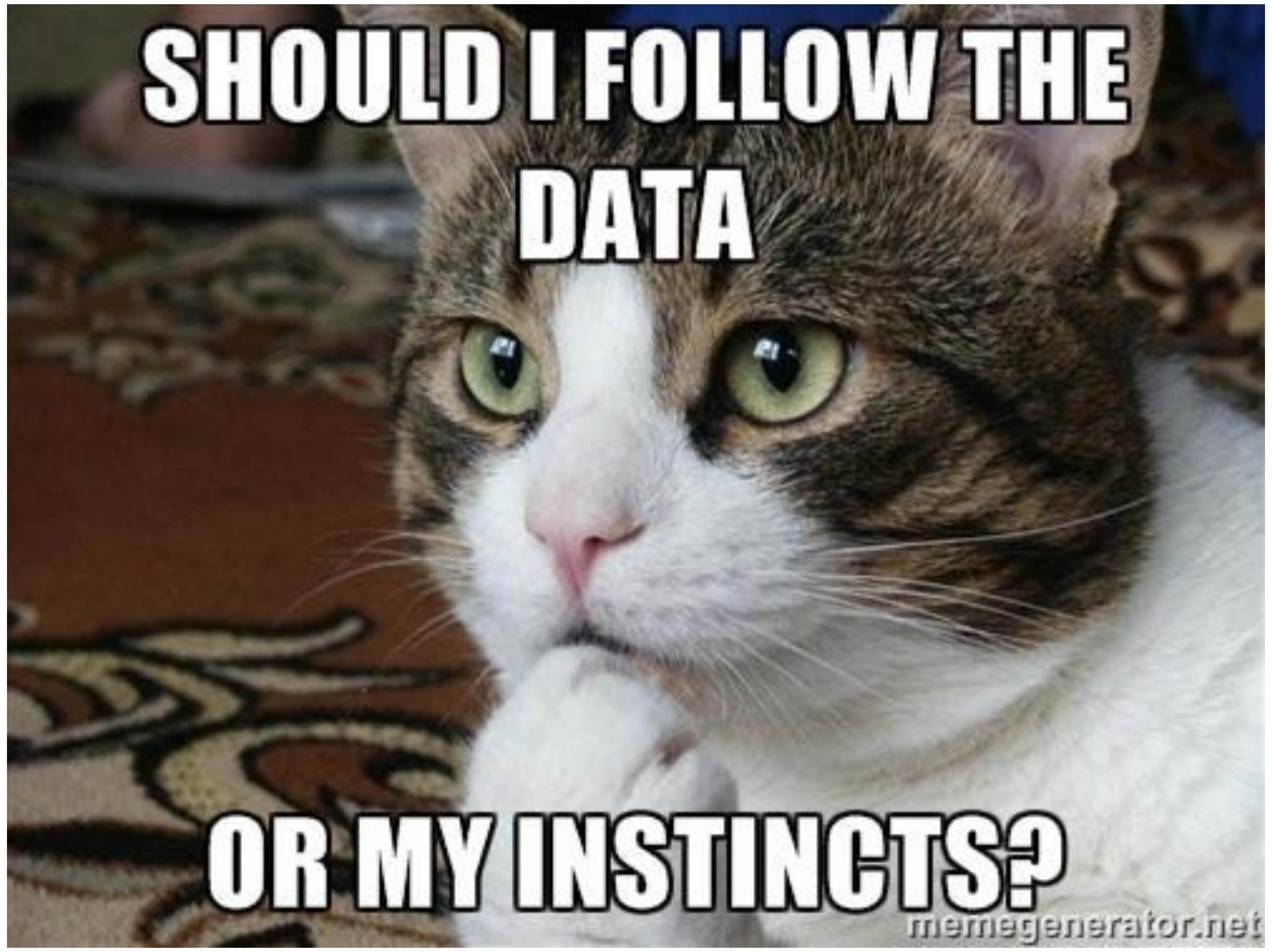
# Data Visualization

Graphical representation of data to make complex information easy to understand and insights more accessible.

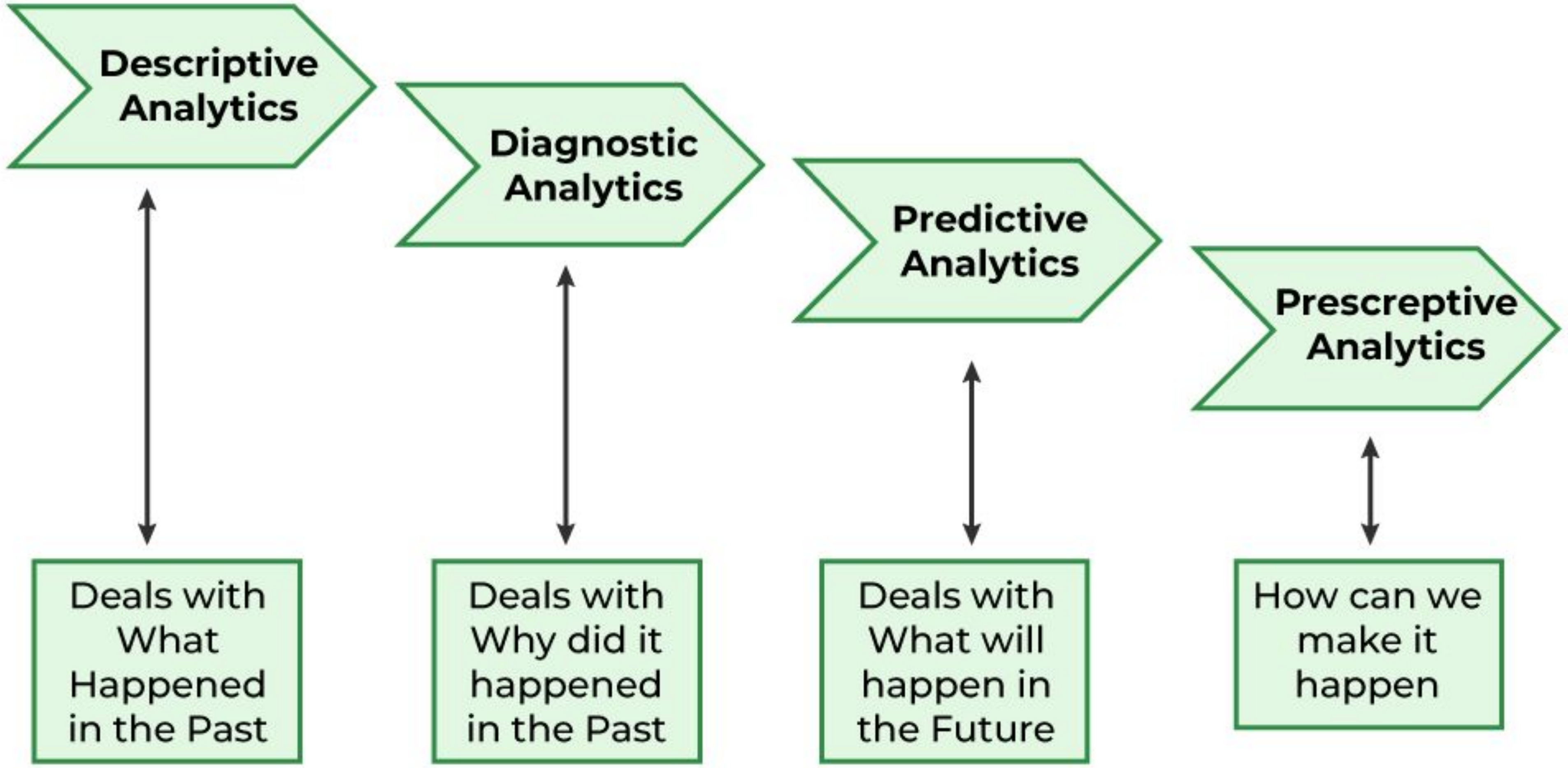
- Simplifies complex data, make it understandable
- Identifies trends and patterns at a glance for quick decision-making
- Improves data-driven communication, making findings more persuasive
- Reveals relationships between variables that may be hard to see in raw data

