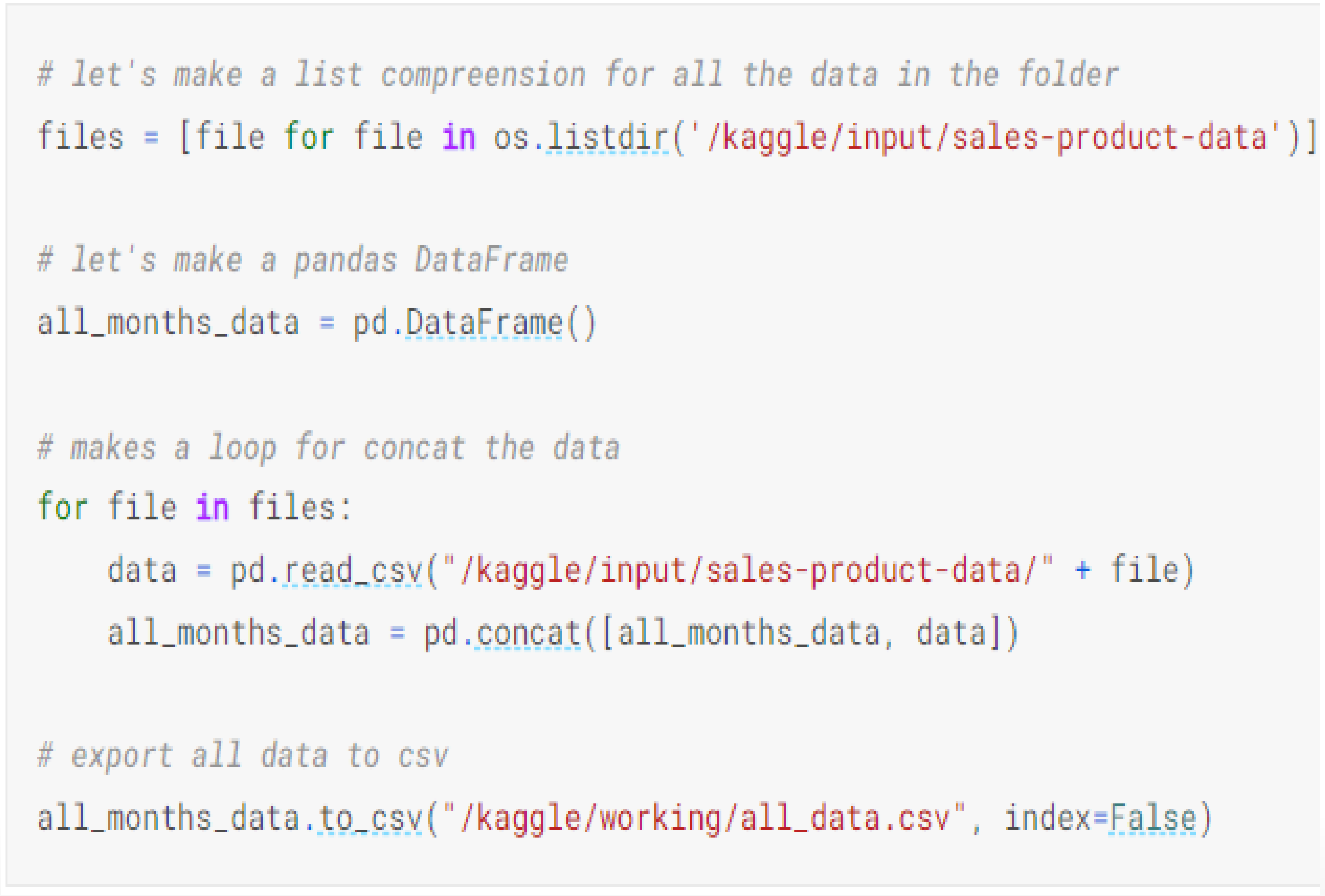
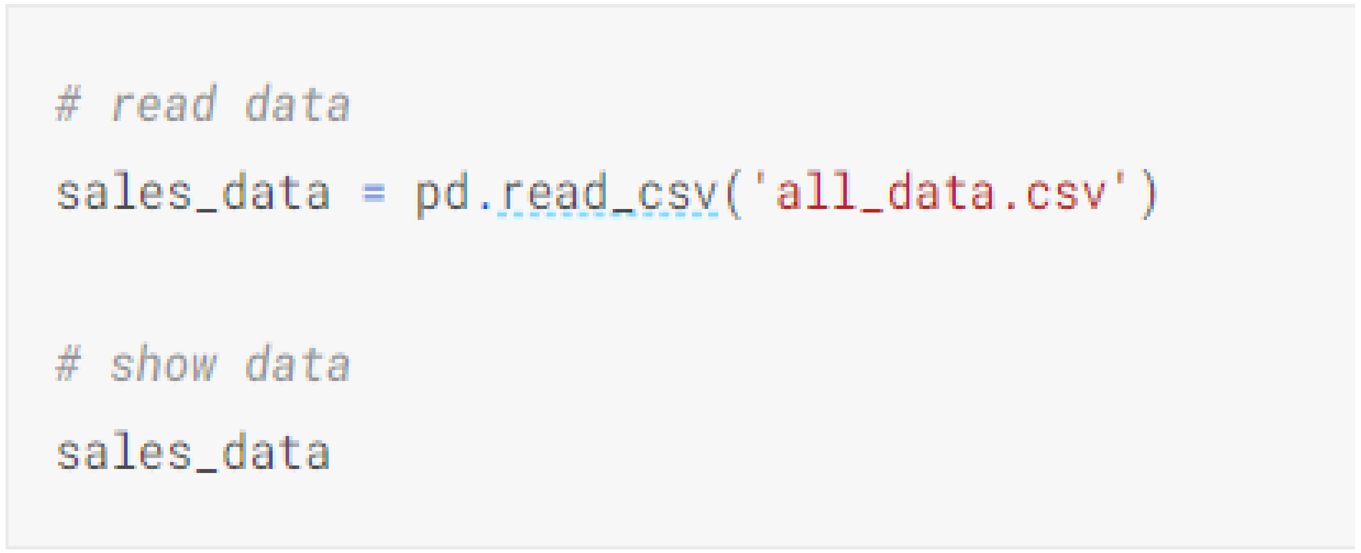
Product sales analysis

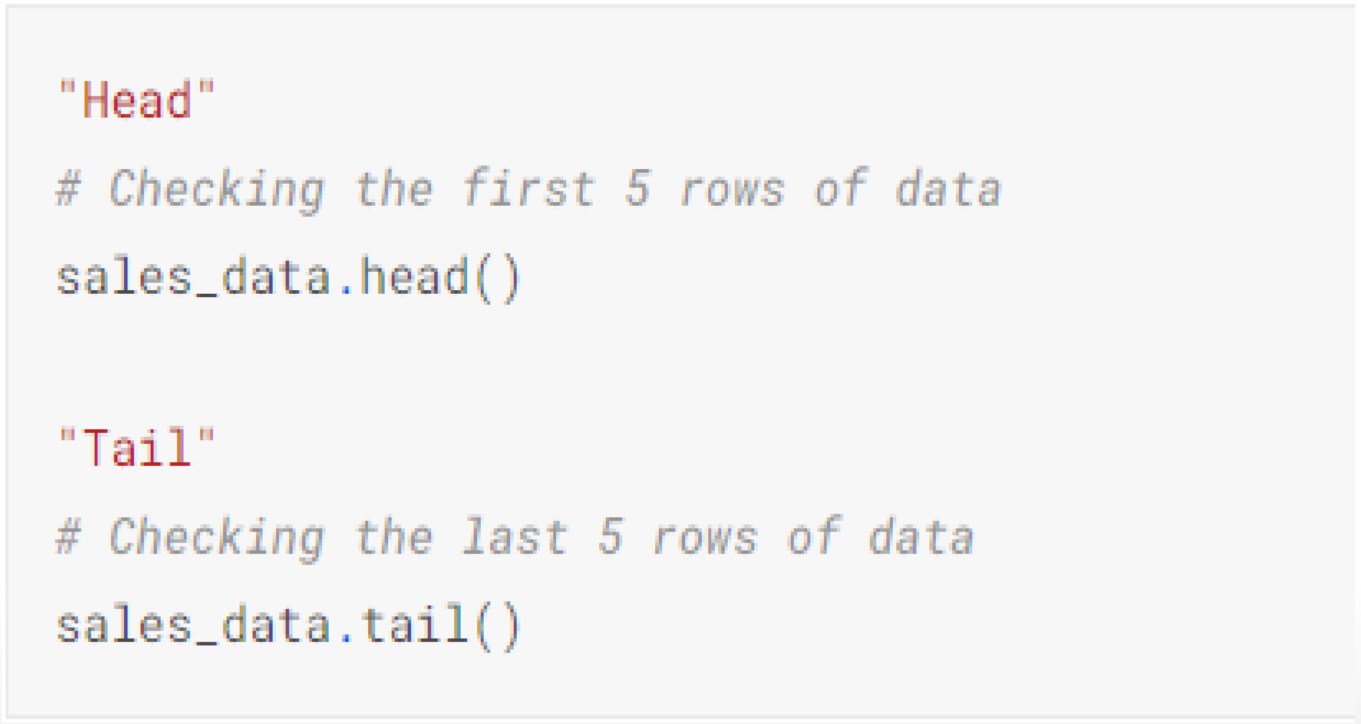
Phase-2



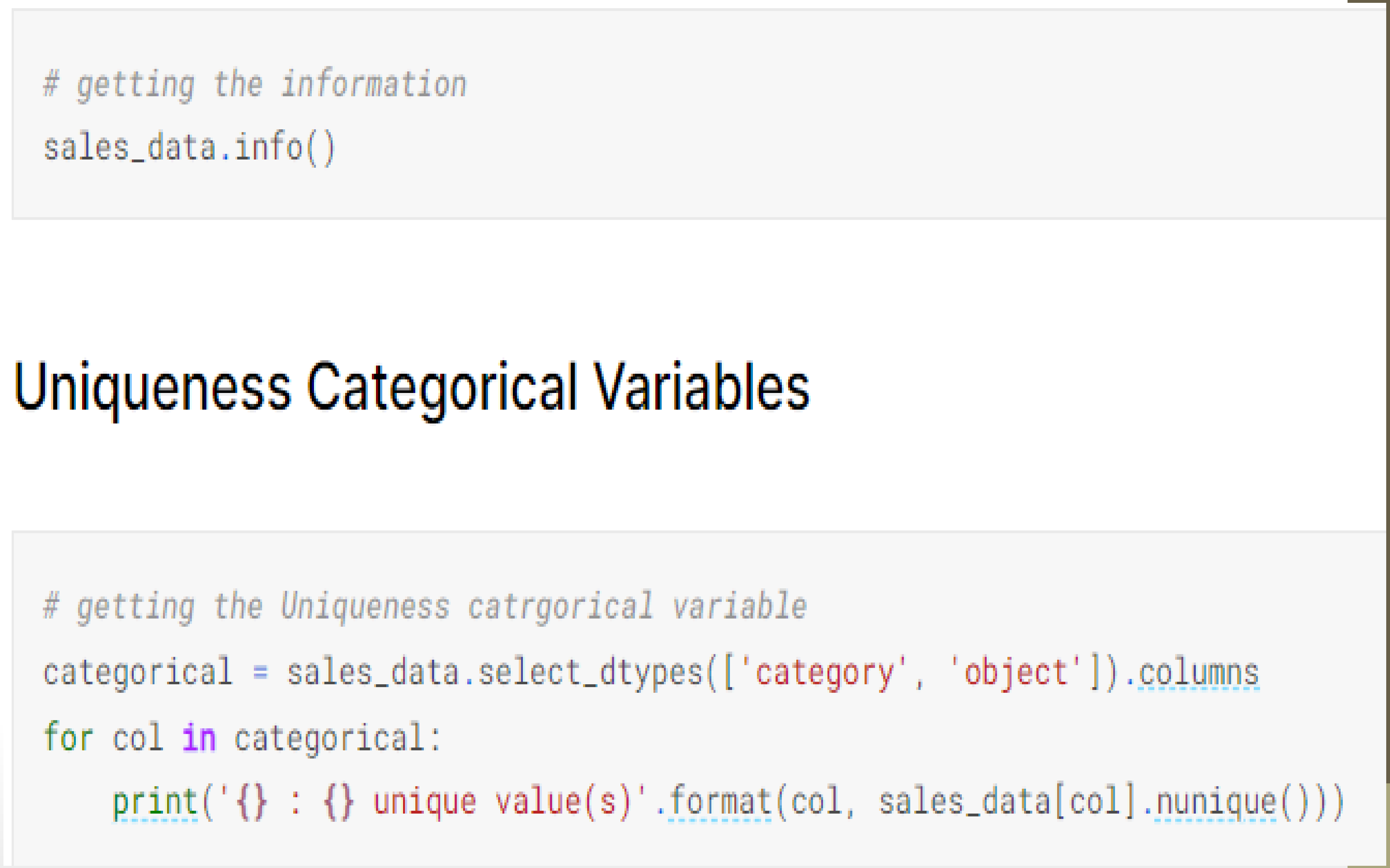
```
# let's make a list compreension for all the data in the folder
files = [file for file in os.listdir('/kaggle/input/sales-product-data')]

# let's make a pandas DataFrame
all_months_data = pd.DataFrame()

# makes a loop for concat the data
for file in files:
