Feature Engineering in Aamzon SageMaker

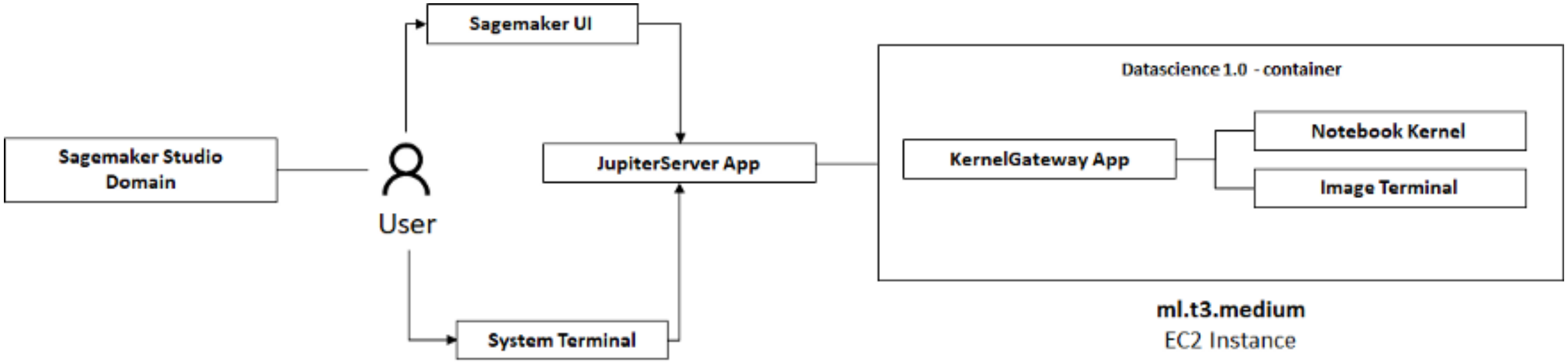
Using SageMaker instance for feature engineering

- Upload the car_prediction_data.csv file in S3
- Open Day 05.ipnyb and read the content to learn how
  - To analyze and visualize the data in Amazon SageMaker notebook instance
  - How to clean data in Amazon SageMaker notebook instance
  - Save cleaned data locally
  - Push the cleaned data into S3 Bucket

SageMaker Studio Data Wrangler

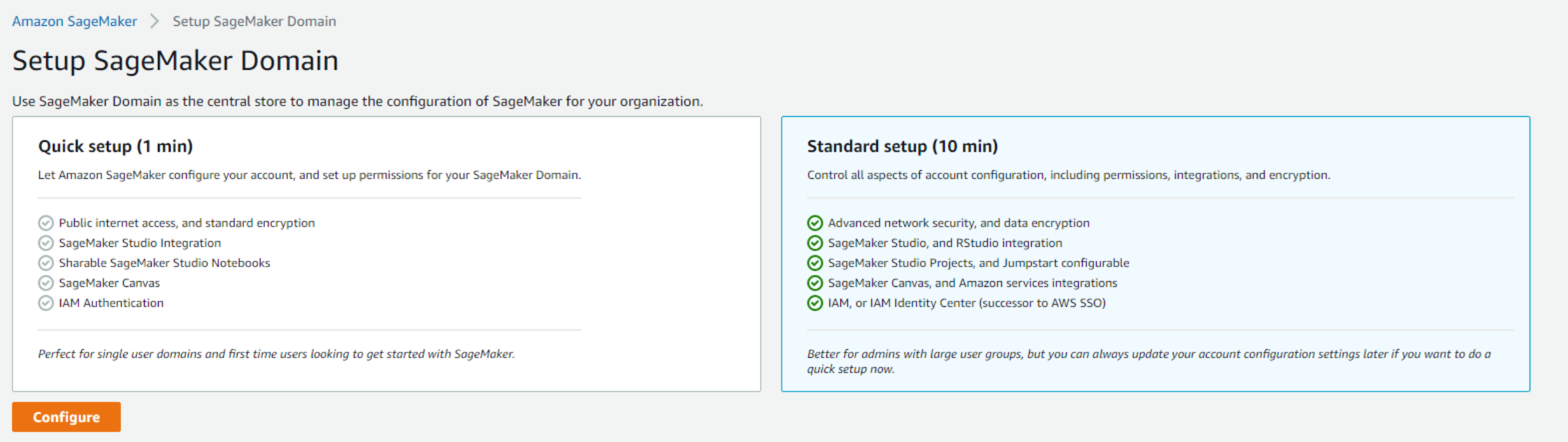
1) Use myapps for this part of lecture
2) Make sure you are in Virginia region

Studio Architecture



Launch SageMaker Studio

- Create SageMaker Domain
- Select Standard Setup



Configure Standard Setup

- Select the domain name and select IAM

Domain name

Name
Domain name should be unique across the AWS account.

day05

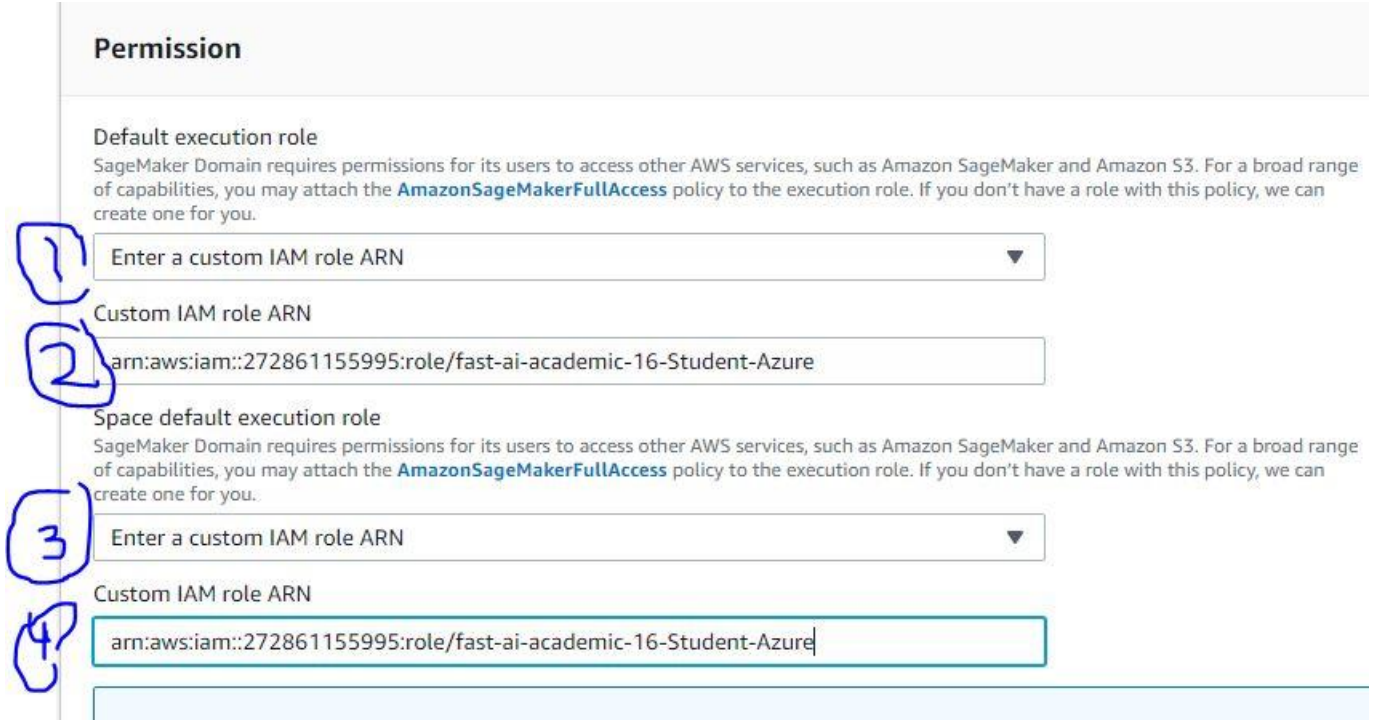
Authentication

The authentication method you choose determines how you can access the SageMaker domain. To use AWS IAM Identity Center (successor to AWS SSO), you must have an IAM Identity Center account in an AWS Region supported by the domain.