# Importance

- **Era of digital transformation :Explosion of data generated by digital technologies**
  - **Improves Decision-Making**
  - **Reveals inefficiencies in processes**
  - **Allows for better resource allocation and cost savings**
  - **Forecasts Trends and Risks**
  - **Supports Customer Experience**
  - **Provides insights that inspire new products**
  - **…**



# Types



Descriptive Analytics

Diagnostic Analytics

Predictive Analytics

Prescreptive Analytics

Deals with What Happened in the Past

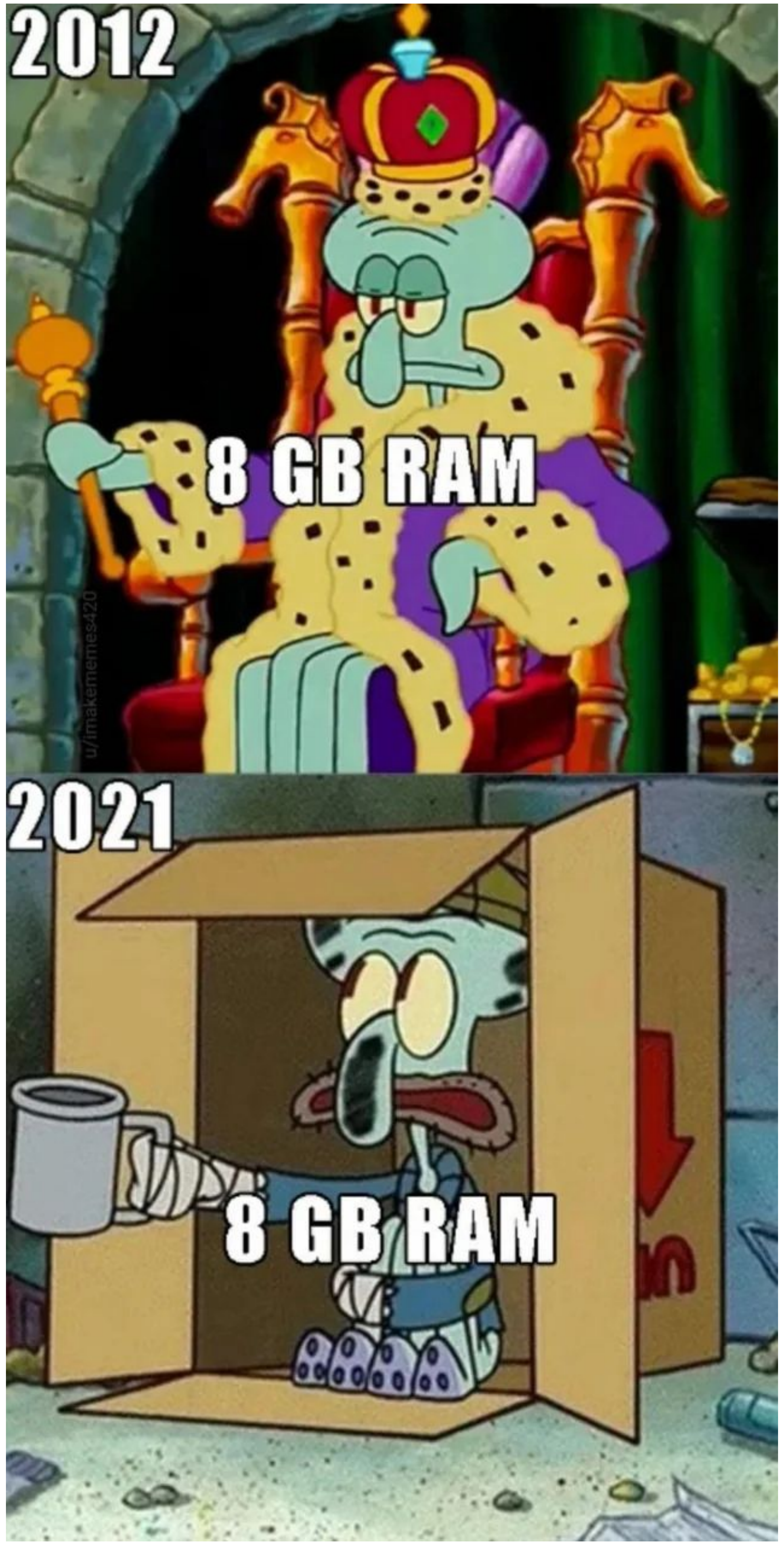
Deals with Why did it happened in the Past