    data = pd.read_csv("/kaggle/input/sales-product-data/" + file)
    all_months_data = pd.concat([all_months_data, data])

# export all data to csv
all_months_data.to_csv("/kaggle/working/all_data.csv", index=False)
```



```python
# read data
sales_data = pd.read_csv('all_data.csv')

# show data
sales_data
```



```python
"Head"
# Checking the first 5 rows of data
sales_data.head()

"Tail"
# Checking the last 5 rows of data
sales_data.tail()
```

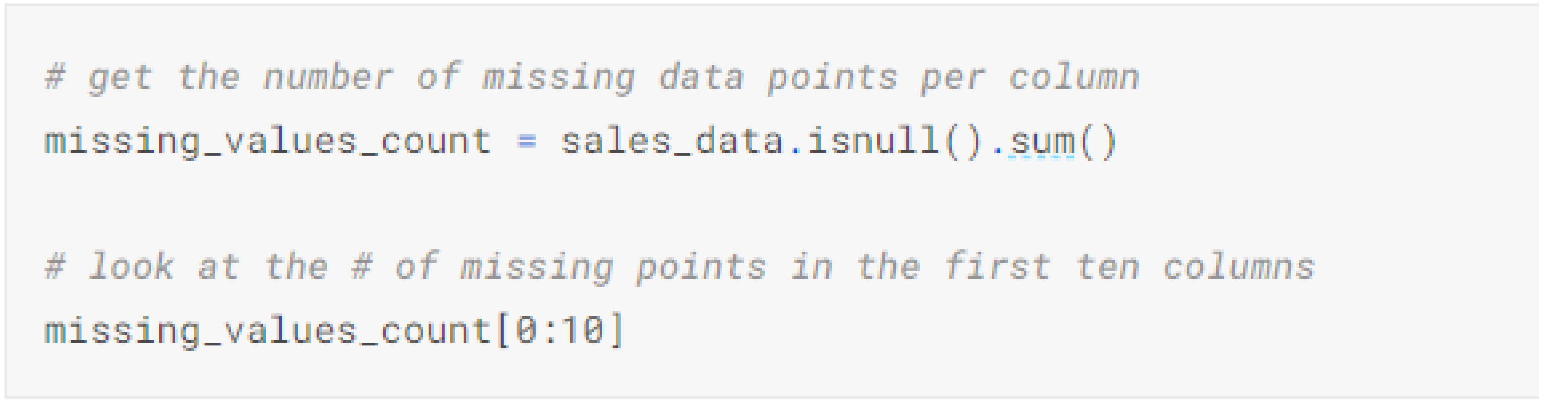


```
# getting the information
sales_data.info()
```

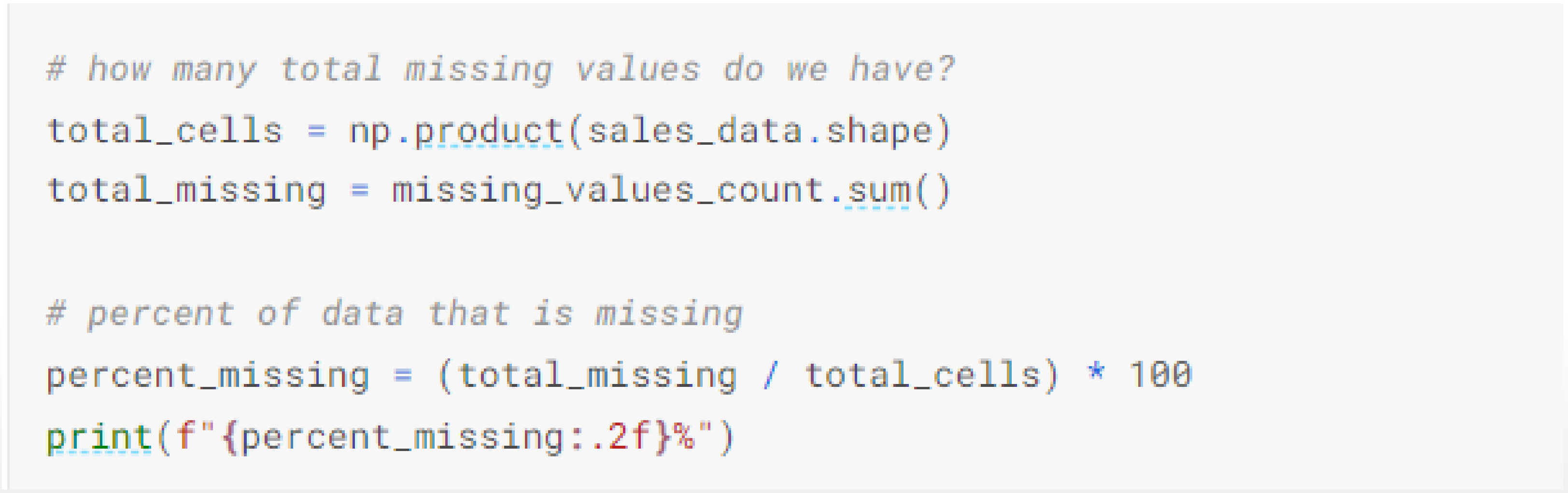
Uniqueness Categorical Variables

```
# getting the Uniqueness catrgorical variable
categorical = sales_data.select_dtypes(['category', 'object']).columns
for col in categorical:
    print('{} : {} unique value(s)'.format(col, sales_data[col].nunique()))
