Prerequisites :

* Azure Subscription
* Anaconda3 installed in your local machine
* Mlflow installed in your anaconda3 packages list
* Your own custom model (carsmlr)

Before Signing up into Azure Portal first we should convert our custom model into Mlflow model

We already have converted code ready and the code file name is cars1.ipynb

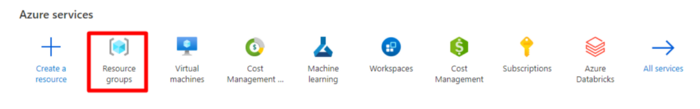
Note : running the code locally to check how the code is running is optional . My suggestion would be you can directly go to the azure ml studio and import your mlflow code under workspace is pretty simple to create endpoint

Now follow the below steps ……..

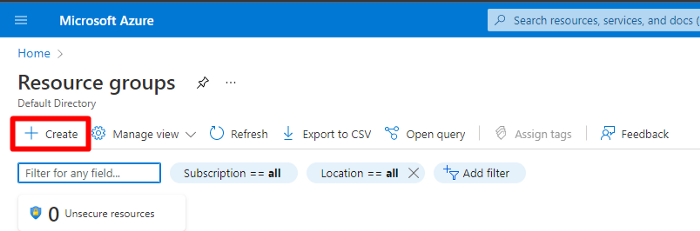
**1. Create a resource group**

A resource group is a container that holds related resources for an Azure solution

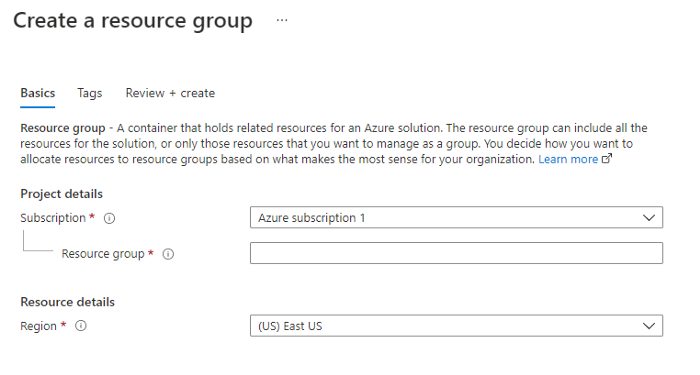
* Go to <https://portal.azure.com/>
* Find “Resource groups” under Azure Services or through the search bar



* Create a new resource group

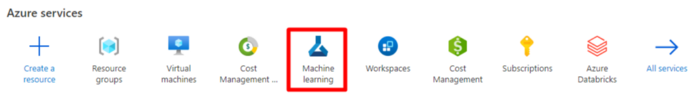


* Fill in the details such as subscription, name of resource group and the region

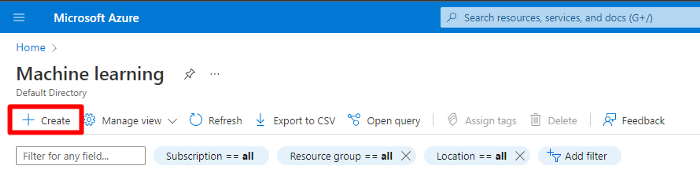


**2. Create Azure ML Workspace**

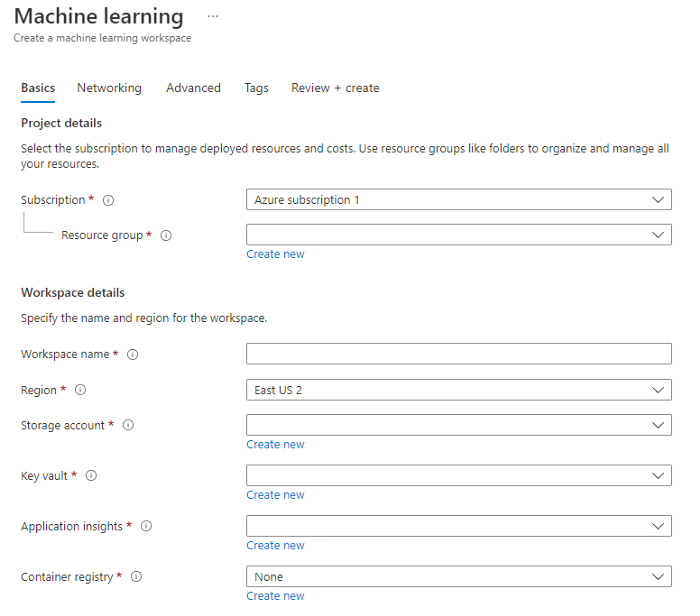
* Find “Machine Learning” under Azure Services or through the search bar.



