

Efficient Summaries with Custom Functions (Updated Solution)

This document includes the question, the solution, and a breakdown of the code provided in the screenshot.

Uploaded Screenshot

Below is the screenshot of the task:

The screenshot shows a web browser window displaying a DataCamp exercise titled "Efficient summaries". The exercise is part of the "Data Manipulation with pandas" course. The instructions are as follows:

- 1. Use the custom `iqr` function defined for you along with `.agg()` to print the IQR of the `temperature_c` column of `sales`.
- 2. Update the column selection to use the custom `iqr` function with `.agg()` to print the IQR of `temperature_c`, `fuel_price_usd_per_l`, and `unemployment`, in that order.
- 3. Update the aggregation functions called by `.agg()` to include `iqr` and `np.median` in that order.

The solution code is provided in a code editor on the right side of the interface. The code is as follows:

```
1 # Import NumPy
2 import numpy as np
3
4 # A custom IQR function
5 def iqr(column):
6     return column.quantile(0.75) - column.quantile(0.25)
7
8 # Make sure to select the correct columns and use .agg() with the iqr function
9 print(sales[['temperature_c', 'fuel_price_usd_per_l', 'unemployment']].agg(iqr))
10
11 # Apply multiple aggregation functions
12 print(sales[['temperature_c', 'fuel_price_usd_per_l', 'unemployment']].agg([iqr, np.median]))
13
```

The output of the code is displayed in the IPython Shell on the right side of the interface. The output is:

```
<script.py> output:
14 5833333333333336
```

Question

Update the column selection to use the custom `iqr` function with `.agg()` to print the IQR of `temperature_c`, `fuel_price_usd_per_l`, and `unemployment`, in that order.

Answer

```
# Import NumPy
import numpy as np
```

```
# A custom IQR function
def iqr(column):
    return column.quantile(0.75) - column.quantile(0.25)
```

```
# Aggregate the selected columns using the iqr function
```

```
print(sales[['temperature_c', 'fuel_price_usd_per_l',  
'unemployment']].agg(iqr))
```

Code Explanation

Explanation of the code:

1. ``import numpy as np``: Imports the NumPy library. Even though not needed for this specific part, it's a good practice to import it for later use.
2. ``def iqr(column):``: Defines a custom function ``iqr`` to calculate the interquartile range (IQR) of a column by subtracting the 25th percentile from the 75th percentile.
3. ``sales[['temperature_c', 'fuel_price_usd_per_l', 'unemployment']].agg(iqr)``: Applies the ``iqr`` function to the selected columns (``temperature_c``, ``fuel_price_usd_per_l``, and ``unemployment``) in the ``sales`` DataFrame and prints the results in the specified order.