Deals with What will happen in the Future

How can we make it happen

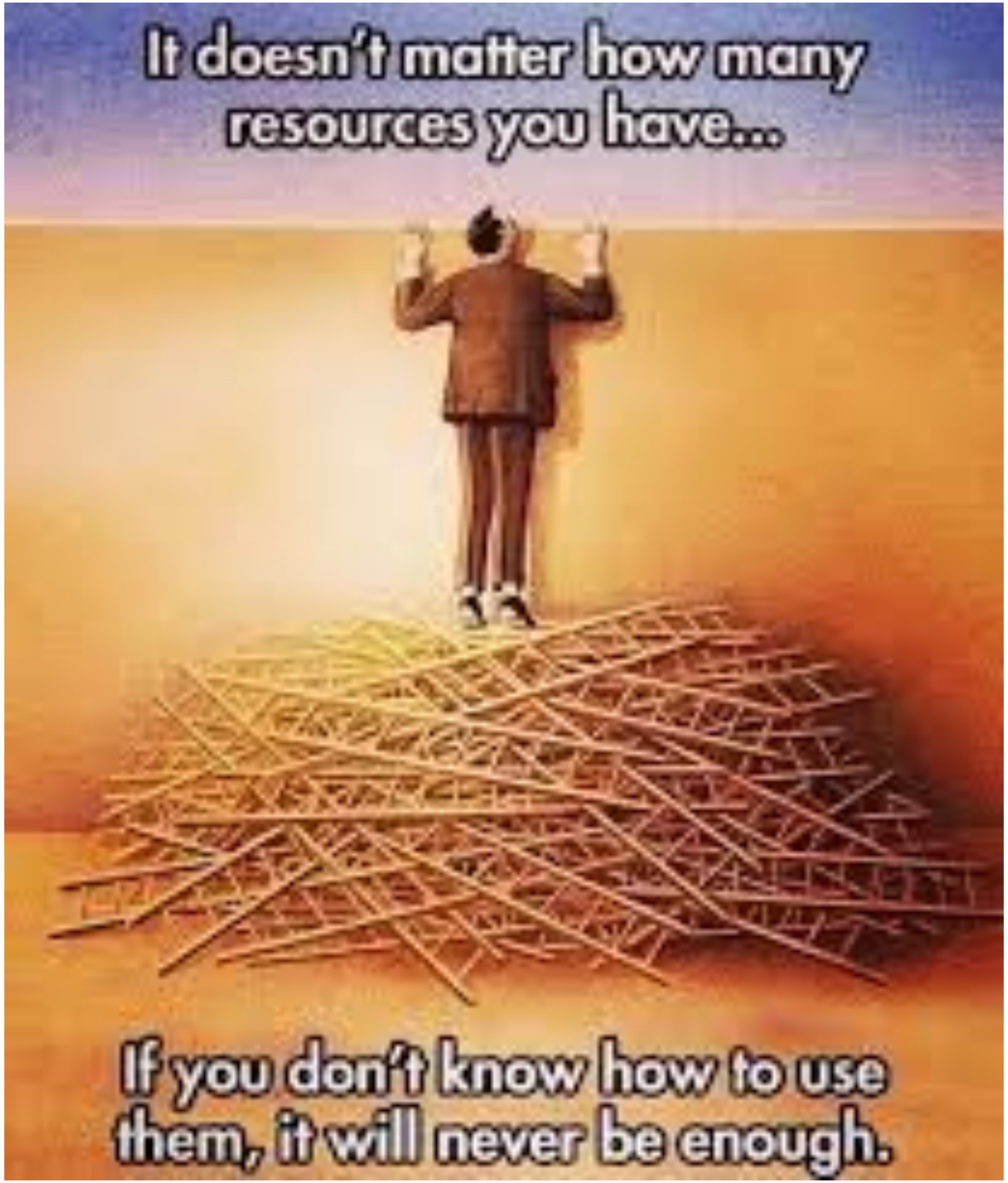
# Laptop not enough

- **Self-purchased**
- **Digital Research Alliance of Canada**
- **Cloud**
- **UBC resources (UBC ARC)**



# Not Enough Resource

- Analysis taking to long
- Not enough computing power
- Not enough storage


It doesn't matter how many resources you have...
If you don't know how to use them, it will never be enough.

# The Digital Research Alliance of Canada (ARC)

- **Mission: To provide Canadian researchers with the tools and infrastructure needed to conduct world-class research.**
- **Key Services:**
  - **Advanced Research Computing (ARC): Provides high-performance computing resources, data storage, and software tools to support computationally intensive research.**
  - **Research Data Management (RDM): Offers services to help researchers manage and preserve their research data.**
  - **Research Software (RS): Supports the development and dissemination of research software.**