```

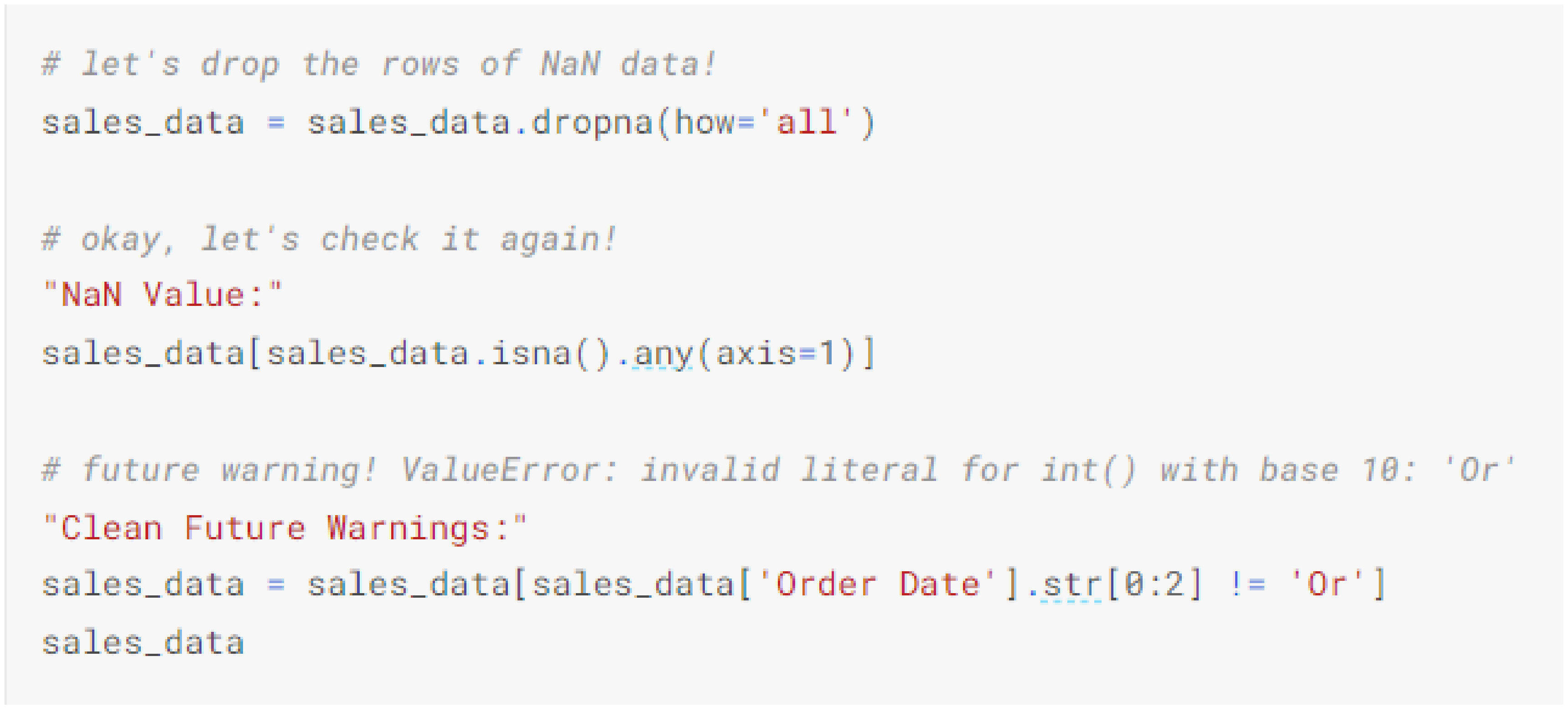
How many missing data points do we have?



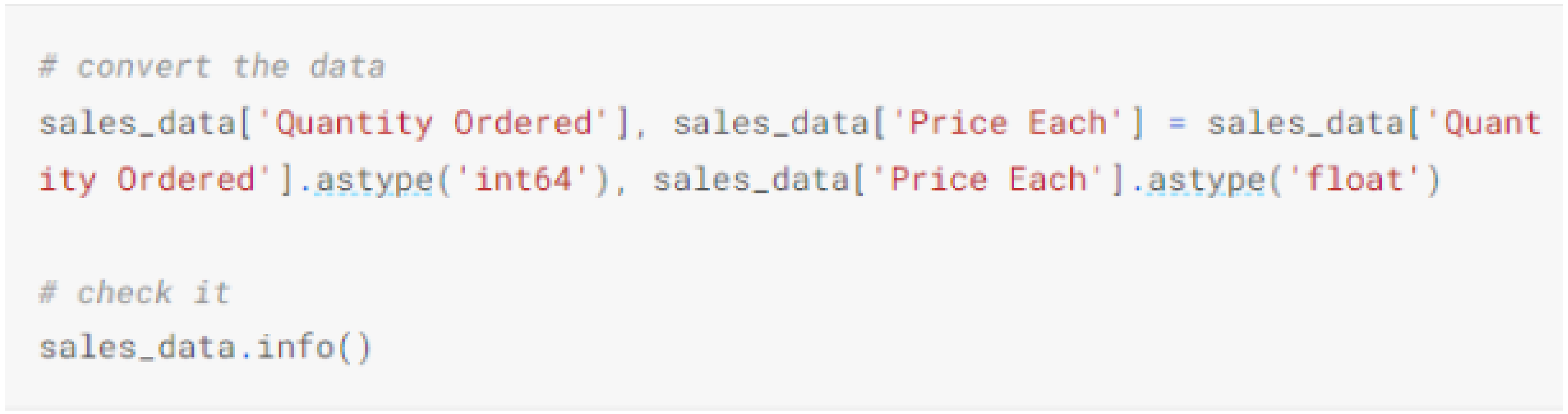
```python
# get the number of missing data points per column
missing_values_count = sales_data.isnull().sum()

# look at the # of missing points in the first ten columns
missing_values_count[0:10]
```



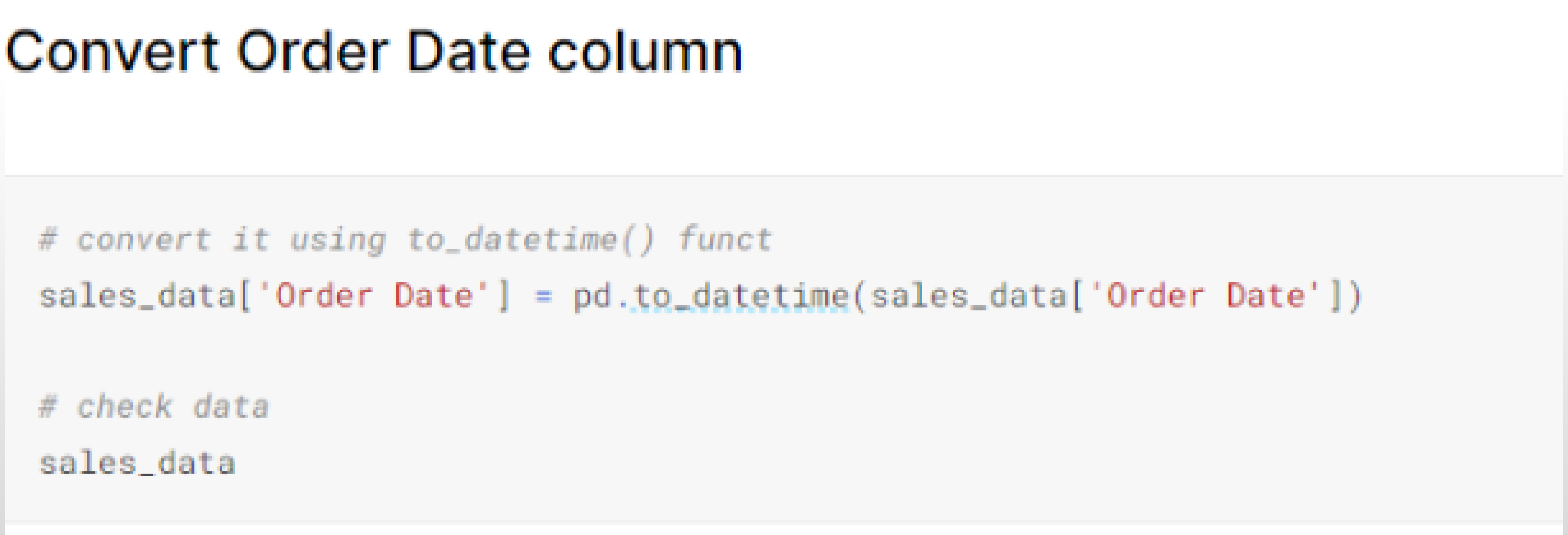
```python
# how many total missing values do we have?
total_cells = np.product(sales_data.shape)