- AWS IAM Identity Center
  Access the domain with a bookmarked URL.
- AWS Identity and Access Management (IAM)
  Access the domain with the Amazon SageMaker console.

Permission

See the following picture first and then read the next two slides that has instructions about item 1-4 in the following picture



Permission

- You need to assemble the role as instructed below:

1) Select **custom IAM Role** when you get to the permissions step. You select **custom** in fields 1 and 3 shown in the picture.

2) Assemble the role like this:

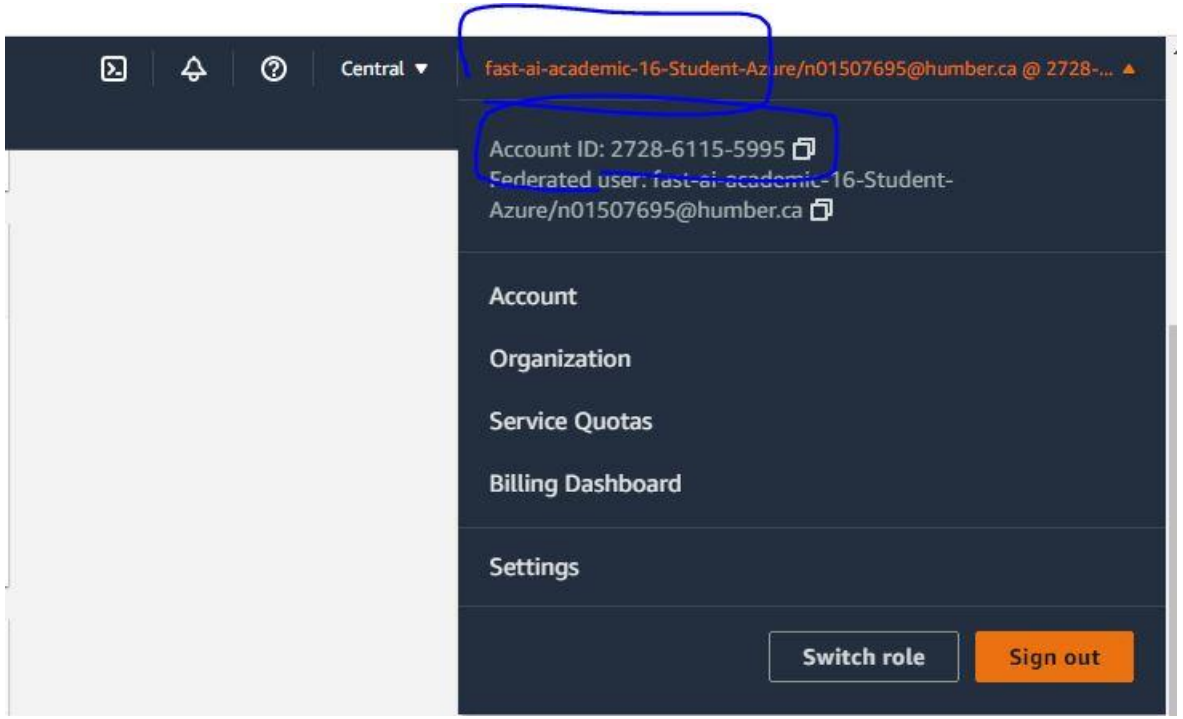
- arn:aws:iam::**AccountName**:role/fast-ai-academic**-nn-**Student-Azure

3) For example, if your AWS account number is 272861155995 and I assigned you the myapps account #16 this is the resulting text:

arn:aws:iam::**272861155995**:role/fast-ai-academic-**16**-Student-Azure

Permission

- You can get the AWS account number by clicking on the top right side, as shown below. Make sure you remove (-) between digits when you assemble the role text I mentioned above.



Network Setting

VPC

To enable internet access, make sure that your VPC has a NAT gateway and your security group allows outbound connections.

vpc-06e729a2a08af20c4 (172.31.0.0/16) | aws-controltower-VPC

Subnet

Choose a subnet in an availability zone supported by Amazon SageMaker.

Choose one or more subnets

subnet-063e600d62c572f34 (172.31.64.0/20) | ca-central-1a aws-controltower-PrivateSubnet1A

subnet-05e6b3d2248c207de (172.31.16.0/20) | ca-central-1b aws-controltower-PrivateSubnet2B

Security group(s)

These security groups will also be associated with the RStudioServerPro App.

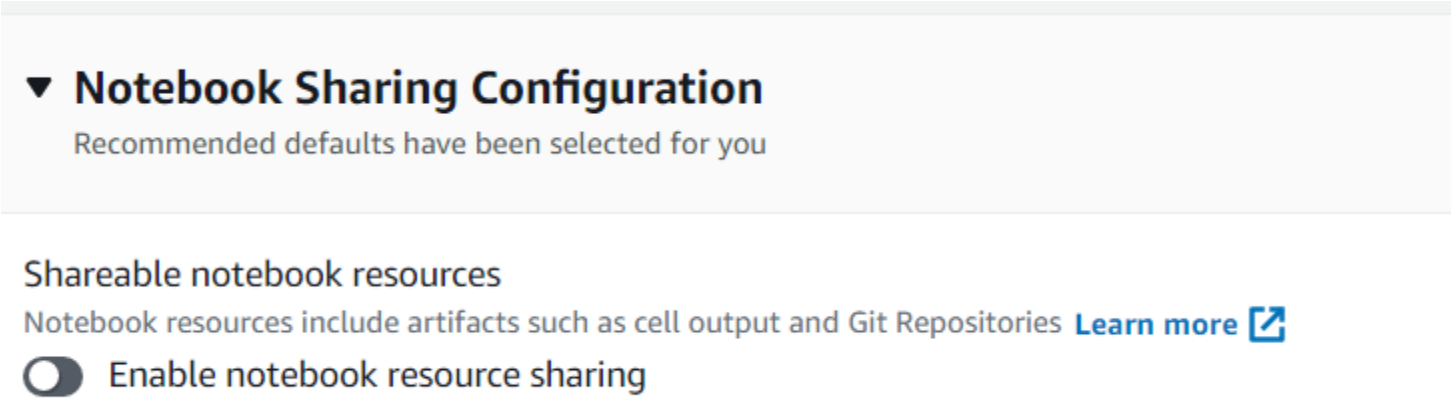
Choose one or more security groups

sg-0aff74529d6b71f04 (default)

- (●) Public Internet Only - The SageMaker domain will use default SageMaker internet access. Your vpc is used only for accessing the attached EFS storage
- ( ) VPC Only - The SageMaker domain will use your VPC. Direct internet access is disabled.
  To enable internet access, make sure that your VPC has a NAT gateway and your security group allows outbound connections.

We do not need to share the notebooks

- This for scenarios that more than one person wants to work on the same notebook



All Jumpstart features must be enabled

SageMaker Projects and JumpStart - *optional*

SageMaker Projects and JumpStart New

Enable access and provisioning of AWS Service Catalog Portfolio of products in Amazon SageMaker Studio for Amazon SageMaker Projects and JumpStart. **Learn more**

Enable Amazon SageMaker project templates and Amazon SageMaker JumpStart for this account

If enabled, the administrator can view the Amazon SageMaker built-in project templates and Amazon SageMaker JumpStart solutions published in AWS Service Catalog. A launch constraint role and a project use role are automatically generated in IAM for your account.