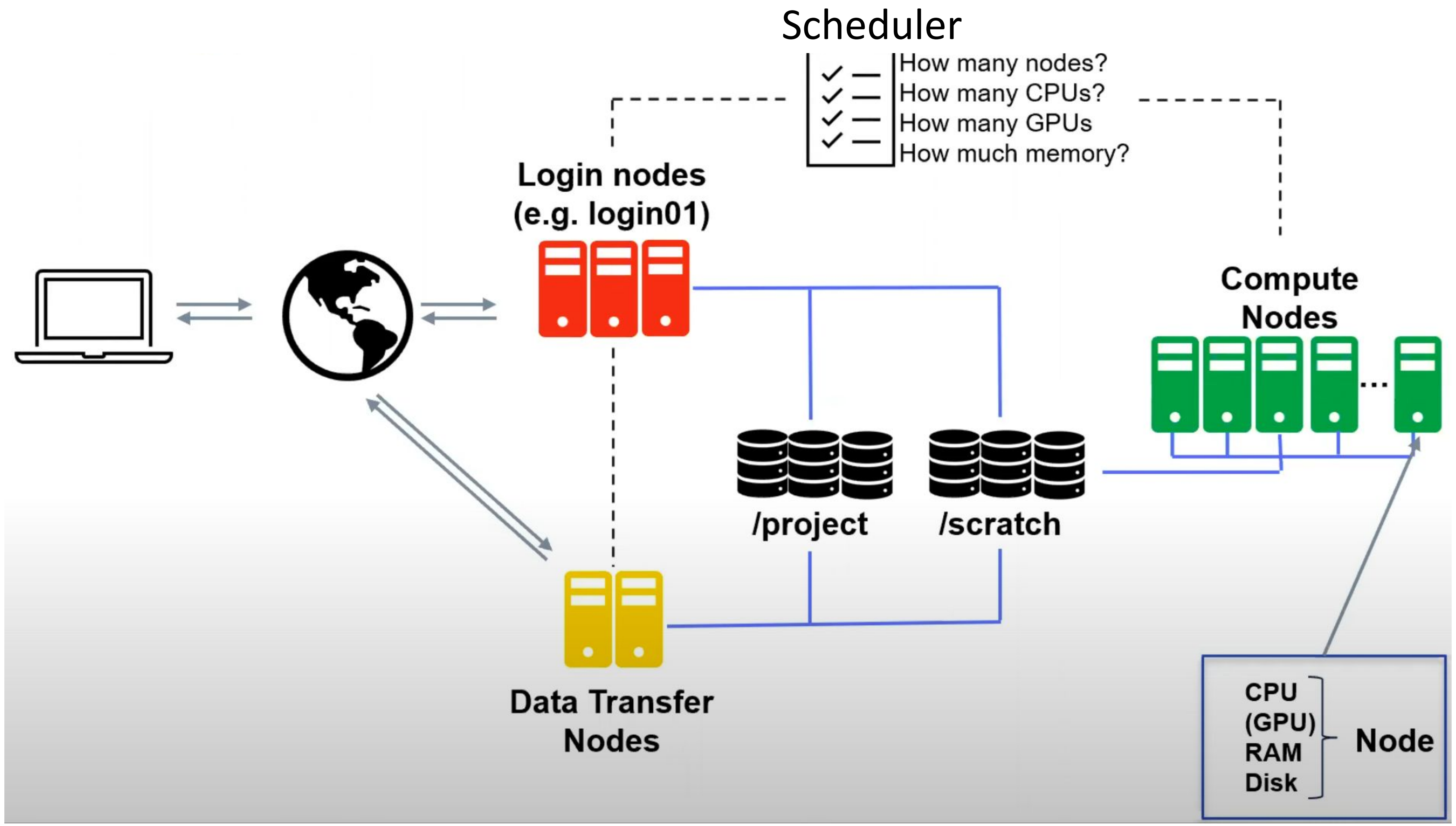
# Account

- Create an account

  https://alliancecan.ca/en/services/advanced-research-computing/account-management/apply-account

- Provide:
  - Your institutional credentials (e.g., university email).
  - Contact information for your sponsor (usually a supervisor or research lead).
  - **Sponsor Account**: Required to access DRI resources.
  - Sponsor information (e.g., name)
- Once submitted, you'll receive credentials and details about your assigned resources.
- Logging in https://www.alliancecan.ca/en

# ARC Structure



UBC ARC https://confluence.it.ubc.ca/display/UARC/Training#Training-TheAlliance

# Structure

- Cluster:
  - A cluster is a group of interconnected computers (nodes) that work together to perform tasks.
  - To handle large-scale computational tasks that a single computer cannot.
  - To divide tasks among multiple nodes for faster processing.
- Node:
  - A node is an individual computer within a cluster.
  - Types of nodes:
    - Login Nodes: Entry points for users to access the cluster. Used for submitting jobs, transferring files, and setting up environments.
    - Compute Nodes: Perform the actual computation. Jobs are executed here.
    - Storage Nodes: Dedicated to managing and storing data.

# Structure

- Job:
  - A job is a task or set of tasks submitted to a cluster's scheduler for execution on the compute nodes.
  - Jobs represent the work you want the system to perform, such as running a program, processing data, or training a machine learning model.
- Scheduler:
  - A scheduler manages how jobs are distributed and executed on the cluster's compute nodes.
  - Slurm (Simple Linux Utility for Resource Management): Widely used on DRI clusters.
  - Prevent resource conflicts between users.
  - Efficiently allocate resources to users.
  - Manage queues for job execution.

# Let’s Start

# Connection

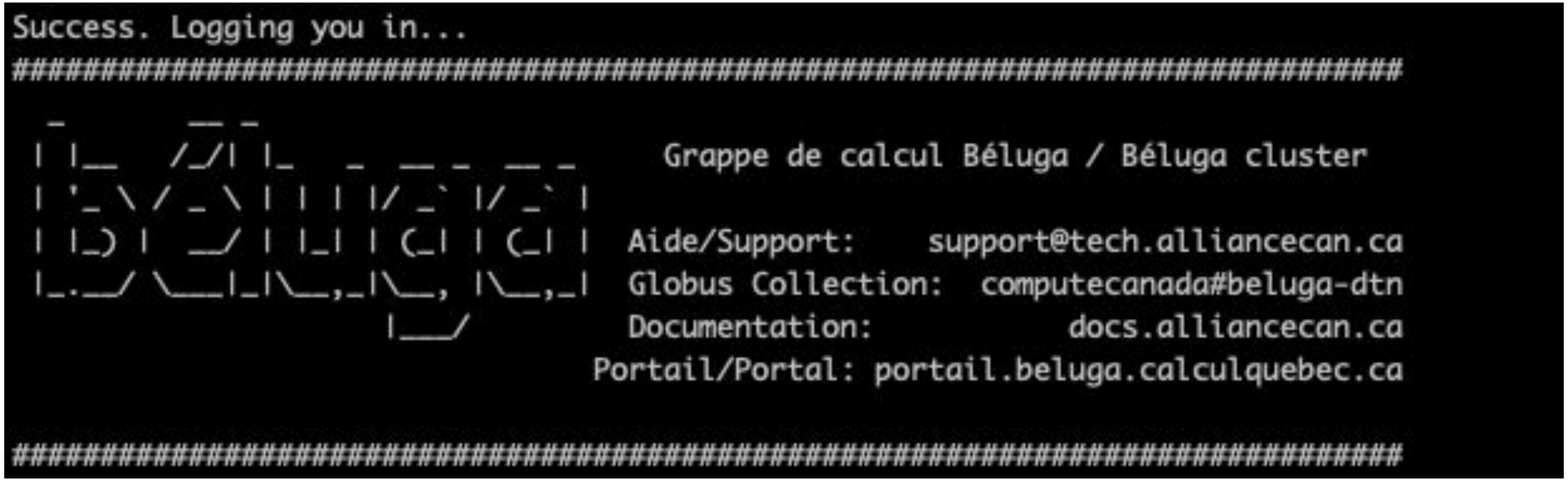
- Terminal
  - The terminal (or command line) is a text-based interface to interact with your computer.
  - search for cmd(windows)/terminal(mac/linux)
  - Host: The server you're connecting to.
  - Username: Your account name on the server.
  - Authentication: Password or SSH keys.
  - SSH: SSH (Secure Shell) is a protocol to securely connect to remote servers (DRI)

