

Create a dataframe of ten rows, four columns with random values. Write a Pandas program to highlight the negative numbers red and positive numbers black.

CODE:

```
10.py - C:/Users/keert/AppData/Local/Programs/Python/Python311/query processing new/10.py (3.11.4)
File Edit Format Run Options Window Help

import pandas as pd
import numpy as np

# Create a DataFrame with random values
np.random.seed(0) # For reproducibility
data = np.random.randn(10, 4) # 10 rows, 4 columns

# Creating the DataFrame
df = pd.DataFrame(data, columns=['A', 'B', 'C', 'D'])
df.index = np.arange(1, 11) # Setting the index to 1 to 10

# Define a function to highlight negative numbers in red and positive numbers in black
def highlight_negative(val):
    color = 'red' if val < 0 else 'black'
    return f'color: {color}'

# Apply the function to the DataFrame
styled_df = df.style.applymap(highlight_negative)

# Displaying the DataFrame as static text with ANSI color codes for console
def highlight_console(df):
    result = ""
    for idx, row in df.iterrows():
        row_str = f"{idx:2d} "
        for val in row:
            color_code = "\033[91m" if val < 0 else "\033[30m"
            row_str += f"{color_code}{val: .6f}\033[0m "
        result += row_str.strip() + "\n"
    return result

print(highlight_console(df))
```

OUTPUT:

```
IDLE Shell 3.11.4
File Edit Shell Debug Options Window Help

Python 3.11.4 (tags/v3.11.4:d2340ef, Jun 7 2023, 05:45:37) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>>
= RESTART: C:/Users/keert/AppData/Local/Programs/Python/Python311/query processing new/10.py

Warning (from warnings module):
File "C:/Users/keert/AppData/Local/Programs/Python/Python311/query processing new/10.py", line 18
    styled_df = df.style.applymap(highlight_negative)
FutureWarning: Styler.applymap has been deprecated. Use Styler.map instead.
1 0.17640528 0.4001578 0.9787388 2.2408938
2 1.8675588 -0.9772788 0.9500888 -0.1513578
3 -0.1032198 0.4105998 0.1440448 1.4542748
4 0.7610388 0.1216758 0.4438638 0.3336748
5 1.4940798 -0.2051588 0.3130688 -0.8540968
6 -2.5529908 0.6536198 0.8644368 -0.7421658
7 2.2697558 -1.4543668 0.0457598 -0.1871848
8 1.5327798 1.4693598 0.1549478 0.3781638
9 -0.8877868 -1.9807968 -0.3479128 0.1563498
10 1.2302918 1.2023808 -0.3873278 -0.3023038

>>>
```