Enable Amazon SageMaker project templates and Amazon SageMaker JumpStart for Studio users

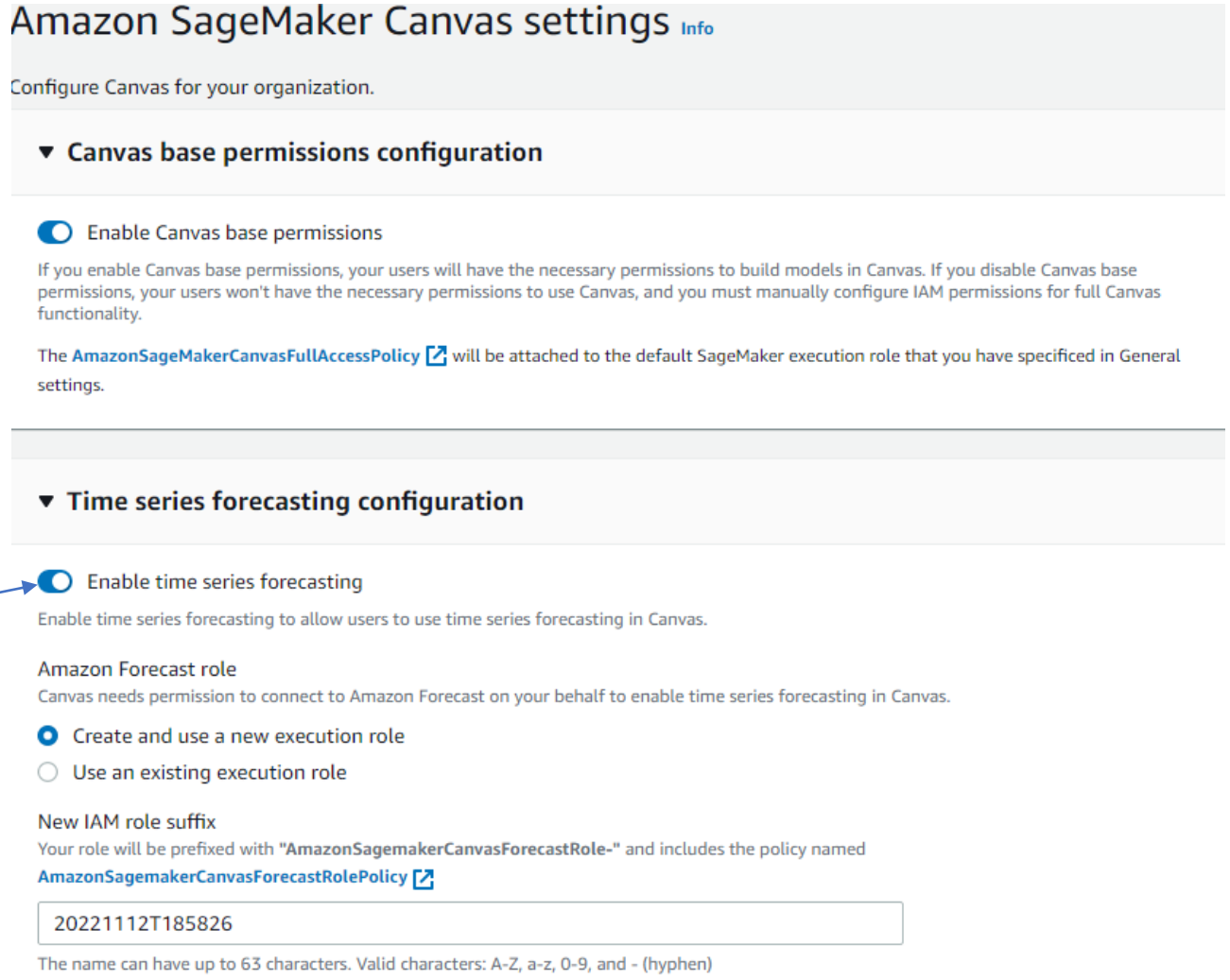
If enabled, this setting allows users who are currently using the domain execution role to create projects using templates and JumpStart solutions published by Amazon SageMaker in AWS Service Catalog. If there are individual users using custom execution roles in your organization, you need to enable them on the user profile page.

After clicking on Next, and Submit

Canvas Settings

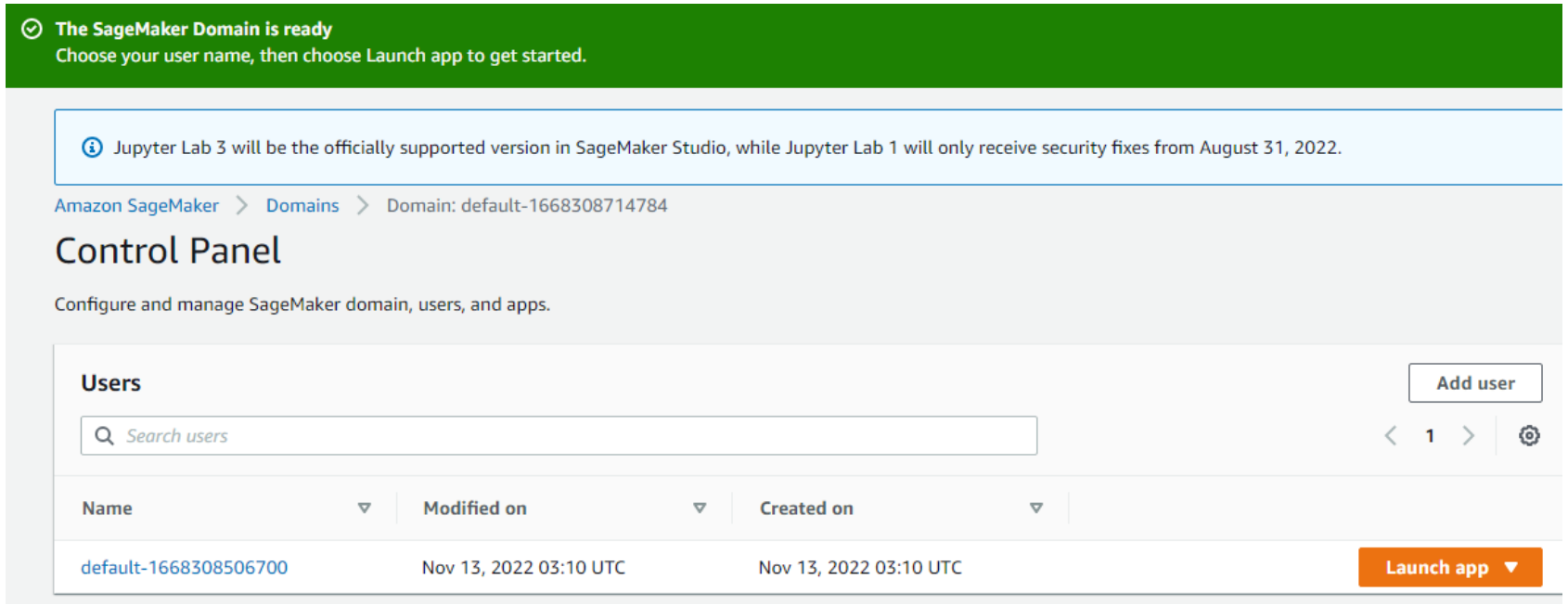
- Ignore R Studio error and continue to Canvas configuration

