V2_modifying_replacing_values

August 17, 2025

1 Activity: Modifying & Replacing Values

1.1 Introduction

In this activity you will practice modifying and replacing values in a DataFrame using the various method that Pandas has to offer. This activity will cover the following, not necessarily in this order: - Checking for anomalous values - Using `.isnumeric()` - Using `min()` and `max()` methods - Using `.loc[]` to replace values - Using `isnull()` and `notnull()` methods

Question 1 Create a `DataFrame` called `df` from the given CSV file `employee_data.csv`, and then create a mask called `valid_names` that checks the `Name` column for any non-numeric values.

[5]:
```
import pandas as pd

df = pd.read_csv("employee_data.csv")

valid_names = df["Name"].str.isnumeric()

print(df[valid_names])
```

```
   Name  Years of Employment  Weeks of Vacation Position
75    1                   -1               43.0  Unknown
76    1                   -5               51.0  Unknown
77    1                   -3               40.0  Unknown
78    1                    0               49.0  Unknown
79    1                    0               47.0  Unknown
80    1                   -5               46.0  Unknown
81    1                   -4               52.0  Unknown
82    1                    0               48.0  Unknown
```

[ ]:
```
# Question 1 Grading Checks

assert isinstance(df, pd.DataFrame), 'Have you created a DataFrame named df?'
assert isinstance(valid_names, pd.Series), 'Have you created a Series named␣
 ↪valid_names?'
```

Question 2 Using the original `DataFrame` `df`, create a mask called `unknown_position` that checks the `Position` column for any values that are equal to the string `Unknown`. Then, replace all such values with `Engineer`.

[3]:
```
unknown_position = df["Position"] == "Unknown"


df.loc[unknown_position, "Position"] = "Engineer"


print(df)
```

```
                Name  Years of Employment  Weeks of Vacation  Position
0   Jennifer Jackson                    9                4.0  Engineer
1    Michael Johnson                    9                6.0   Analyst
2         Robert Lee                   13                3.0  Engineer
3        Linda Jones                    3                6.0   Manager
4       Karen Thomas                   14                2.0    Intern
..               ...                  ...                ...       ...
78                 1                    0               49.0  Engineer
79                 1                    0               47.0  Engineer
80                 1                   -5               46.0  Engineer
81                 1                   -4               52.0  Engineer
82                 1                    0               48.0  Engineer

[83 rows x 4 columns]
```

[ ]:
```
# Question 2 Grading Checks

assert isinstance(unknown_position, pd.Series), 'Have you created a Series␣
↪named unknown_position?'
```

Question 3 Using the original `DataFrame` `df`, create a mask called `invalid_vacation` that checks the `Weeks of Vacation` column for any values that are null or missing. Then, use that mask to assign the value 0 to them.

[4]:
```
invalid_vacation = df["Weeks of Vacation"].isnull()

df.loc[invalid_vacation, "Weeks of Vacation"] = 0

print(df)
```

```
                Name  Years of Employment  Weeks of Vacation  Position
0   Jennifer Jackson                    9                4.0  Engineer
1    Michael Johnson                    9                6.0   Analyst
2         Robert Lee                   13                3.0  Engineer
3        Linda Jones                    3                6.0   Manager
```

```
4       Karen Thomas                   14                2.0    Intern
..               …                    …                 …       …
78                1                    0               49.0  Engineer
79                1                    0               47.0  Engineer
80                1                   -5               46.0  Engineer
81                1                   -4               52.0  Engineer
82                1                    0               48.0  Engineer

[83 rows x 4 columns]
```

[ ]:
```
# Question 3 Grading Checks

assert isinstance(invalid_vacation, pd.Series), 'Have you created a Series␣
↪named invalid_vacation?'
```