# Connection

- SSH

ssh username@cluster_name.dri.ca

okpas30@beluga.computecanada.ca

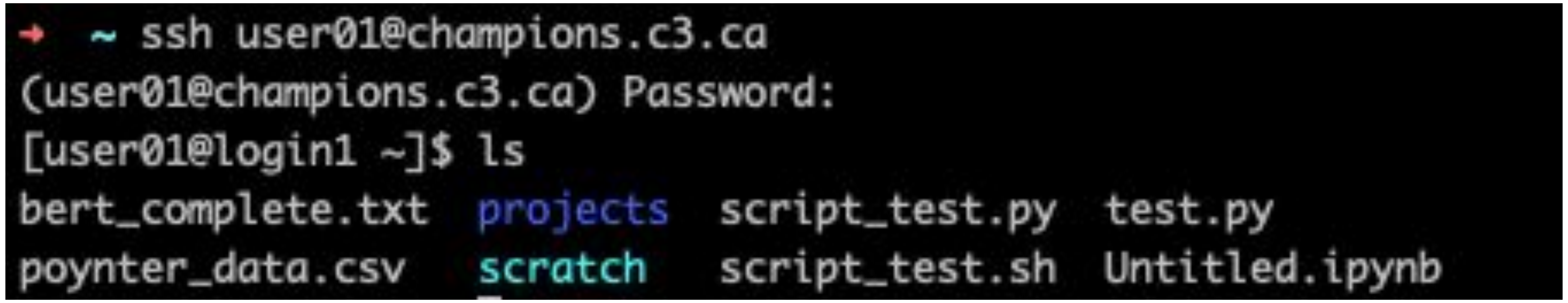


```
Success. Logging you in...
##################################################################################

  _        __ _
 | |__    /_/| |_   _  __ _  __ _        Grappe de calcul Béluga / Béluga cluster
 | '_ \ / _ \ | | | | |/ _` |/ _` |
 | |_) |  __/ | |_| | (_| | (_| |  Aide/Support:     support@tech.alliancecan.ca
 |_.__/ \___|_|\__,_|\__, |\__,_|  Globus Collection:  computecanada#beluga-dtn
                     |___/         Documentation:             docs.alliancecan.ca
                                 Portail/Portal: portail.beluga.calculquebec.ca

##################################################################################
```

# Connection

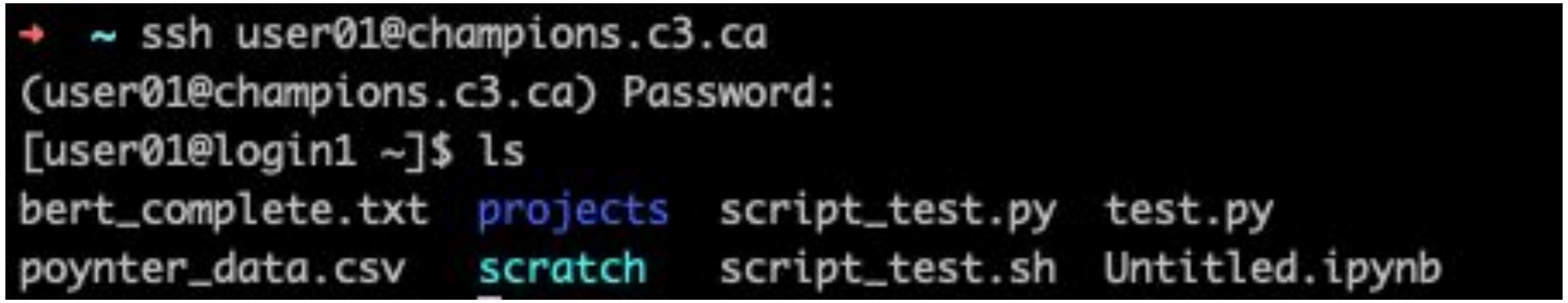
- Terminal

| | |
|---|---|
| ○ Connect to the host | ssh username@hostname |
| ○ for this workshop | username@champions.c3.ca |
| ○ password | champions-24 |



```
➜  ~ ssh user01@champions.c3.ca
(user01@champions.c3.ca) Password:
[user01@login1 ~]$ ls
bert_complete.txt  projects  script_test.py  test.py
poynter_data.csv   scratch   script_test.sh  Untitled.ipynb
```

# Connection



```
➜  ~ ssh user01@champions.c3.ca
(user01@champions.c3.ca) Password:
[user01@login1 ~]$ ls
bert_complete.txt  projects  script_test.py  test.py
poynter_data.csv   scratch   script_test.sh  Untitled.ipynb
```

- projects:  A directory (folder) -> A place where related files and scripts are grouped together.
- scratch: A directory -> Often used as temporary storage for work in progress.

# Data/File types

Data

Script/Code

Intermediate files/Data

Results

Log files

- How big are the files?
- Would you like others to access these files?