total_missing = missing_values_count.sum()

# percent of data that is missing
percent_missing = (total_missing / total_cells) * 100
print(f"{percent_missing:.2f}%")
```



```
# let's drop the rows of NaN data!
sales_data = sales_data.dropna(how='all')

# okay, let's check it again!
"NaN Value:"
sales_data[sales_data.isna().any(axis=1)]

# future warning! ValueError: invalid literal for int() with base 10: 'Or'
"Clean Future Warnings:"
sales_data = sales_data[sales_data['Order Date'].str[0:2] != 'Or']
sales_data
```



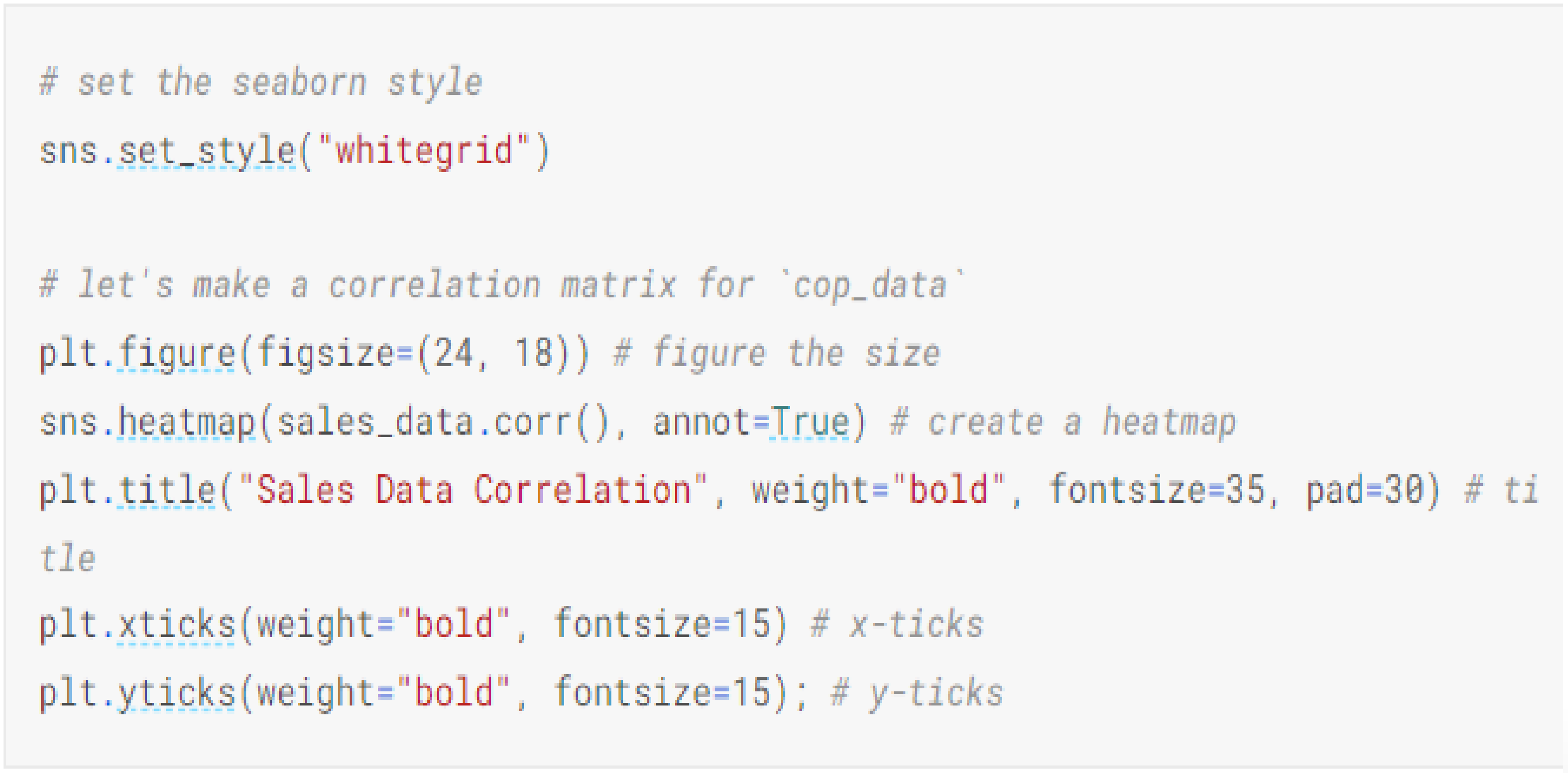
```
# convert the data
sales_data['Quantity Ordered'], sales_data['Price Each'] = sales_data['Quant
ity Ordered'].astype('int64'), sales_data['Price Each'].astype('float')

# check it
sales_data.info()
```



Convert Order Date column

```
# convert it using to_datetime() funct
sales_data['Order Date'] = pd.to_datetime(sales_data['Order Date'])

# check data
sales_data
```

Data Analysis



```
# set the seaborn style
sns.set_style("whitegrid")

# let's make a correlation matrix for `cop_data`
plt.figure(figsize=(24, 18)) # figure the size
sns.heatmap(sales_data.corr(), annot=True) # create a heatmap
plt.title("Sales Data Correlation", weight="bold", fontsize=35, pad=30) # ti
tle
plt.xticks(weight="bold", fontsize=15) # x-ticks
plt.yticks(weight="bold", fontsize=15); # y-ticks
```

Recap Data:

- We have total 186850 records and 6 columns cateogircal type
- The total of missing value that we have is 0.29167 %
- Order ID : 178438 unique value(s)
- Product : 20 unique value(s)
- Quantity Ordered : 10 unique value(s)
- Price Each : 24 unique value(s)
- Order Date : 142396 unique value(s)
- Purchase Address : 140788 unique value(s)