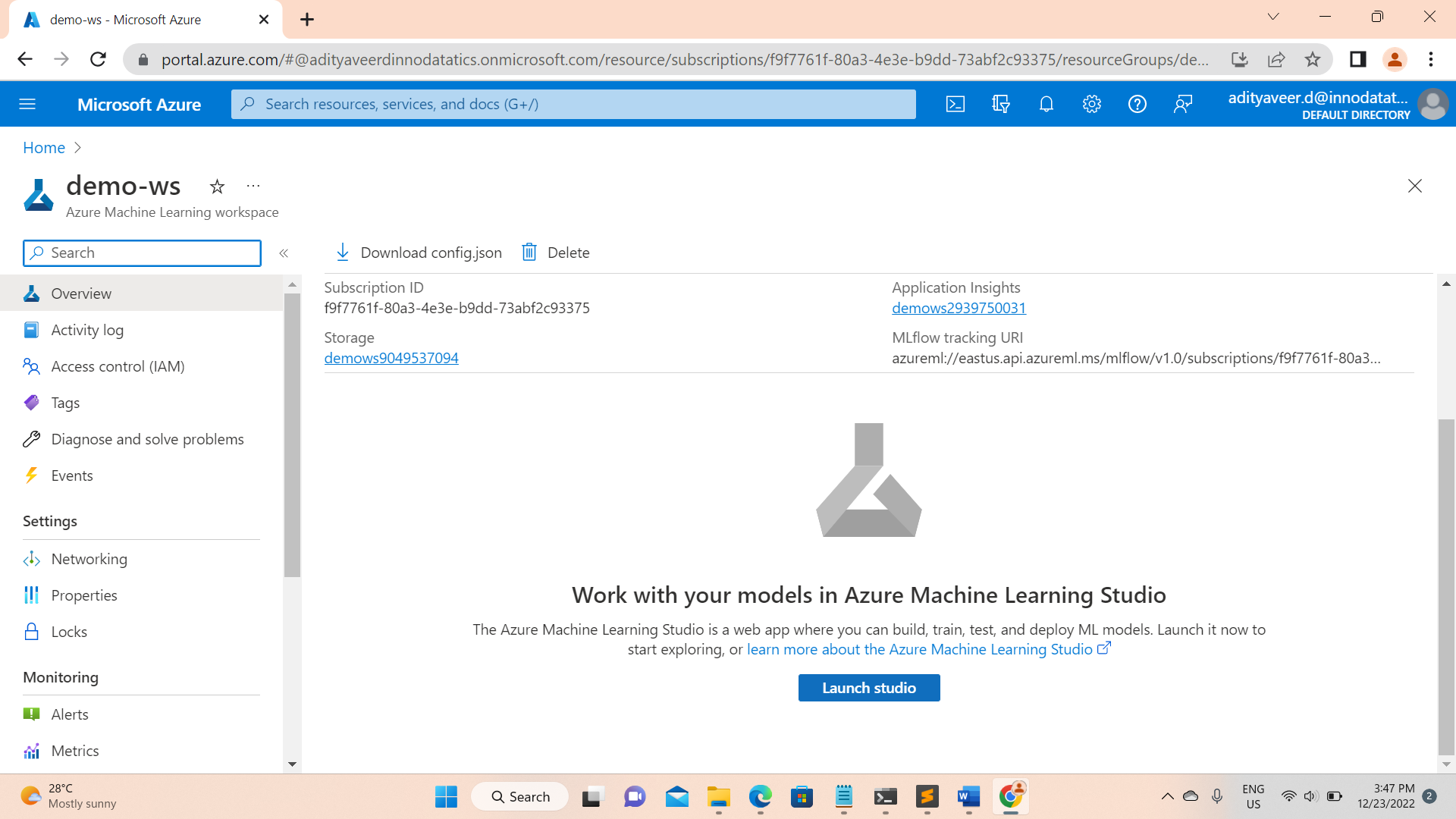
* Click on create



* Fill in the blanks. Resource group is the one which we created in the previous step.



* MLFlow tracking server is automatically created as part of the Azure ML workspace
* Launch the Azure Machine Learning studio



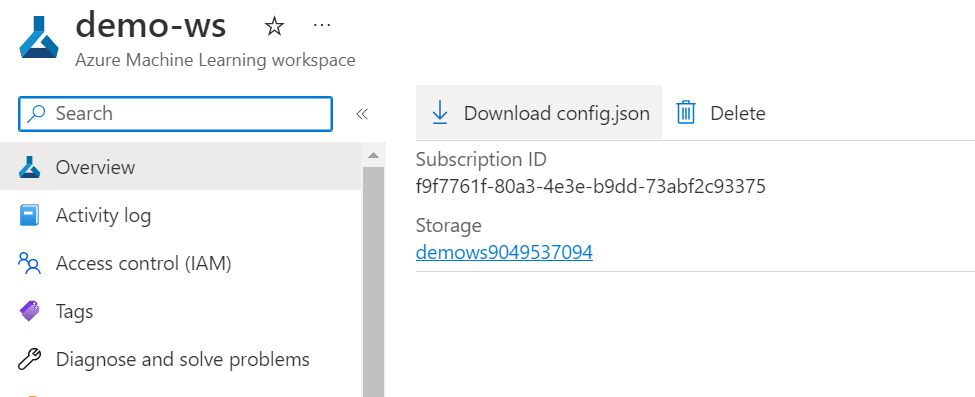
So far we good ….

Just adding a note here ….

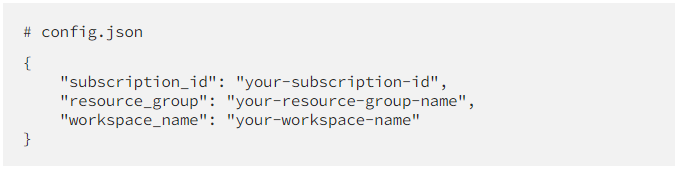
In section 3 below I have mentioned how to download workspace configs but download config file is only required for complete local deployment (meaning python sdk ) but our case we are using Azure ML studio . So no need to download config file .we can skip this step

3. Azure Machine learning Workspace Configs

* Download the Azure Machine Learning workspace configurations.



The config file is in JSON format and it contains the following information:

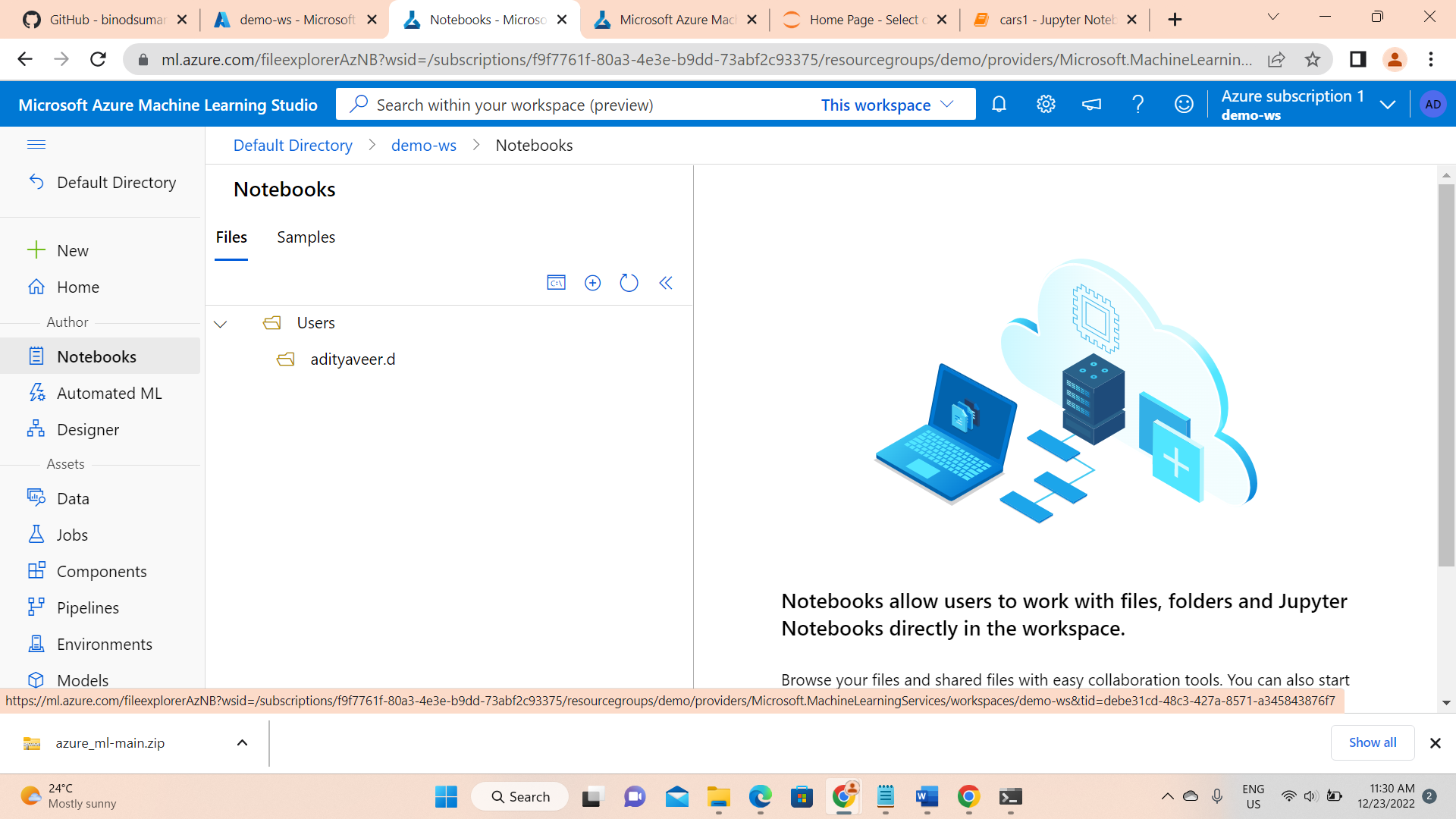


We will need these information to connect to AML workspace for logging of experiments.