# Commands

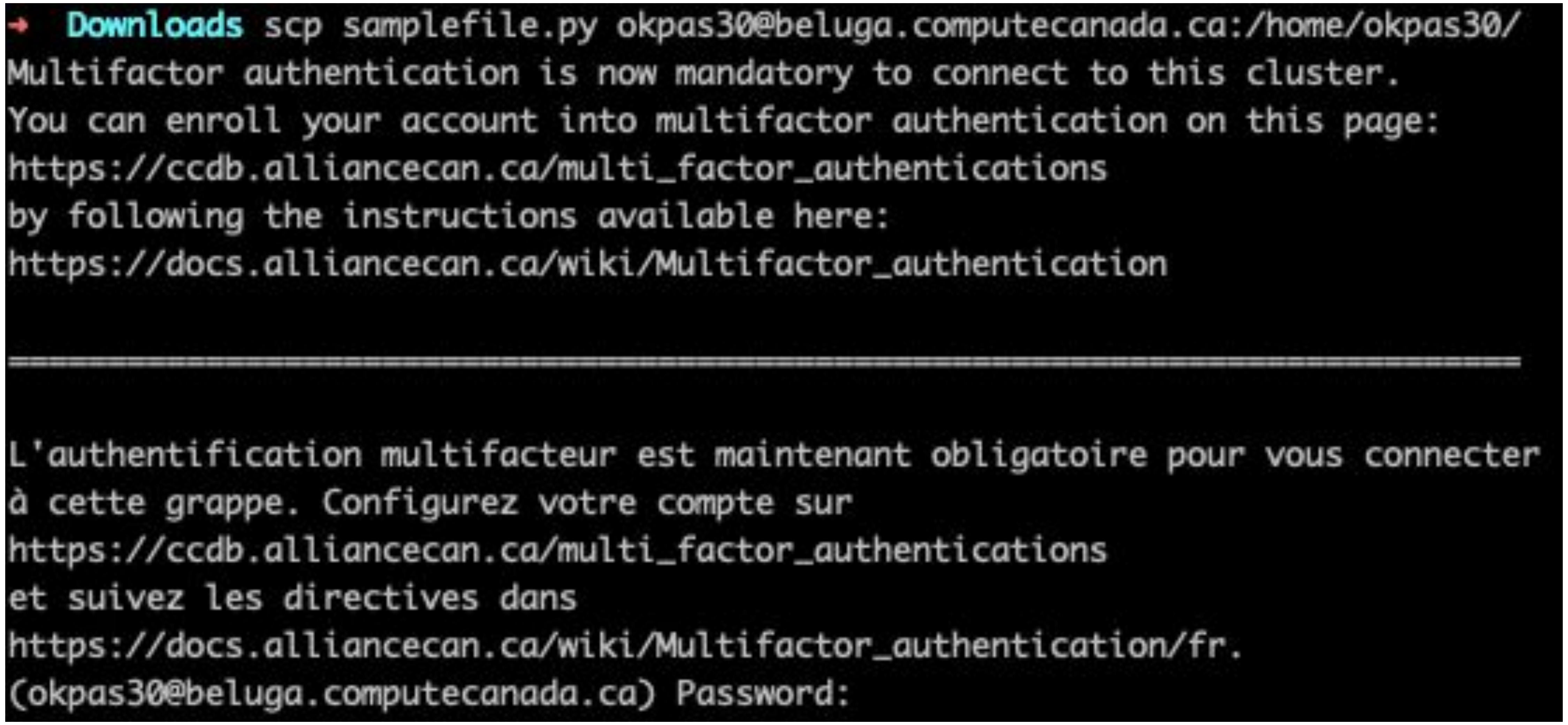
| | |
|---|---|
| Create link for project space | ln -s /arc/project/tr-bootcamp-1 project |
| Create personal directory | mkdir < personal-directory-name> |
| Transfer local file to your directory | scp /local-path/local-file-name user-name@cluster<host>:/file-path |
| Transfer the whole directory | scp-r/local-path/local-directory user-name@cluster<host>:/directory-path |

# Commands

- Upload scripts to the cluster:
  - In your local machine run below command

scp “local file” username@cluster.computecanada.ca:”path of destination”

scp samplefile.py okpas30@beluga.computecanada.ca:/home/okpas30/



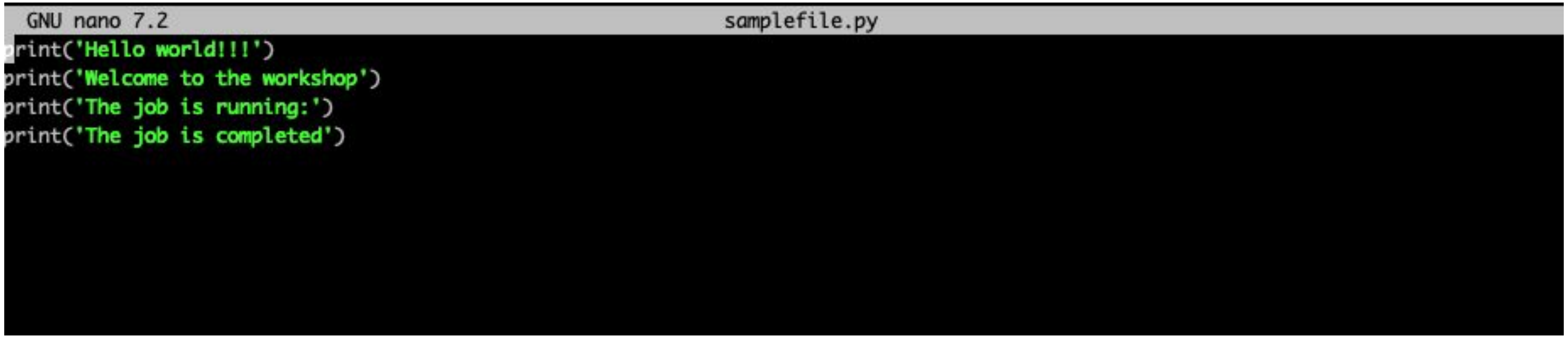
```
➜  Downloads scp samplefile.py okpas30@beluga.computecanada.ca:/home/okpas30/
Multifactor authentication is now mandatory to connect to this cluster.
You can enroll your account into multifactor authentication on this page:
https://ccdb.alliancecan.ca/multi_factor_authentications
by following the instructions available here:
https://docs.alliancecan.ca/wiki/Multifactor_authentication

==============================================================================

L'authentification multifacteur est maintenant obligatoire pour vous connecter
à cette grappe. Configurez votre compte sur
https://ccdb.alliancecan.ca/multi_factor_authentications
et suivez les directives dans
https://docs.alliancecan.ca/wiki/Multifactor_authentication/fr.
(okpas30@beluga.computecanada.ca) Password:
```

# Access Files

- cat, nano, ….

nano filename

nano samplefile.py

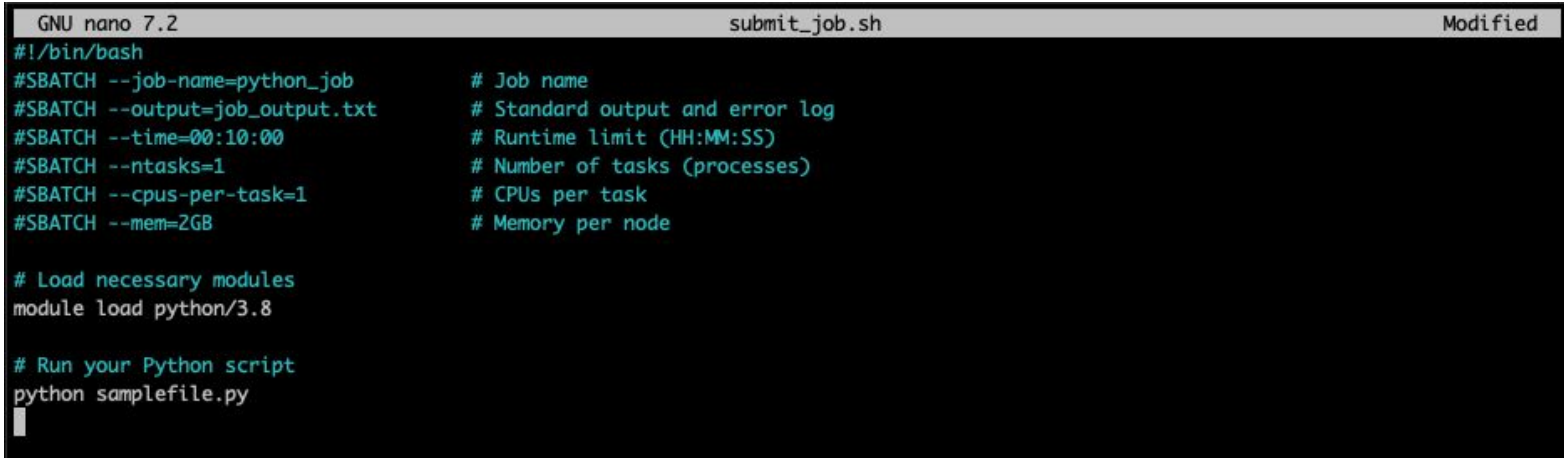


```
  GNU nano 7.2                                            samplefile.py
print('Hello world!!!')
print('Welcome to the workshop')
print('The job is running:')
print('The job is completed')
```