Disable time series forecast

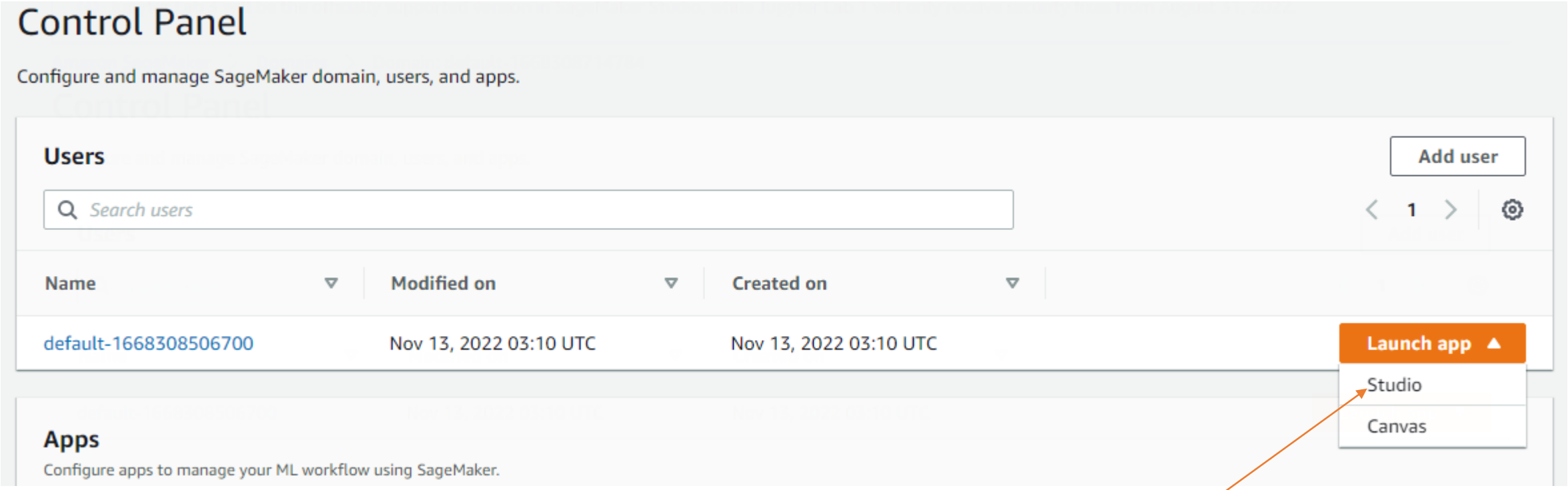


Before clicking on Create

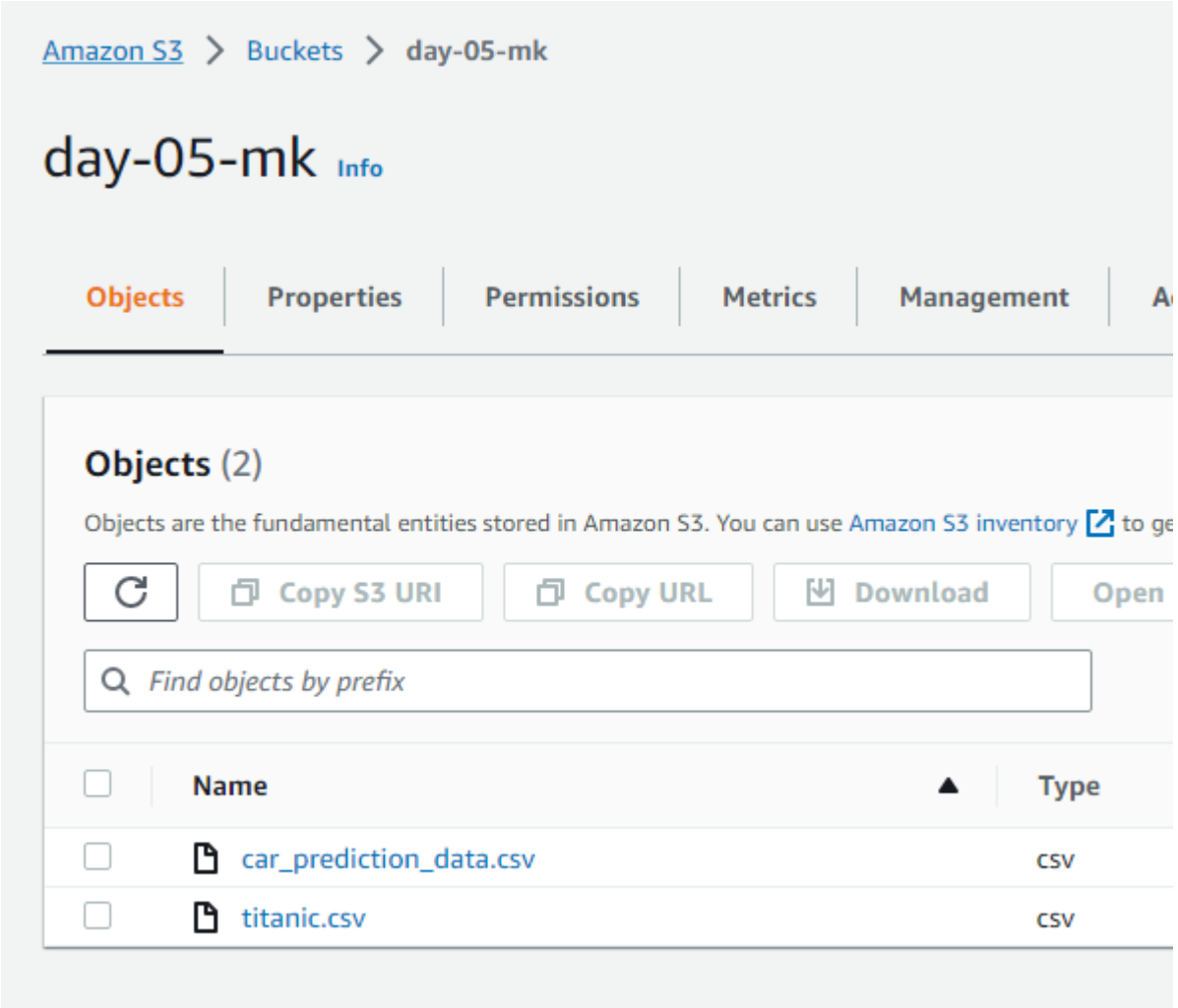
- After creating domain, make sure the **default** user is there or create it if it is not there