Very Important step…..

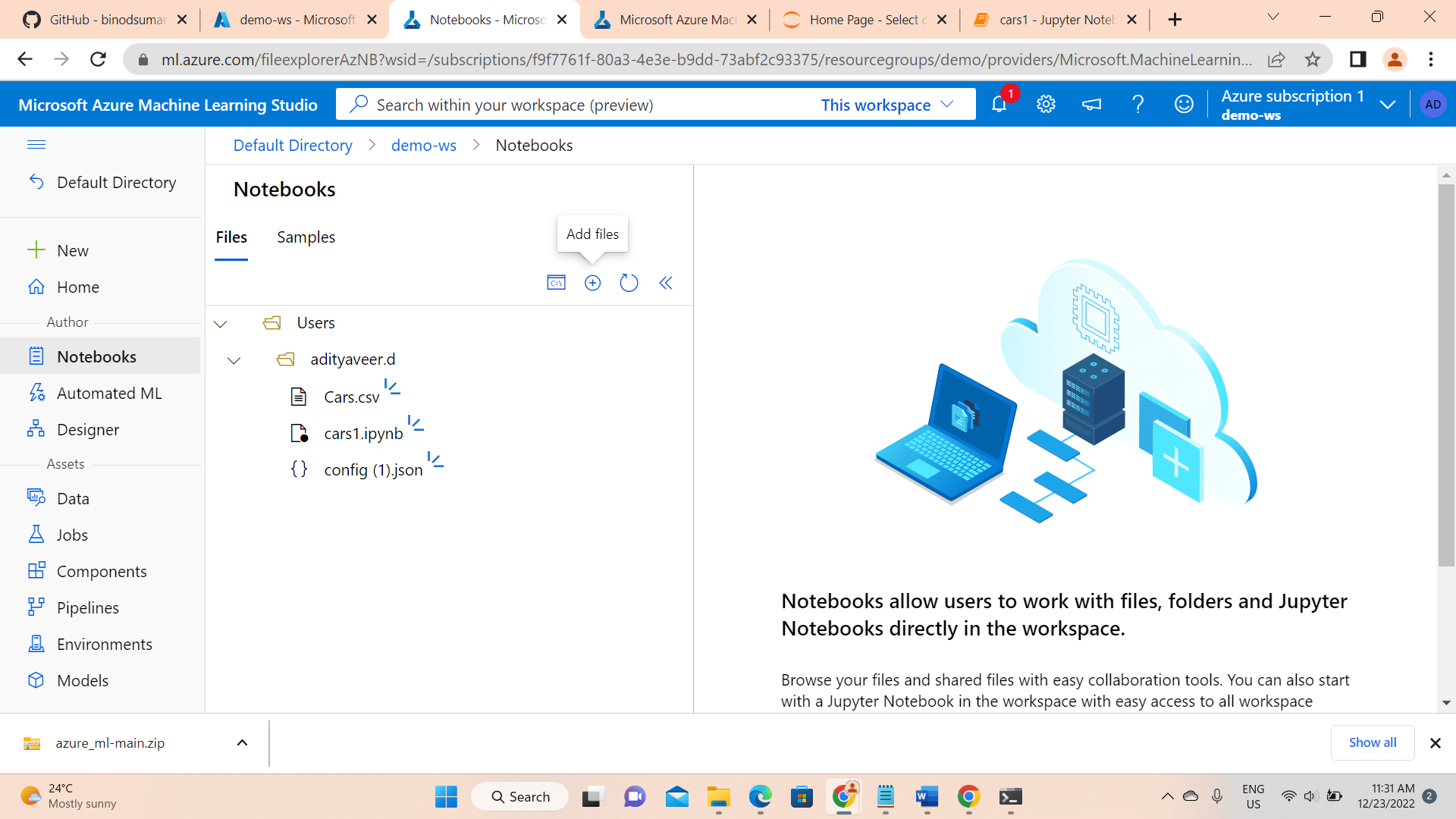
4. Now setup training environment

* Go to the notebooks section



* Upload your MlFlow Inference code and csv file, config file

Note : config file is not mandatory as I said before



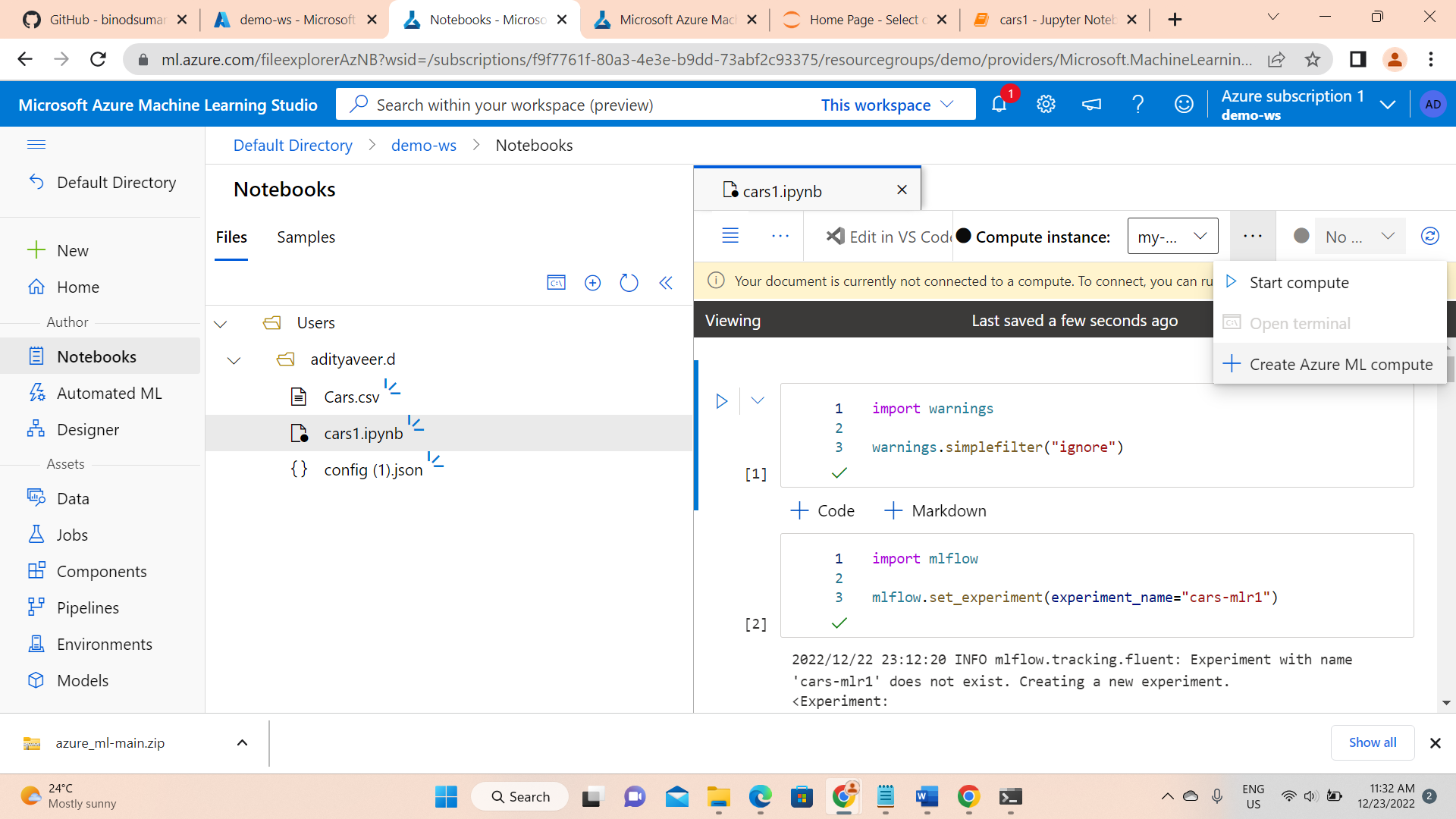
* create compute instance

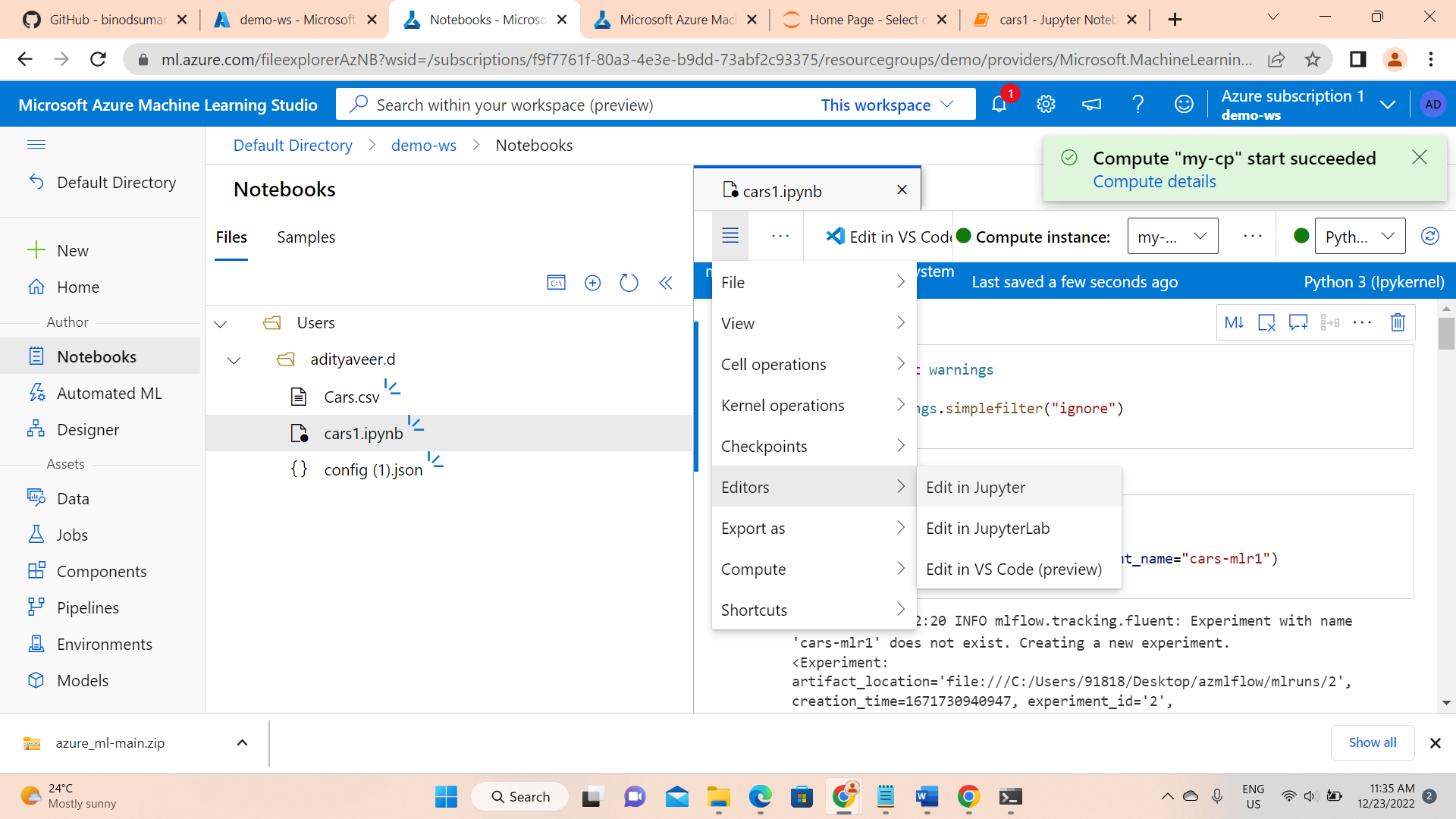
A compute instance is required for creating a notebook kernel .

First create compute instance and then it will start automatically.

Note : choose compute instance SKU : Standard\_DS12\_v2

In below pic, it is showing start compute because my compute instance is already created. If you are using it for first time please go and create with mentioned SKU