# Create a Job

- Create a new job script

nano submit_job.sh



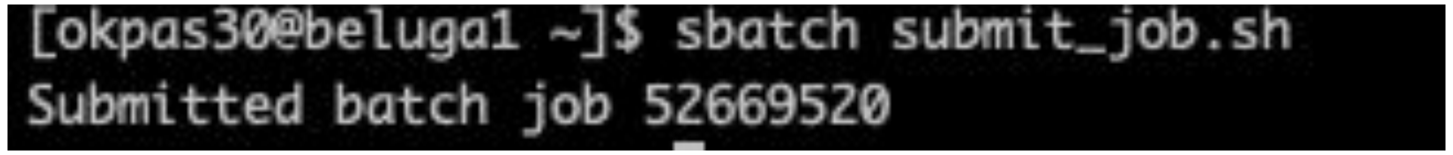
```
#!/bin/bash
#SBATCH --job-name=python_job          # Job name
#SBATCH --output=job_output.txt        # Standard output and error log
#SBATCH --time=00:10:00                # Runtime limit (HH:MM:SS)
#SBATCH --ntasks=1                     # Number of tasks (processes)
#SBATCH --cpus-per-task=1              # CPUs per task
#SBATCH --mem=2GB                      # Memory per node

# Load necessary modules
module load python/3.8

# Run your Python script
python samplefile.py
```

# Submit The Job

- Submit the job script using the sbatch command

sbatch submit_job.sh

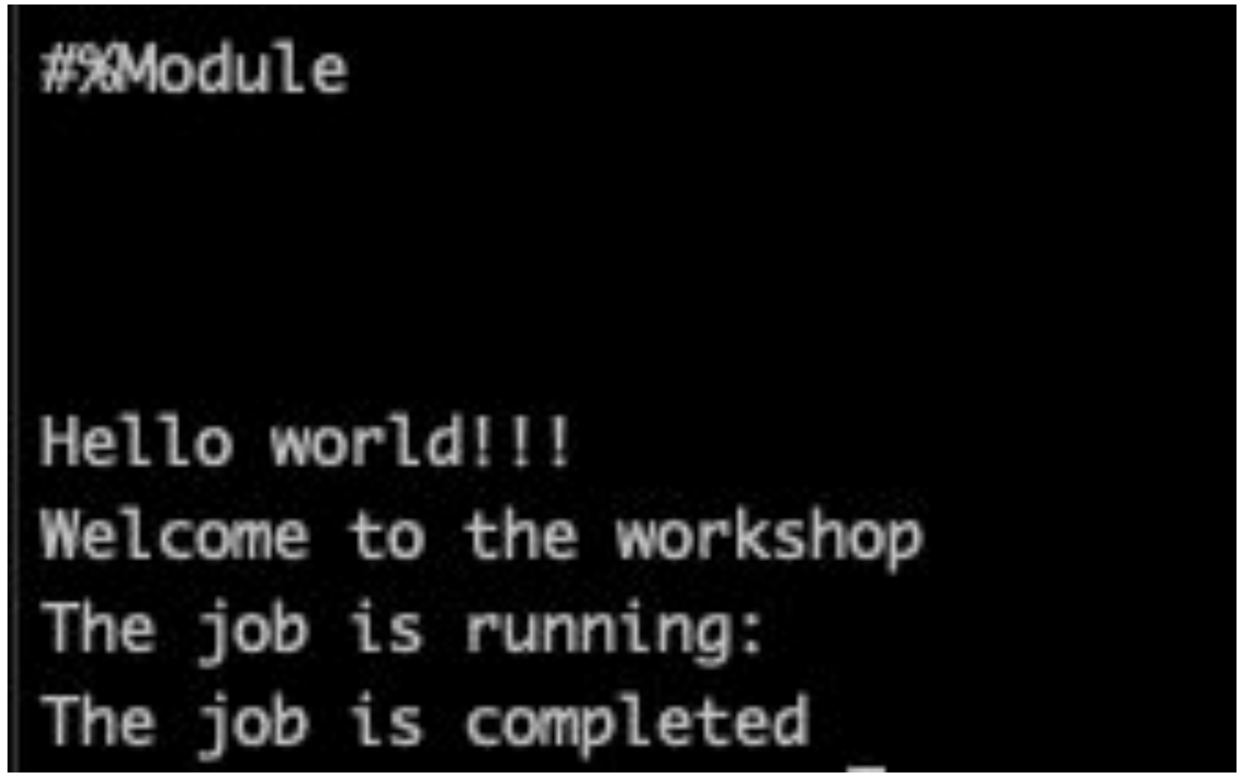
- After submitting, Slurm will provide a job ID



```
[okpas30@beluga1 ~]$ sbatch submit_job.sh
Submitted batch job 52669520
```

- output

cat job_output.txt



```
#%Module



Hello world!!!
Welcome to the workshop
The job is running:
The job is completed
```

# Prepare Environment

- Module
    - Modules are used to load software or tools you need for your project.

      ```
      module load python/3.8
      ```

    - Check loaded module

      ```
      module list
      ```

    - Make a directory (folder)

      ```
      mkdir data_analysis_project
      ```

    - move between directories(cd)

      ```
      cd data_analysis_project
      ```

# Virtual Environment

- Avoid Conflicts:
  - Different projects might require different versions of the same library.
  - Without a virtual environment, installing a new version of a library globally could break existing projects.
  - Example:
  - Project A requires pandas==1.2.0.
  - Project B requires pandas==1.3.0.
  - A virtual environment ensures each project uses its required version.