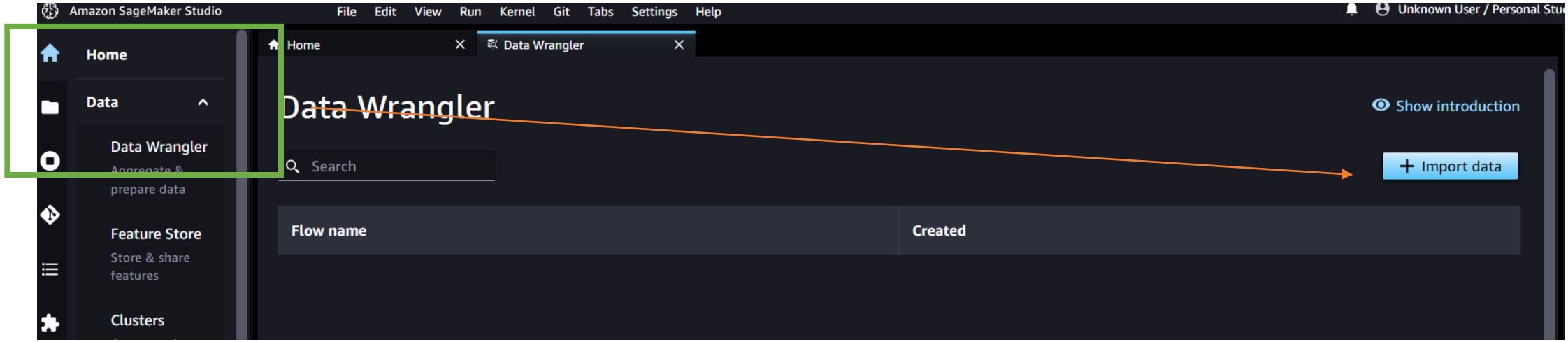
Launch the Studio



Upload Titanic dataset to S3

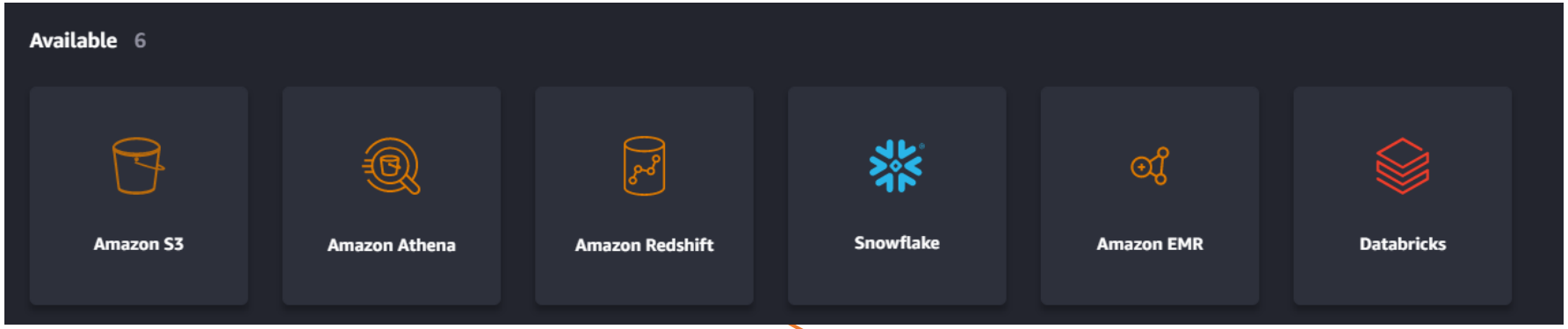


Start DW flow and import titanic data

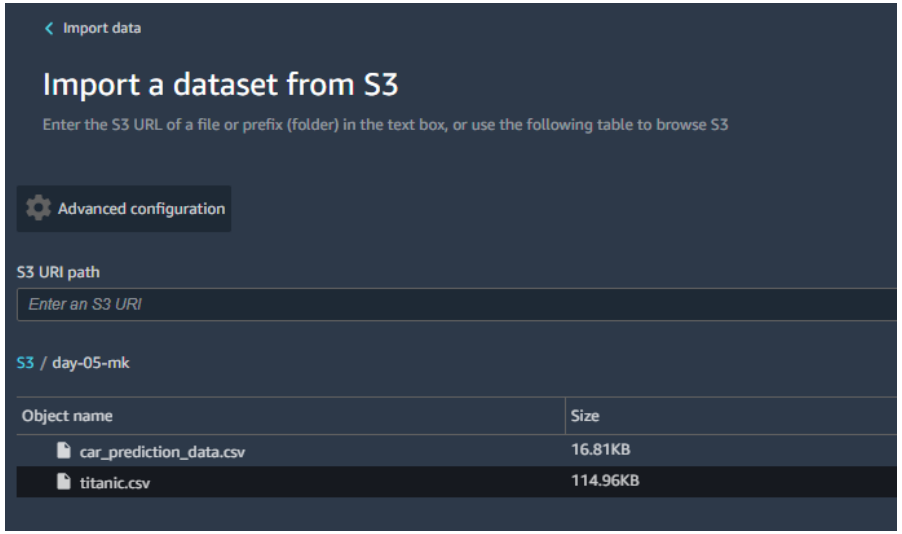


NOTE: when you click on Import, The studio creates a new server for DW. If you do not terminate the server after you are done with DW, your credit will be used up. Make sure you terminate the DW server. I show you how to do that.

Select the bucket and data set

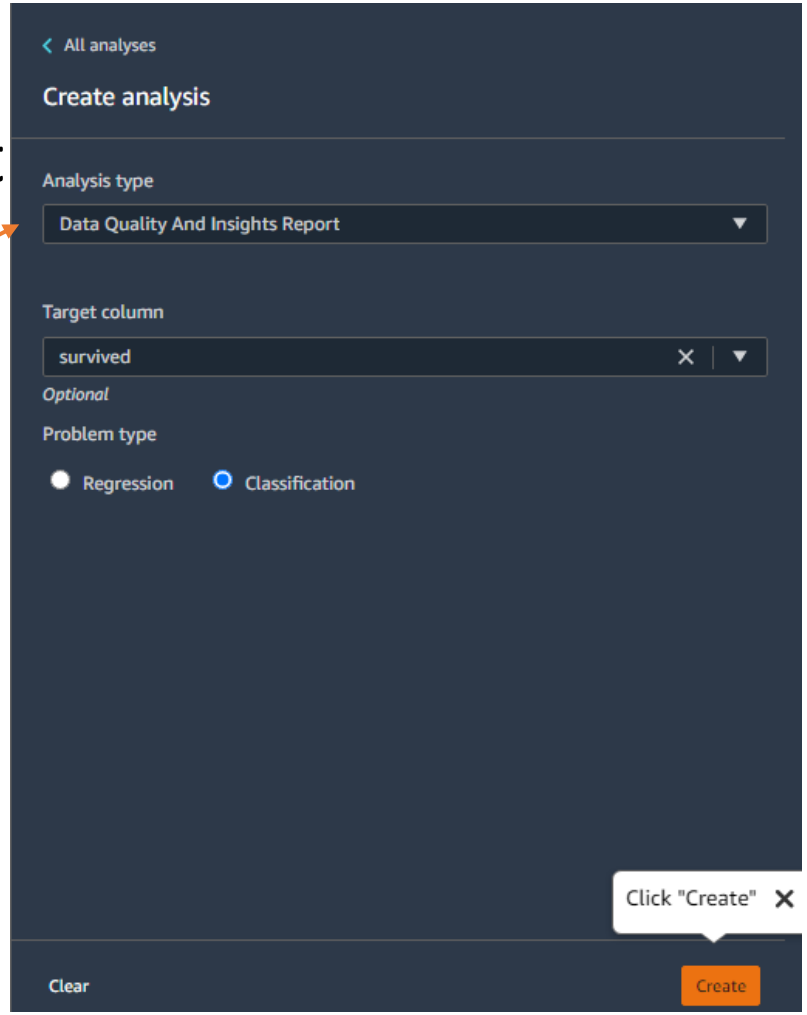


1. PassengerId: Unique Id of a passenger
2. Survived: If the passenger survived(0-No, 1-Yes)
3. Pclass: Passenger Class (1 = 1st, 2 = 2nd, 3 = 3rd)
4. Name: Name of the passenger
5. Sex: Male/Female
6. Age: Passenger age in years
7. SibSp: No of siblings/spouses aboard
8. Parch: No of parents/children aboard
9. Ticket: Ticket Number
10. Fare: Passenger Fare
11. Cabin: Cabin number
12. Embarked: Port of Embarkation (C = Cherbourg; Q = Queenstown; S = Southampton)



Generating a quick report

- Click on the **+** sign and click on **Get data insight**
- Select **Data Quality** report and **Survived** in Target column
- Select **Classification** in prediction type



Data quality and insights report

- A report to get information that might help you with data exploration and feature engineering
- It gives you information such as the number of missing values and the number of outliers.
- If you have issues with your data, such as **target leakage** or i**mbalance**, the insights report can bring those issues to your attention.