Another Important step

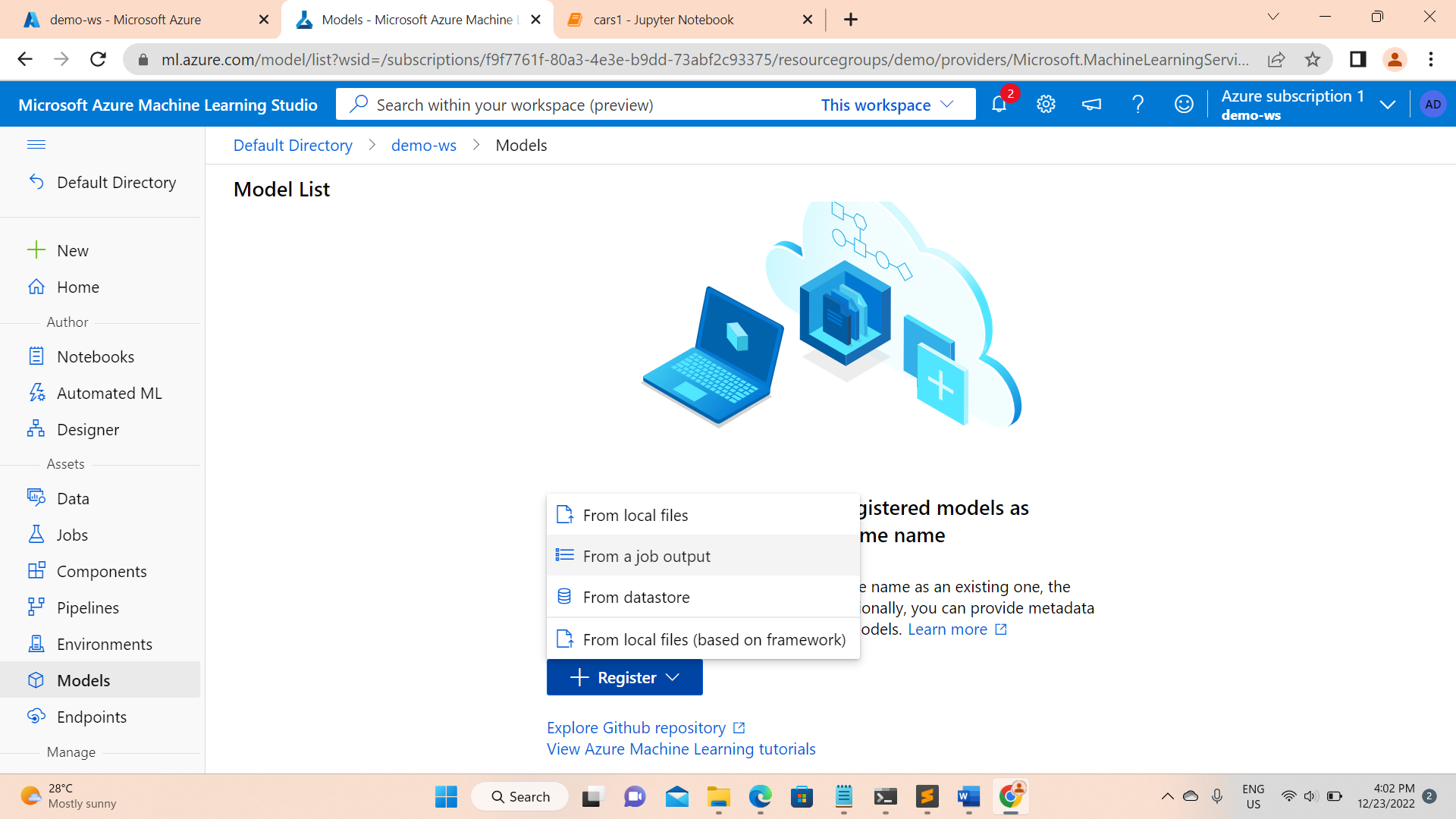
* Select Python3.10-sdkv2 kernel

Once you selected Python 3.10 sdk v2 kernel

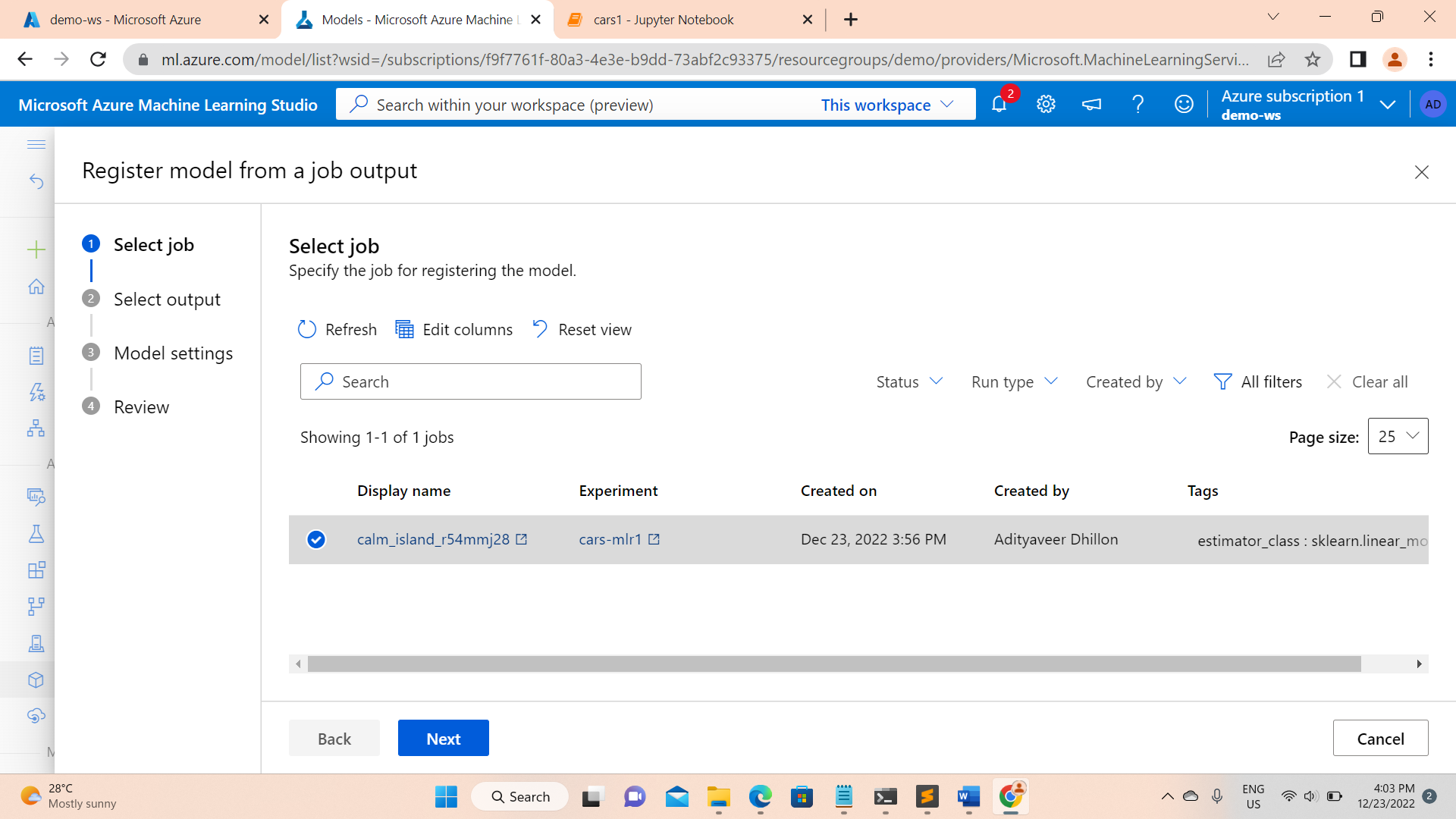
* In code first you must run the %pip install -r requirements.txt
* After successful Installation just comment the # %pip install -r requirements.txt line and run the entire code

