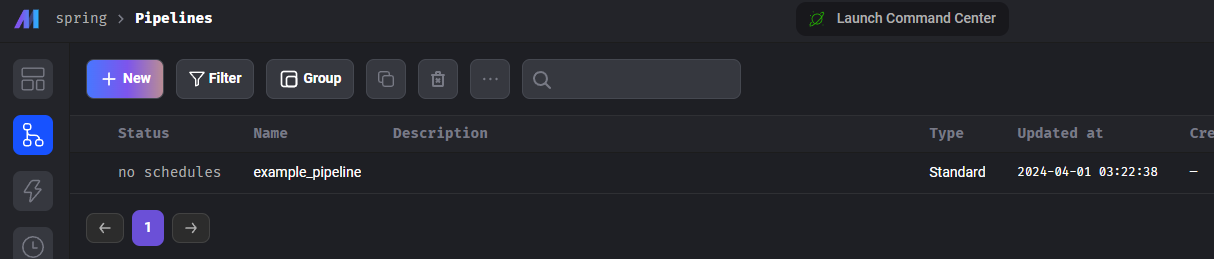
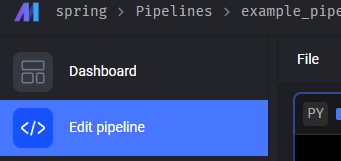
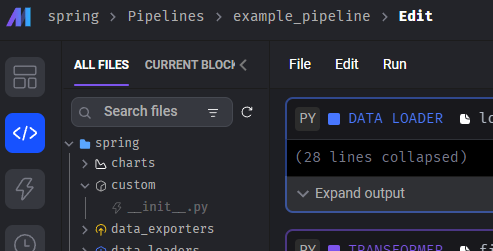
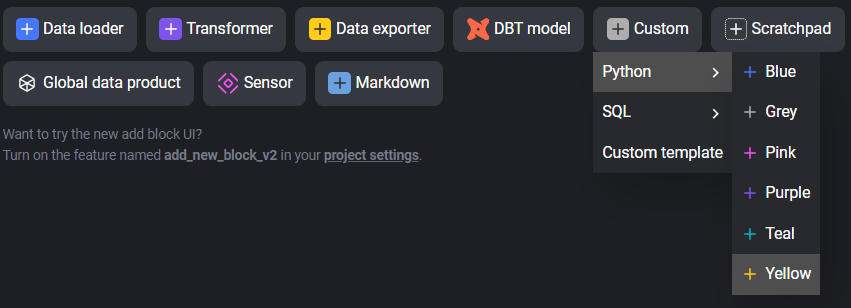
**Machine Learning with mage.ai**

(파이썬 파일 편집은 vscode에서 직접 수정하는 것을 권장)

1. (mage) mage start spring
2. 
3. Example\_pipline 클릭 >> 
4. 블록을 확인하고 단계적으로 실행해본다.
5. Custom으로 machine learning block을 만들고 up/down stream을 구성(선 연결하기)
6.  
7. 파이썬코드는 아래와 같이 만들고 저장한다. (vs code 에서 직접 작성해도 됨)

import pandas as pd

from sklearn.ensemble import RandomForestClassifier

from spring.utils.variables import (

X\_COLS,

Y\_COLS,

)

if "custom" not in globals():

from mage\_ai.data\_preparation.decorators import custom

if "test" not in globals():

from mage\_ai.data\_preparation.decorators import test

def \_model\_save(rf\_model):

"""

Save RandomForest model to a file.

"""

# Here you would implement the logic to save your trained model to a file

# Example:

# with open('random\_forest\_model.pkl', 'wb') as file:

# pickle.dump(rf\_model, file)

pass

@custom

def random\_forest\_train(df: pd.DataFrame, \*args, \*\*kwargs):

"""

Train a Random Forest Classifier and predict the 'Survived' column.

Args:

df: Data frame containing the training data.

Returns:

Data frame with a new column 'Survived\_predict' with predictions.

"""

# Prepare the data

x\_train = df[X\_COLS]

y\_train = df[Y\_COLS].values.ravel() # RandomForest expects a 1D array for y

# Initialize the Random Forest Classifier

rf\_model = RandomForestClassifier()

# Train the model

rf\_model.fit(x\_train, y\_train)

# Predict using the trained model

\_pred = rf\_model.predict(x\_train)

# Optionally save the model

\_model\_save(rf\_model)

# Assign predictions to a new column in the dataframe

df = df.assign(Survived\_predict=\_pred)

return df

@test

def test\_output(output, \*args) -> None:

"""

Template code for testing the output of the block.

Args:

output: The output from the random\_forest\_train function.