Observe the following reports

- SUMMARY → MISSING VALUE
- DUPLICATE ROWS
- QUICK MODEL
- CONFUSION Matrix
- Prediction Power

SUMMARY

Dataset statistics

Key	Value
Number of features	14
Number of rows	1309
Missing	1.55%
Valid	96.4%
Duplicate rows	0%

Feature type	Count
numeric	6
categorical	2
text	4
datetime	0
binary	1
unknown	0

Data Exploration

- Choose the **+** next to the **Data type** step in your data flow and select **Add analysis**
- In the **Analysis** area, select **Table summary** from the dropdown list.
- Give the table summary a **Name**.
- Select **Preview** to preview the table that will be created.
- Choose **Save** to save it to your data flow. It appears under **All Analyses**.

Observations

- Fare average (mean) is around $33, while the max is over $500. This column likely has outliers.
- This dataset uses *?* to indicate missing values. A number of columns have missing values: *cabin*, *embarked*, and *home.dest*

cabin	embarked
1309	1309
None	None
None	None
?	?
T	S

- The age category is missing over 250 values (different between 1309 and 1046).
- Go back to the data flow. Next, clean your data using the insights gained from these stats.

Drop Unused Columns

- Choose **+** next to the **Data type** step in your data flow and choose **Add transform**.
- Choose **Manage columns**
- Under **Transform**, make sure **Drop column** is selected
- Under **Columns to drop**, specify the following column names: cabin, ticket, name, sibsp, parch, home.dest, boat, body
- Choose **Preview**
- Choose **Add**

Using Pandas to drop the columns

- Alternatively, you could drop the columns by using the following code as well

```
cols = ['name', 'ticket', 'cabin', 'sibsp', 'parch', 'home.dest','boat', 'body']
df = df.drop(cols, axis=1)
```

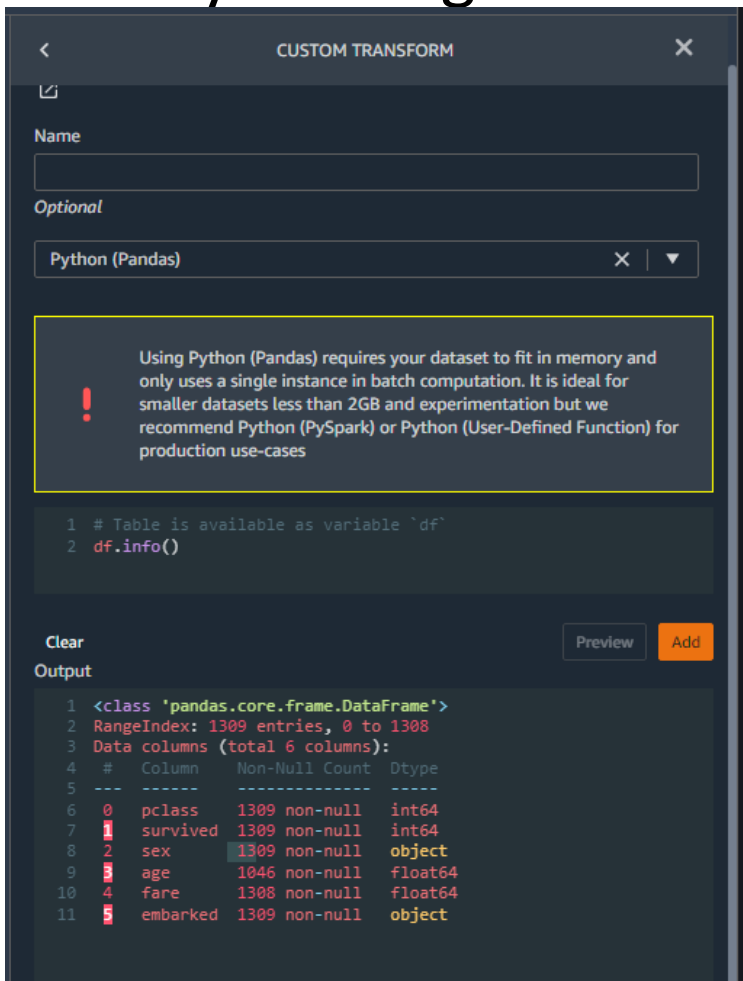
Clean up Missing Values

- Before we start fixing the missing values let's see how many missing value we have. Run df.info() as shown.
- You **do not** need to add it to the flow, that is

Just for you to get a better idea

As you see **Age** and **Fare** have different counts

```
 0   pclass    1309 non-null   int64
 1   survived  1309 non-null   int64
 2   sex       1309 non-null   object
 3   age       1046 non-null   float64
 4   fare      1308 non-null   float64
 5   embarked  1309 non-null   object
dtypes: float64(2), int64(2), object(2)
memory usage: 61.5+ KB
```



Handing missing value

- Click on + after Drop Column → Data Transform
- Choose **Handling missing**.
- Choose **Drop missing** for the **Transformer**.
- ~~Choose **Drop Rows** for the **Dimension**.~~
- Choose *age* for the **Input column**.
- Choose **Preview** to see the new data frame, and then choose **Add** to add the transform to your flow.
- Repeat the same process for *fare*.

Post report

- Use df.info() again to see the result
- All the counts are the same now

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1045 entries, 0 to 1044
Data columns (total 6 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   pclass    1045 non-null   int64
 1   survived  1045 non-null   int64
 2   sex       1045 non-null   object
 3   age       1045 non-null   int64
 4   fare      1045 non-null   float64
 5   embarked  1045 non-null   object
dtypes: float64(1), int64(3), object(2)
memory usage: 49.1+ KB
```

CUSTOM TRANSFORM

Use Pyspark, Pandas, or Pyspark (SQL) to define custom transformations. Learn more.

Name

Optional

Python (Pandas)

Using Python (Pandas) requires your dataset to fit in memory and only uses a single instance in batch computation. It is ideal for smaller datasets less than 2GB and experimentation but we recommend Python (PySpark) or Python (User-Defined Function) for production use-cases

```
# Table is available as variable `df`
df.info()
```

Clear

Preview

Add

Output

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1045 entries, 0 to 1044
Data columns (total 6 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   pclass    1045 non-null   int64
 1   survived  1045 non-null   int64
 2   sex       1045 non-null   object
 3   age       1045 non-null   int64
 4   fare      1045 non-null   float64
 5   embarked  1045 non-null   object
```

One Hot Encoding

- **Custom Pandas→ Name: Encode**
- In the **Custom Transform** section, choose **Python (Pandas)** from the dropdown list and add the code
- Choose **Preview** and **Add**

```
import pandas as pd

dummies = []
cols = ['pclass','sex','embarked']
for col in cols:
    dummies.append(pd.get_dummies(df[col]))

encoded = pd.concat(dummies, axis=1)

df = pd.concat((df, encoded),axis=1)
```