Now go to Models section and register the model

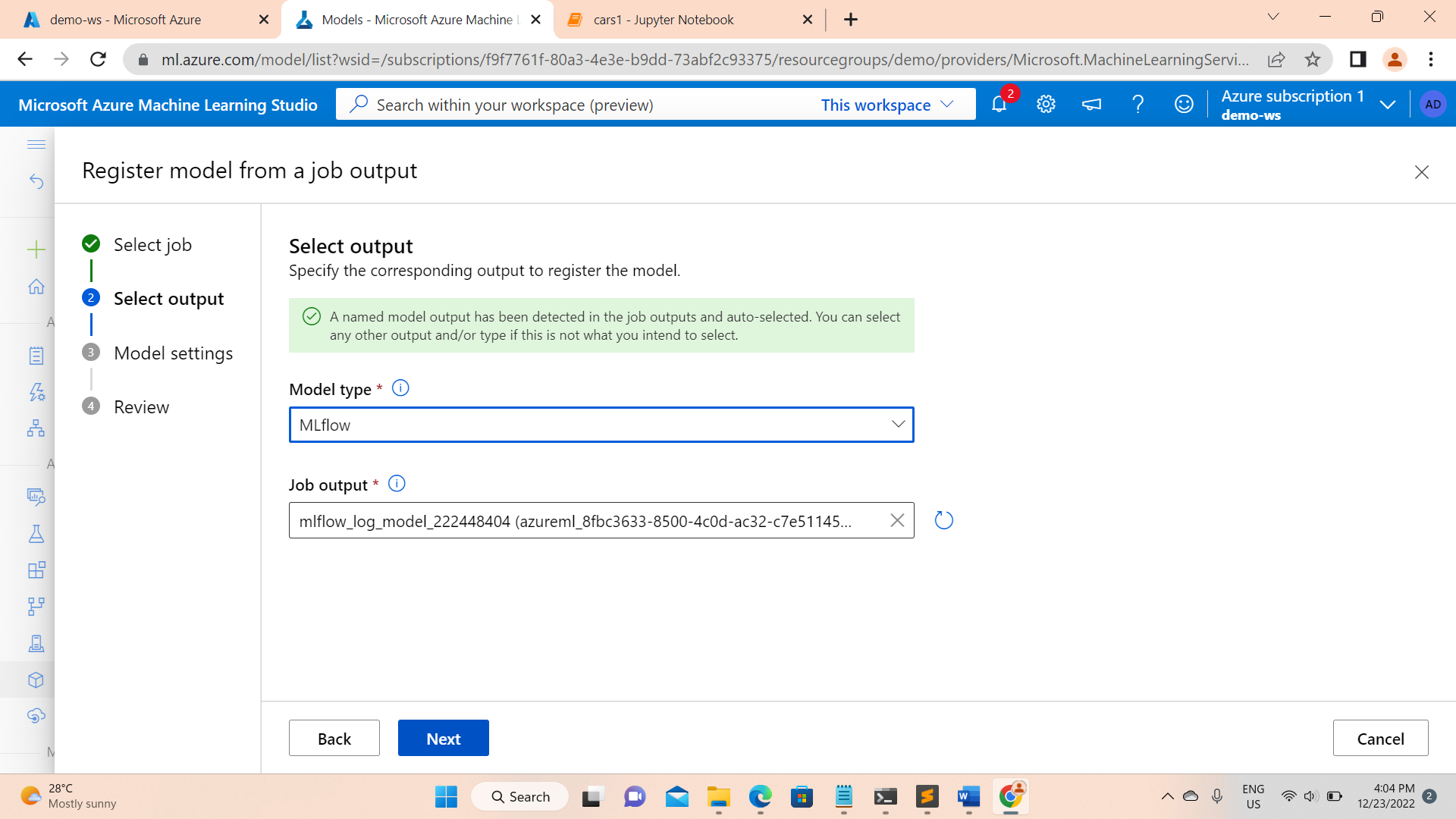
Register the model with output from a job output (click on)



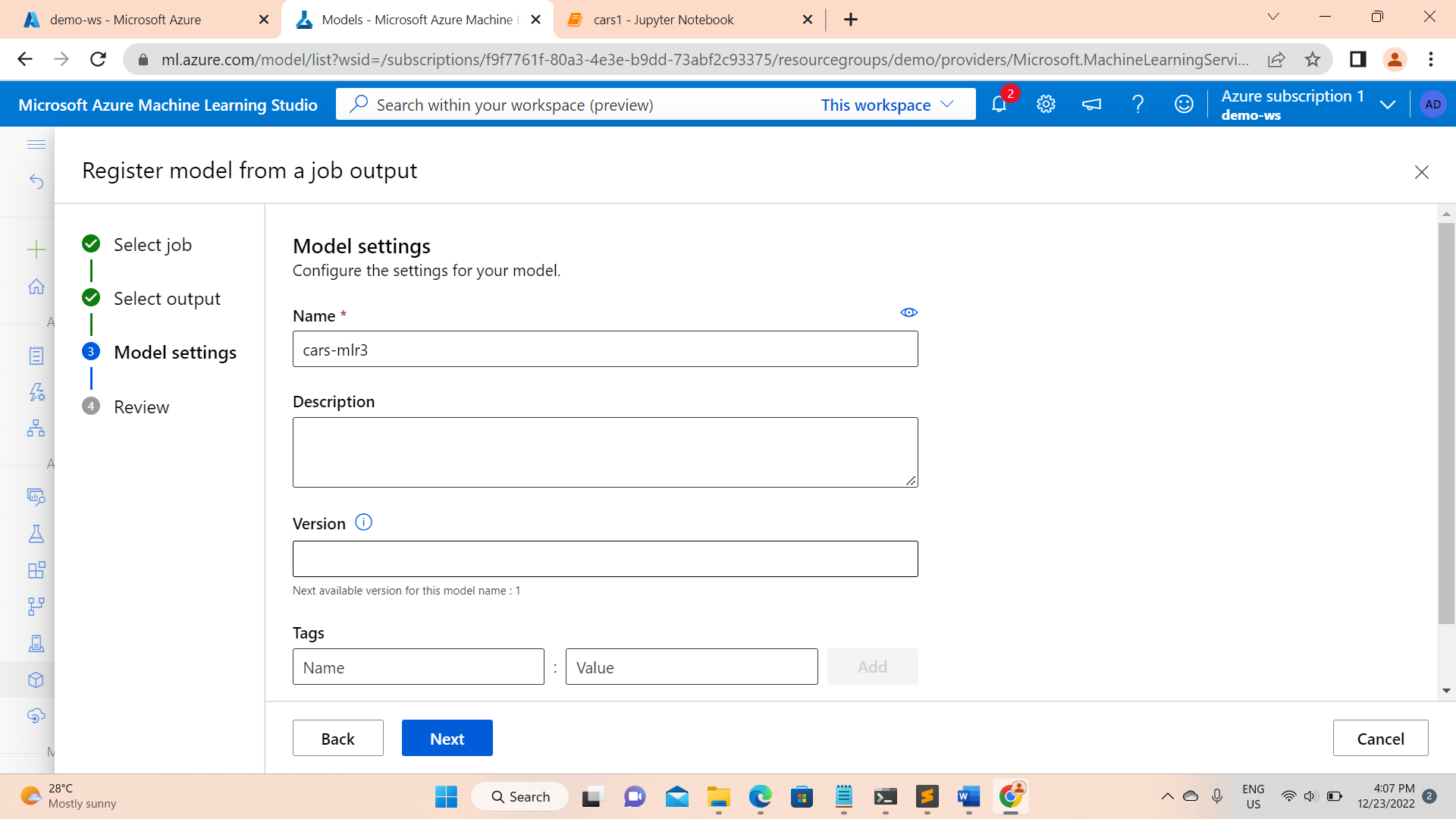
Now select the latest job you can check with created time . Sometimes there will be multiple jobs with same time you will see , just select the top most one