# Virtual Environment

- Avoid Conflicts
- Keeps Global Environment Clean:
    - Prevents unnecessary clutter in the global Python environment.
    - Avoids polluting the global Python installation with libraries that might only be used for a single project.
- Easier Debugging:
    - Isolated environments reduce the risk of unforeseen bugs caused by mismatched dependencies.

# Virtual Environment

- Avoid Conflicts
- Keeps Global Environment Clean
- Easier Debugging

| | |
|---|---|
| 1. Create a virtual environment: | python -m venv myenv |
| 2. Activate virtual environment | source myenv/bin/activate |
| 3. Closing virtual environment | deactivate |
| 4. remove virtual environment | rm -r myenv |

# Install Requirements

1. Create a virtual environment: python -m venv myenv
2. Activate virtual environment source myenv/bin/activate

- Install libraries:
    - pandas
    - matplotlib
- Freeze and install them at once pip freeze > requirements.txt
- Use pip install -r requirements.txt

# Connect to Jupyter Notebook

- Install jupyter notebook **if not already installed**

```
pip install notebook ipykernel
```

- Add the VE to the jupyter notebook

```
python -m ipykernel install --user --name=myenv--display-name "Python (myenv)"
```

- Restart jupyter notebook

```
pkill jupyter
```

- Start again

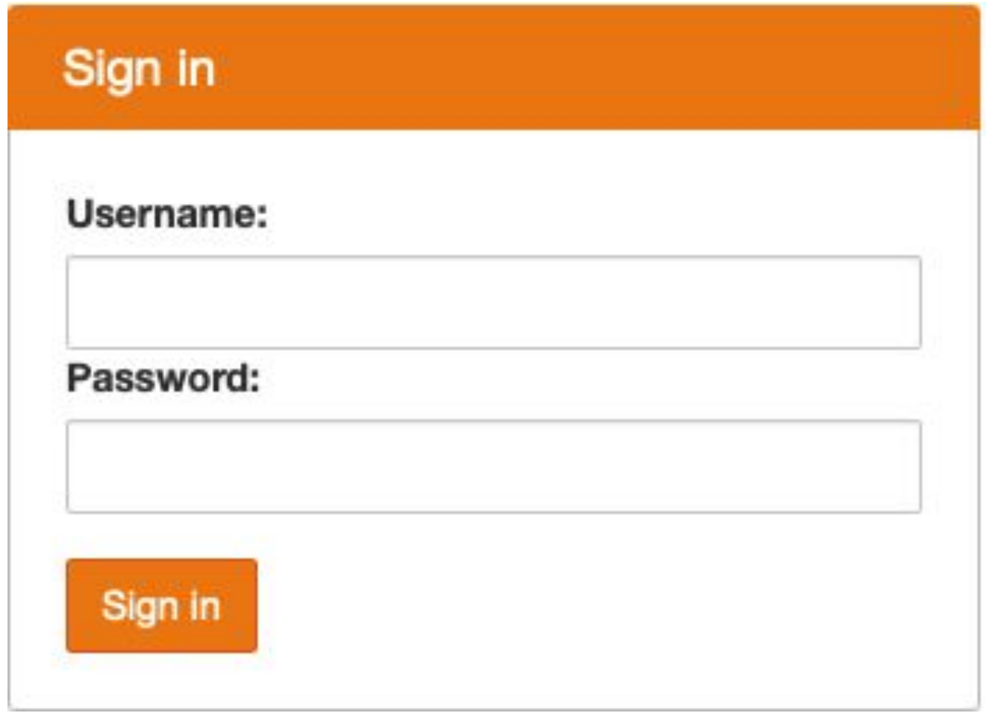
```
jupyter notebook
```

- Open your Jupyter Notebook in the browser. If you're using a web-based Jupyter setup, log back in as needed.
- Choose the kernel tab and change kernel

# Data Analysis

# Connection

- Web
  - In your browser https://jupyter.champions.c3.ca/hub/login



Sign in

Username:

Password:

  - username user01
  - password champions-24

# Libraries

- Python libraries
  - A Python library is a collection of related modules.
  - It makes Python Programming simpler and convenient for the programmer.
- List of Libraries

  https://docs.python.org/3/library/index.html

  https://www.geeksforgeeks.org/libraries-in-python/

# Load Data

- Load data from web and different libraries

```
penguins = sns.load_dataset('penguins')
```

- Upload data using scp into server.

```
scp “local file” username@cluster.computecanada.ca:”path of destination”
```

- Check data
  - head(n):      prints the n first columns of the dataset.
  - penguins.info():   prints information about the columns of the dataset.
  - penguins.isnull().sum():    Check for missing values.

# Handling missing values

- Why?
  - Missing values can lead to biased or invalid results.
  - Many machine learning algorithms do not handle missing data well.
  - incorrect handling of missing data can distort statistical calculations.
- Solution
  - Remove Missing Data: The dataset is large, and the missing data is minimal.
  - Impute Missing Data: Missing values are not random or constitute a significant part of the dataset.
  - Predict Missing Value:  Complex relationships exist between features, and simple imputation isn't sufficient.
  - …

# Analyse

- Scatterplot:
  - Purpose: Visualizes the relationship between two numerical variables.
  - Example: Flipper length vs. body mass, with points colored by penguin species.
  - Insights: Highlights trends, clusters, or correlations between variables.

- Boxplot:
  - Purpose: Shows the distribution of a numerical variable across categories.
  - Example: Bill depth by species.
  - Insights: Identifies medians, spread, and potential outliers.

# Analyse

- Pairplot:
  - Purpose: Provides pairwise scatterplots of all numerical variables.
  - Example: Pairwise relationships of features (e.g., body mass, bill length).
  - Insights: Comprehensive overview of correlations and distributions.

- Heatmap
  - Purpose: Displays the correlation matrix between numerical variables.
  - Example: Correlation matrix of penguin features.
  - Insights: Shows how strongly numerical variables are related.

# Outliers

- Outliers can significantly influence data analysis.
- Detecting and handling them ensures more reliable insights.
- Different methods of outlier detection
  - Here for simplicity we use the same boxplot
- Handling outliers
  - Removing outliers
  - Capping Outliers: Replace with nearest acceptable value.
  - Transform Data (log,etc): Reduce the impact of outliers.

# Questions