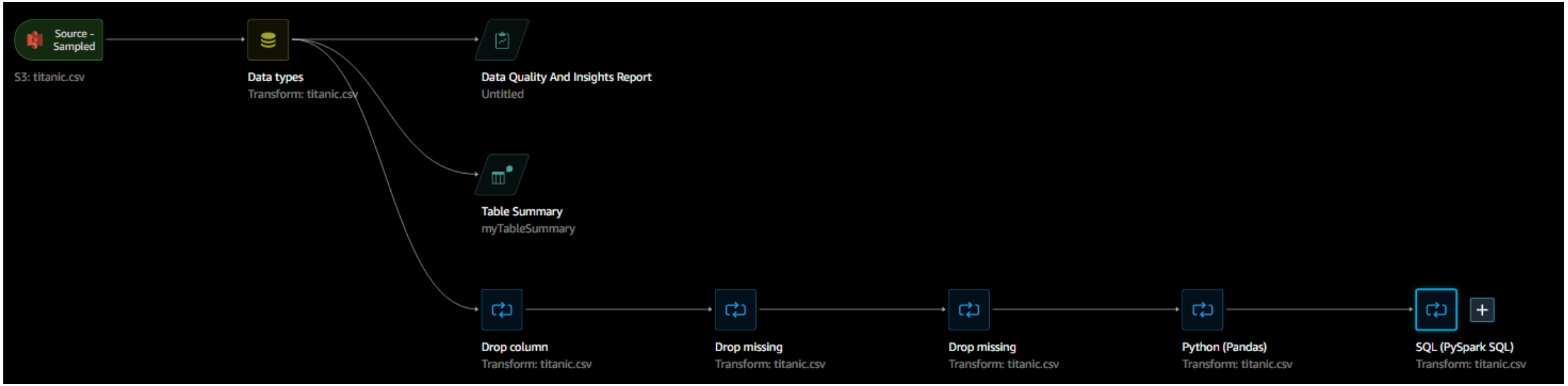
Custom SQL: SELECT Columns

- You can select the columns you want to keep using SQL.
- In the **Custom Transform** section, select **SQL (PySpark SQL)** from the dropdown list
- Enter the following in the code box.

```
SELECT survived, age, fare, 1, 2, 3, female, male, C, Q, S FROM df;
```

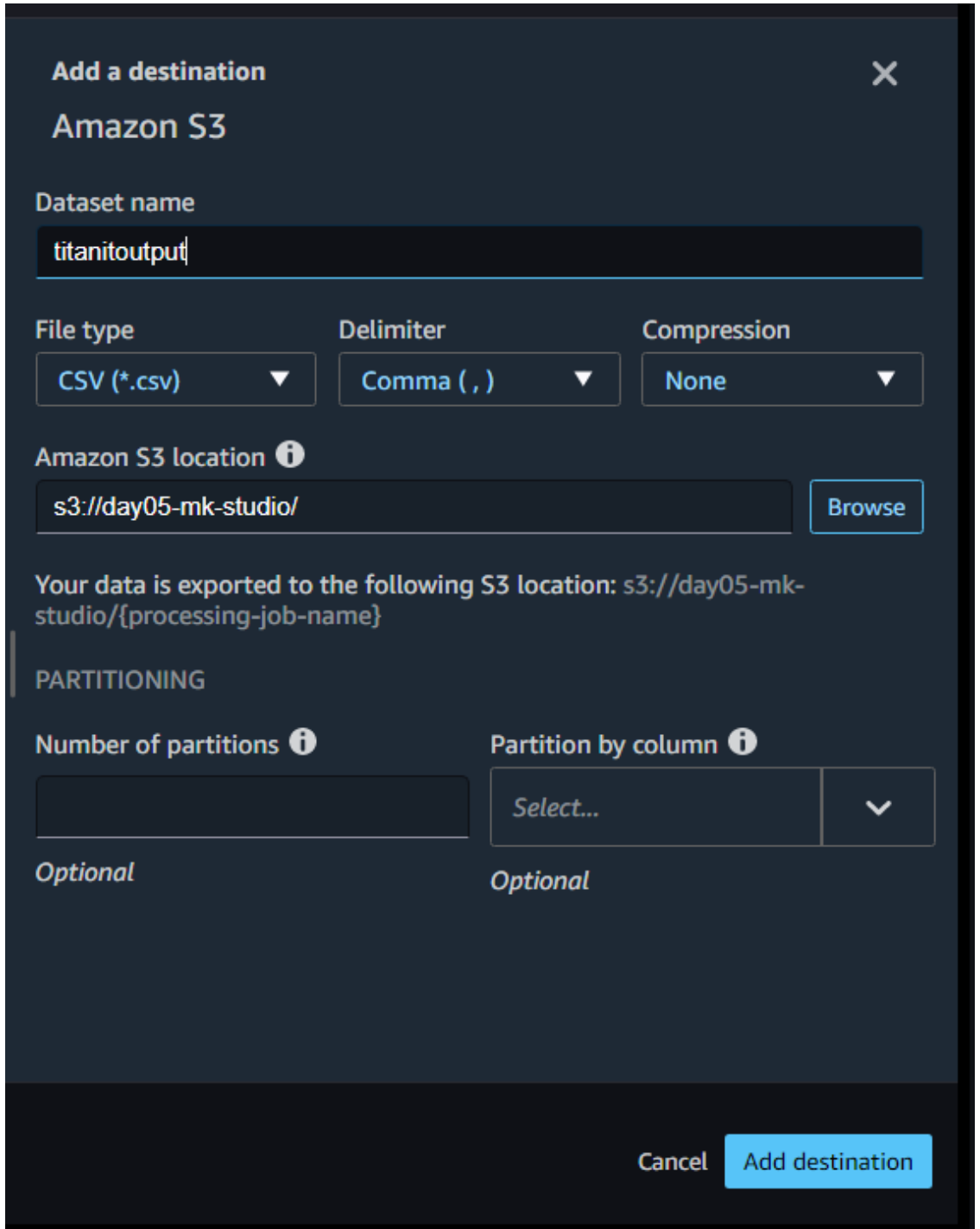
- Choose **Preview** and then **Add**
- **The columns listed in your SELECT statement** are the only remaining **columns**

This is what you should have now



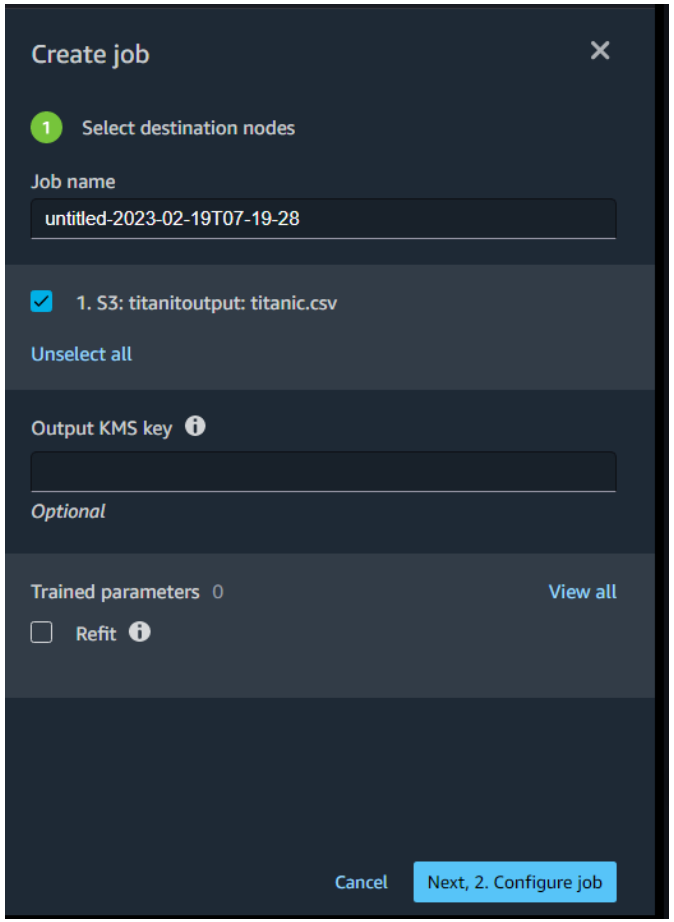
Add a destination

- You can let the data flow to be Executed and the result to be saved into S3 bucket