This model take automatically as Mlflow type

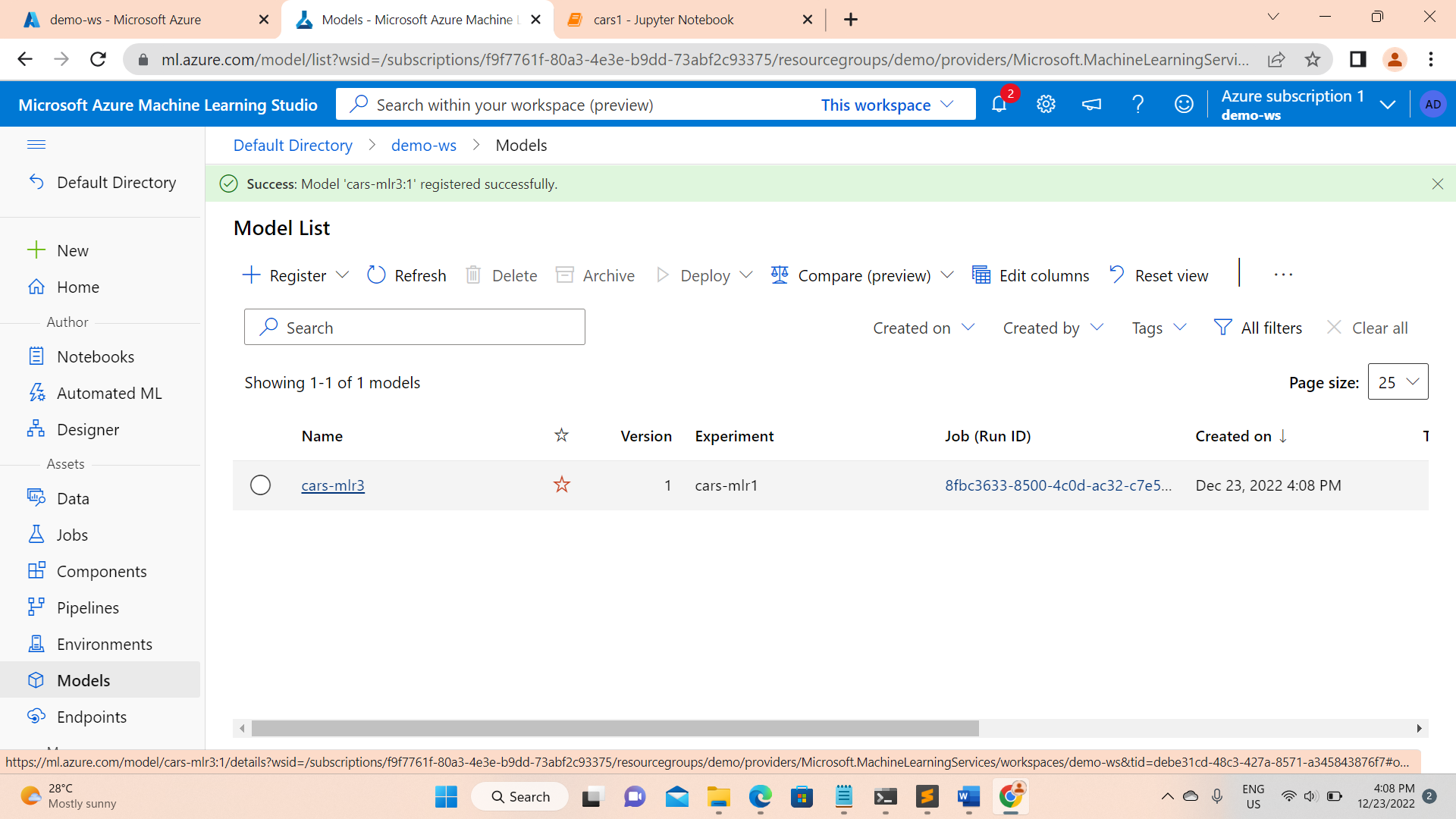


Now configure the model settings (just give the name remaining options are optional)



Review it and create

You can see successful status on top….



Create Endpoint

Last step and very important step……

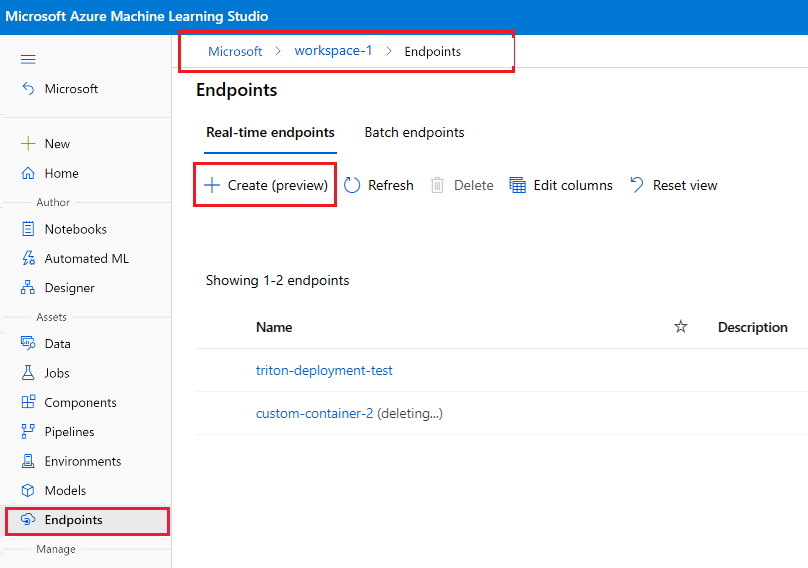
Creating an endpoint required a resource quota

