Python data validation using the *schema* library

Owen Price - linkedin.com/in/owenhprice

As an example, I have a DataFrame I want to filter on three columns

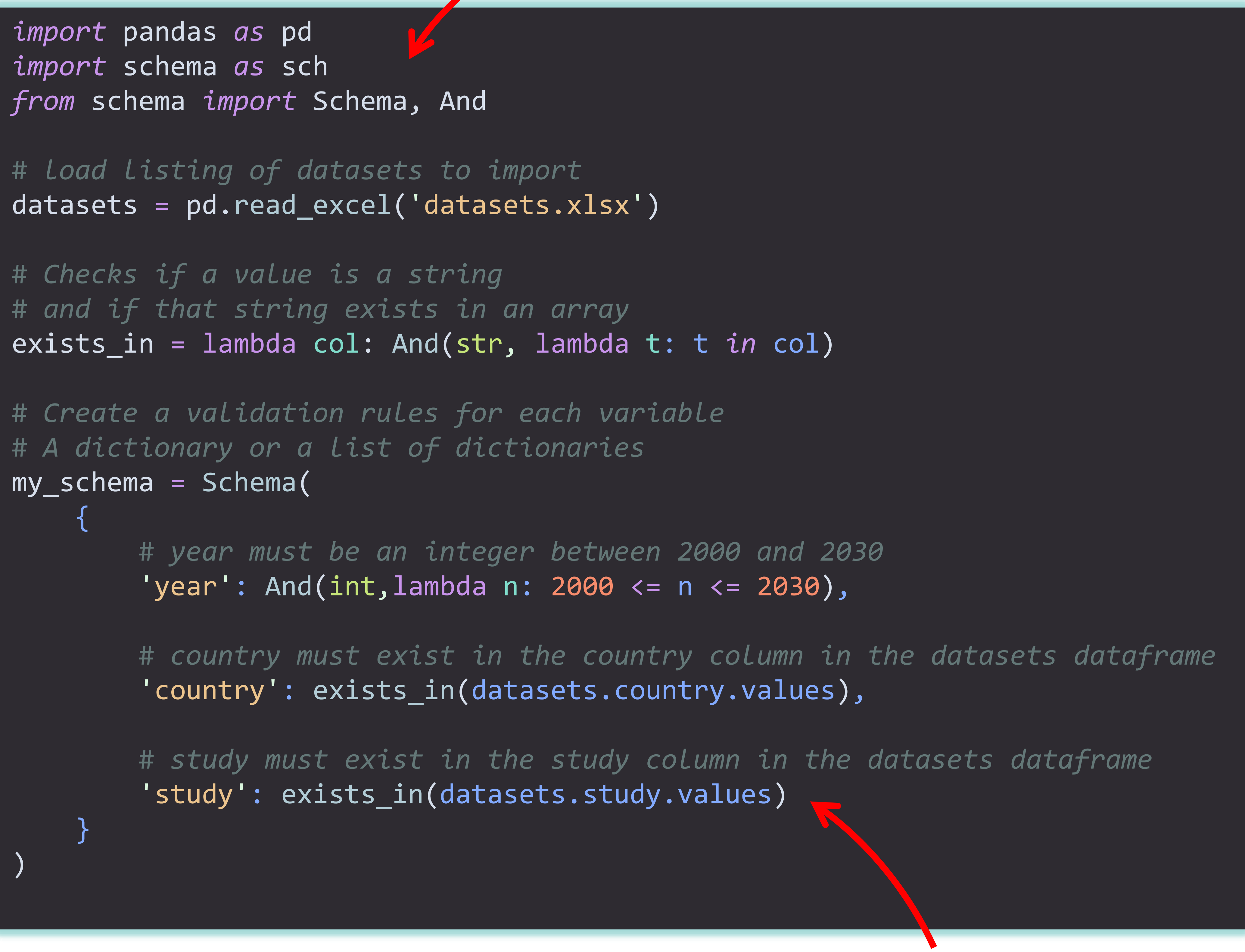
I want to be sure the values I use for the filters are valid and appropriate for those columns