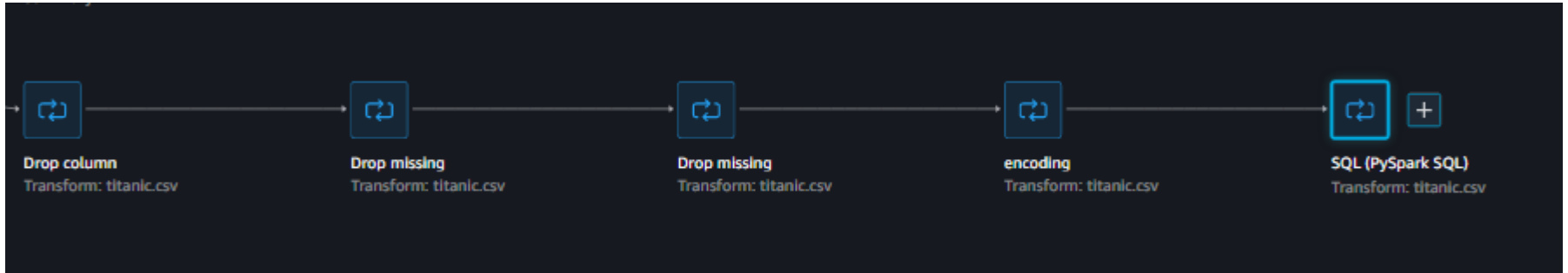
After you create a flow with S3 as destination

- You need to create a **job** to run that flow now or schedules



Export to a Data Wrangler Notebook

- When you export your data flow using a **Data Wrangler job**, the process automatically **creates** a Jupyter Notebook
- This notebook automatically opens in your Studio instance and is configured to run a SageMaker processing job to run your Data Wrangler data flow, which is referred to as a Data Wrangler job.
- Save your data flow→ Select **File** and then select **Save Data Wrangler Flow**.
- Click on + in the last step and Select **Amazon S3 (via Jupyter notebook)**



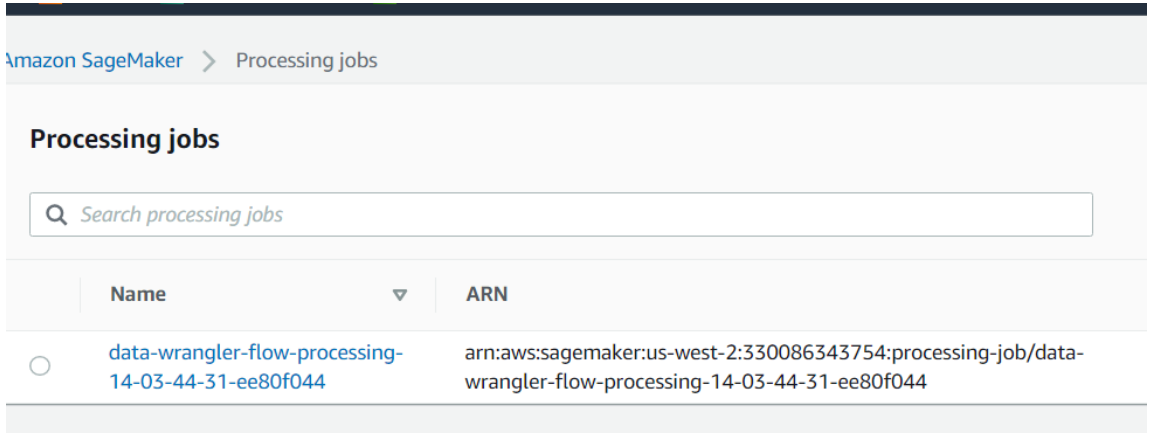
Using the notebook

- Choose any **Python 3 (Data Science)** kernel for the **Kernel**.
- Before start running the cells read the following instructions:
  - Do not run optional parts related to training **but** run the **“ (Optional) Configure Spark Cluster Driver Memory**”
  - **Do not start training**
  - Change the output bucket to your own bucket where you downloaded the titanic data

```
# You can configure this with your own bucket name, e.g.
bucket = "day-05-mk"
```

See the results

- Follow the execution flow in SageMaker console processing job and in the S3



Clean up

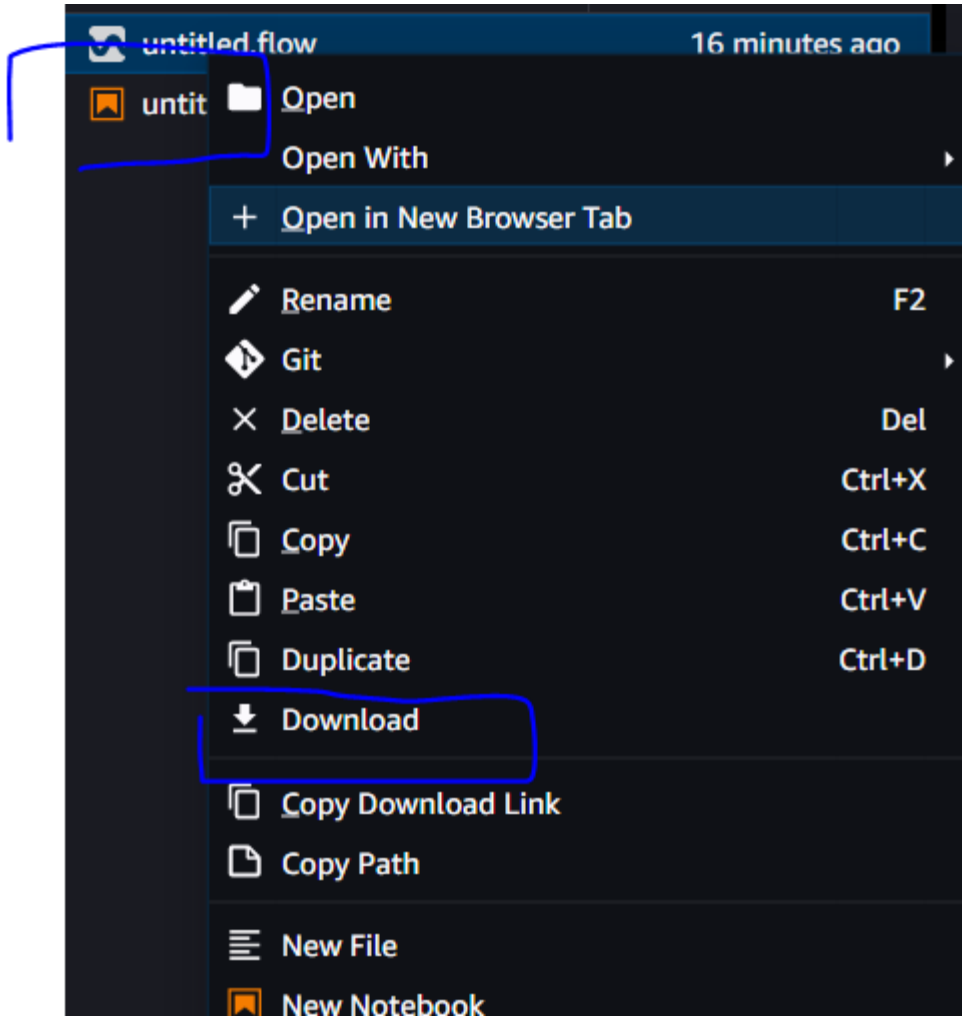
- Very important: Make sure you terminate the DW server otherwise it will use up your credit.

Assignment

- Go to https://www.openml.org/
- Select a data set
- Analyze the data in DW and find some opportunities to improve the quality of data (like removing missing value or scaling data or one hot encoding, etc)
- Clean/transform data in Studio DW
- Upload the following items in BB
  - Your selected data set
  - DW .flow file (to learn how to download the .flow file see next slide)
  - A report that has covers at least the suggested ToC (please see next slides)

Download the .flow file

- Right click on the file
- Select Download



ToC for the report

- Data set fields descriptions
- The visualization and analysis that you have done in DW and what you learn out of those visualizations. Each analysis comes with a picture and your description beneath
- The transformations you have done, again you have to explain why you have chosen that transformation, include the picture and explain the results after transformation (with pictures)
- If you have added a code, include that in the report in the right spot