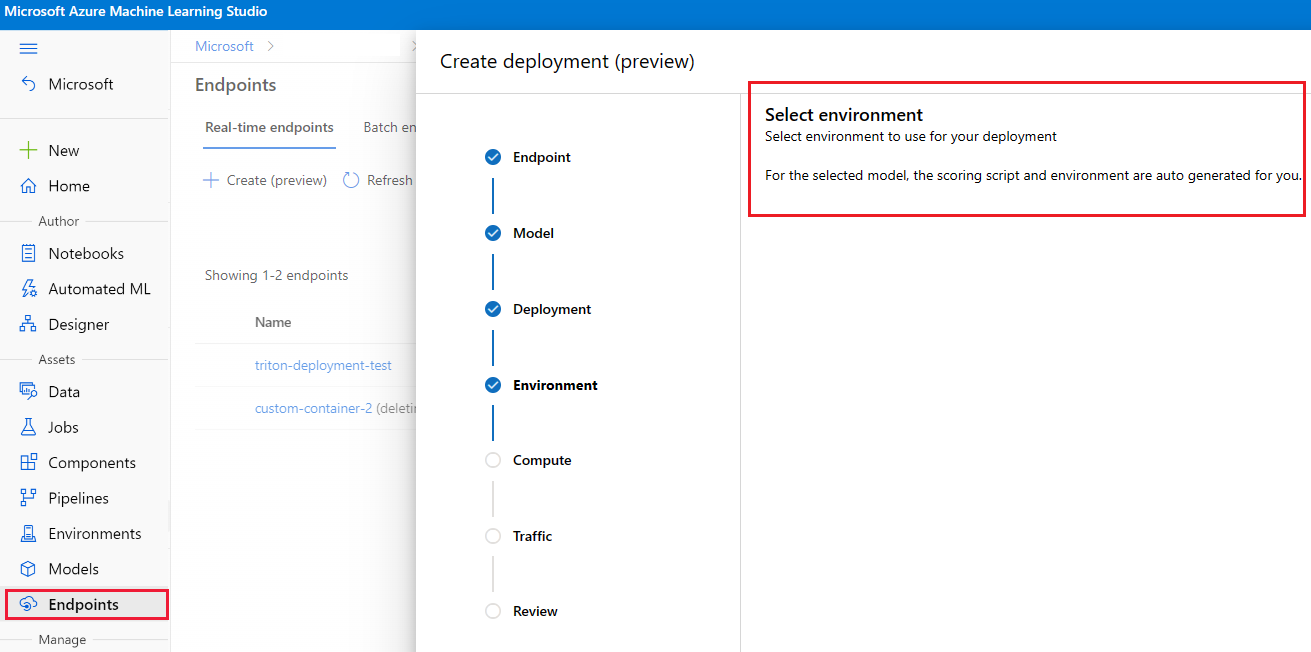
Ideal Resource Quota : Standard\_DS2\_v2 2 cores,7GB (RAM) 14GB(disk) and Instance count : 3

Note : single core SKU wont work atlest 2 cores and min of 7gb ram

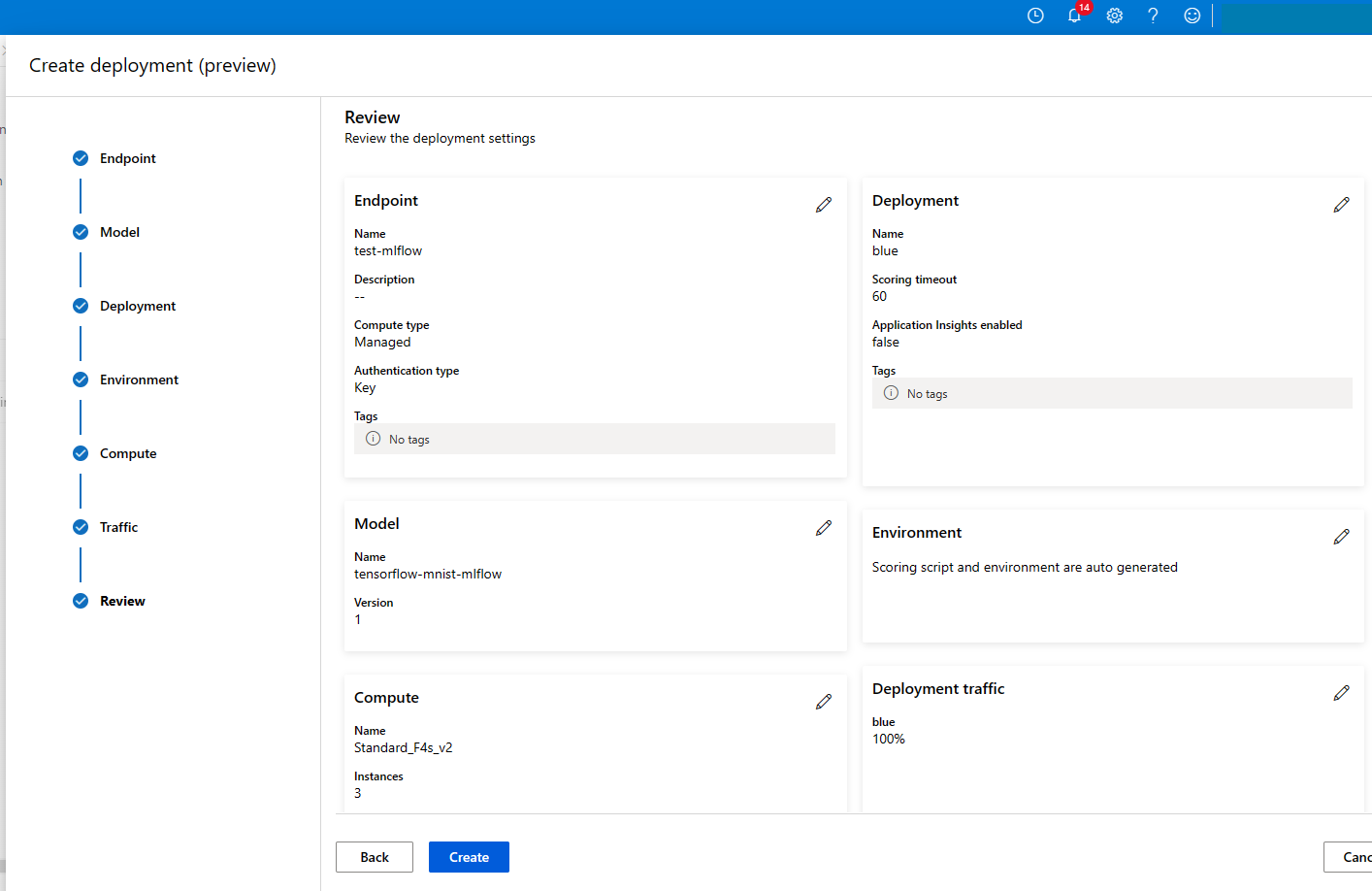
Now go and create enpoint

It is very simple just give the endpoint name and select the model which you registered previously





Complete the wizard to deploy the model to the endpoint.



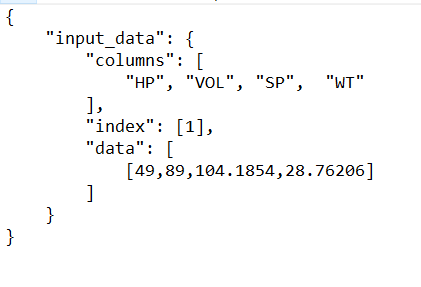
Endpoint creation will take a while (around 20 min to 30 min)

Test your Endpoint :

Copy the input.json file provided inside the folder

Once the endpoint successfully created you will see the “Test” section right next to it

Click on test section , by default you see an input json will already there but sometimes It wont work . Ideally copy our input json file and paste it there .



You can see the results in right side of test section