Owen Price - linkedin.com/in/owenhprice

With the *schema* library, I can define validation rules for each of the filters

1. Define a schema

import from the schema library so we can define a validation schema



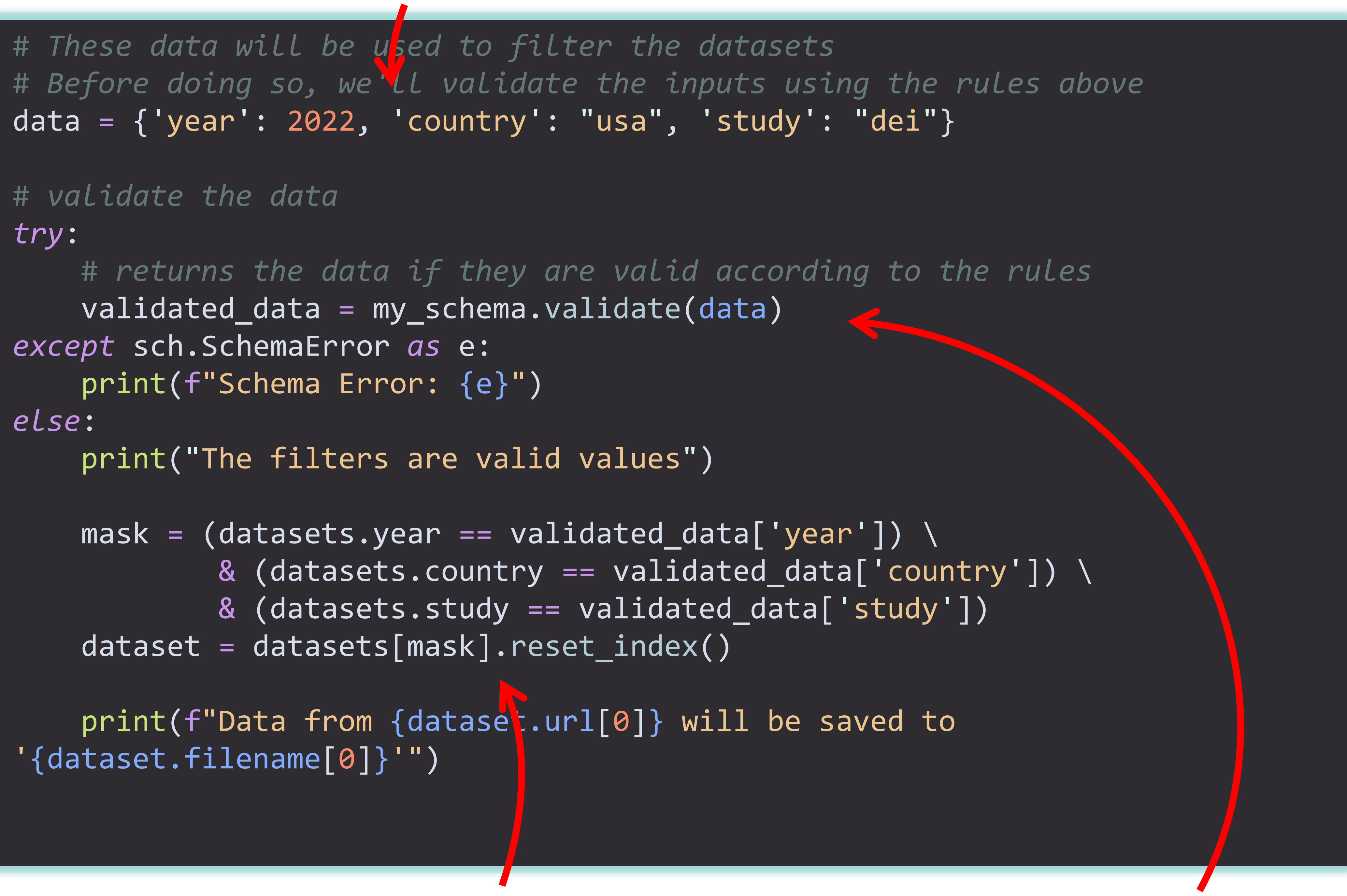
```python
import pandas as pd
import schema as sch
from schema import Schema, And

# load listing of datasets to import
datasets = pd.read_excel('datasets.xlsx')

# Checks if a value is a string
# and if that string exists in an array
exists_in = lambda col: And(str, lambda t: t in col)

# Create a validation rules for each variable
# A dictionary or a list of dictionaries
my_schema = Schema(
    {
        # year must be an integer between 2000 and 2030
        'year': And(int,lambda n: 2000 <= n <= 2030),

        # country must exist in the country column in the datasets dataframe
        'country': exists_in(datasets.country.values),

        # study must exist in the study column in the datasets dataframe
        'study': exists_in(datasets.study.values)
    }
)
```

Define a rule for each variable we want to validate.

Owen Price - linkedin.com/in/owenhprice

2. Validate the data

Data in the same structure as the schema we defined



```
# These data will be used to filter the datasets
# Before doing so, we'll validate the inputs using the rules above
data = {'year': 2022, 'country': "usa", 'study': "dei"}

# validate the data
try:
    # returns the data if they are valid according to the rules
    validated_data = my_schema.validate(data)
except sch.SchemaError as e:
    print(f"Schema Error: {e}")
else:
    print("The filters are valid values")

    mask = (datasets.year == validated_data['year']) \
           & (datasets.country == validated_data['country']) \
           & (datasets.study == validated_data['study'])
    dataset = datasets[mask].reset_index()

    print(f"Data from {dataset.url[0]} will be saved to
'{dataset.filename[0]}'")
```

Use the validated inputs knowing they have passed the tests

If the validation rules pass, the data are passed through to ‘validated_data’