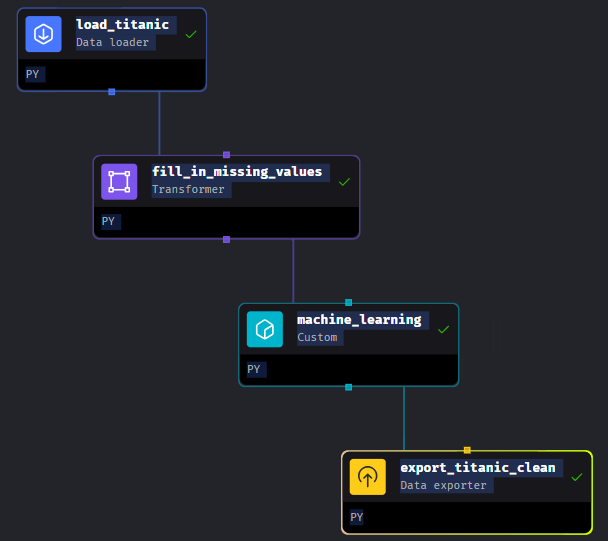
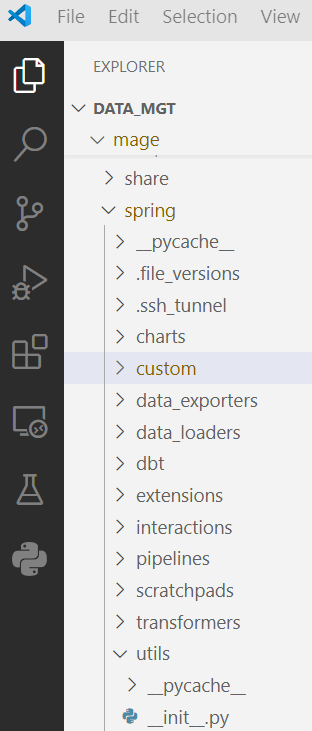
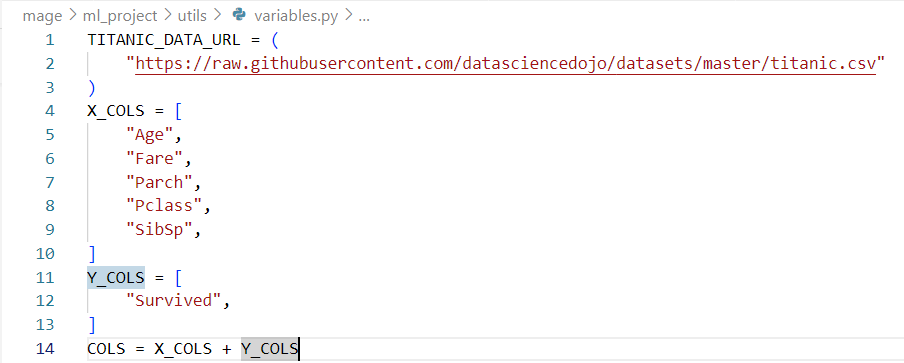
"""

assert output is not None, "The output is undefined"

assert 'Survived\_predict' in output.columns, "Prediction column is missing in the output dataframe"

# You can add more tests to check the quality of your predictions,

# such as accuracy score, confusion matrix, etc.

1. Pipeline 구성 
2. C:\data\_mgt\mage\spring\utils\variables.py 파일을 만든다.
3. 
4. 
5. (mage) C:\data\_mgt\mage>mage start spring 재실행해서 block을 모두 실행한다.
6. ‘spring’를 다시 실행하려고 하며 반드시 폴더 위치를 지켜라 아래와 같이 다른 폴더에서 시작하면 새로운 프로젝트가 생성된다.(주의)

(mage) C:\data\_mgt\mage>mage start spring

# mage 폴더 밑에 spring폴더가 존재

참고자료

* + - 1. 모델을 저장하고 싶으면 [model\_persistence](https://scikit-learn.org/stable/model_persistence.html)
      2. 저장할 파일의 폴더는 만들고 경로를 만드는 방법 [파이썬 폴더/경로 생성](https://minimin2.tistory.com/40)
* Import os
* os.mkdirs(“a/b/c”. exit\_ok = True)
  + - 1. [오픈소스 데이터 파이프라인 툴 mage.ai](https://velog.io/@rektpunk/%EC%98%A4%ED%94%88%EC%86%8C%EC%8A%A4-%EB%8D%B0%EC%9D%B4%ED%84%B0-%ED%8C%8C%EC%9D%B4%ED%94%84%EB%9D%BC%EC%9D%B8-%ED%88%B4-mage.ai) (<https://github.com/RektPunk/mage-ai-example>)