4.5.5 Remove Strings: strip() vs. replace()

Maria appreciates your help finding all of the prefixes and suffixes in students' names and would now like your help removing them. This is all part of the data cleaning process. To clean the students' names, you'll be using Python string methods.

To clean the students' names, you will have to remove the unique prefixes and suffixes. To recap our findings from the last section, the unique prefixes we need to remove are:

- Dr.
- Mr.
- Mrs.
- Miss

And, the unique suffixes we need to remove are:

- MD
- DDS

- DVM
- Ph.D

Remember, we don't want to remove the family-related suffixes like Jr., II, III, IV, and V.

We have two options to remove these prefixes and suffixes: the `strip()` method and the `replace()` method. Let's compare them to figure out which one we should use.

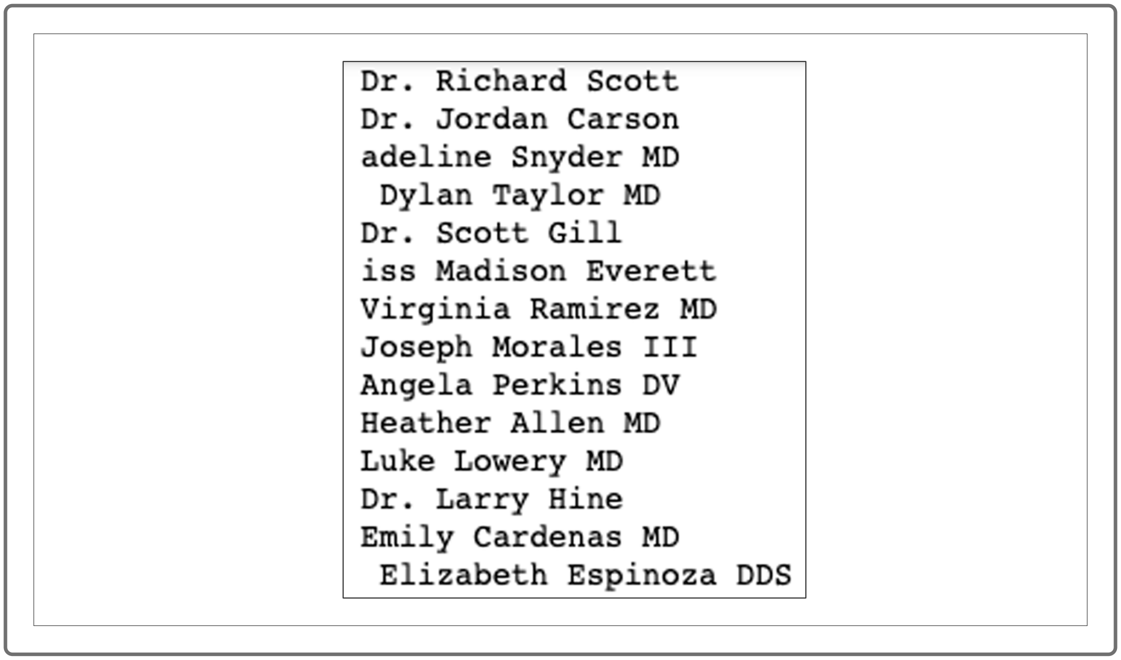
The strip() Method

The `strip()` method removes any combination of letters and words that are inside the parentheses. For example, if we put "Mrs." inside the parentheses of the `strip()` method, any combination of the letters "M," "r," and "s" will be stripped from the student names.

Let's test this in the `cleaning_student_names.ipynb` file. Strip "Mrs." from the students' names using the following code:

```
# Strip "Mrs." from the student names
for name in students_to_fix:
    print(name.strip("Mrs."))
```

You may notice something odd in the results:



```
Dr. Richard Scott
Dr. Jordan Carson
adeline Snyder MD
 Dylan Taylor MD
Dr. Scott Gill
iss Madison Everett
Virginia Ramirez MD
Joseph Morales III
Angela Perkins DV
Heather Allen MD
Luke Lowery MD
Dr. Larry Hine
Emily Cardenas MD
 Elizabeth Espinoza DDS
```

In the third line, the "M" has been removed from "Madeline." In the fourth line, "Mr." has been removed from "Dylan Taylor MD." In the sixth line, the "M" in "Miss" has been removed, and in the last line, the prefix "Mrs." has been removed from "Elizabeth Espinosa DDS." From these results, we can see that the `strip()` method will not work for us because it removes letters from the students' names, not just the prefix "Mrs."

NOTE

> For more information, read the **Python documentation on the strip() method** (https://docs.python.org/3/library/stdtypes.html#string-methods) .

The replace() Method

When we use the `replace()` method, we can replace an "old" phrase, or string, with a new phrase, or string, inside the parentheses.

In our case, we want to replace prefixes with an empty string, or "nothing." For example, to replace "Dr." with an empty string, we would use the following syntax: `replace("Dr.", "")`.

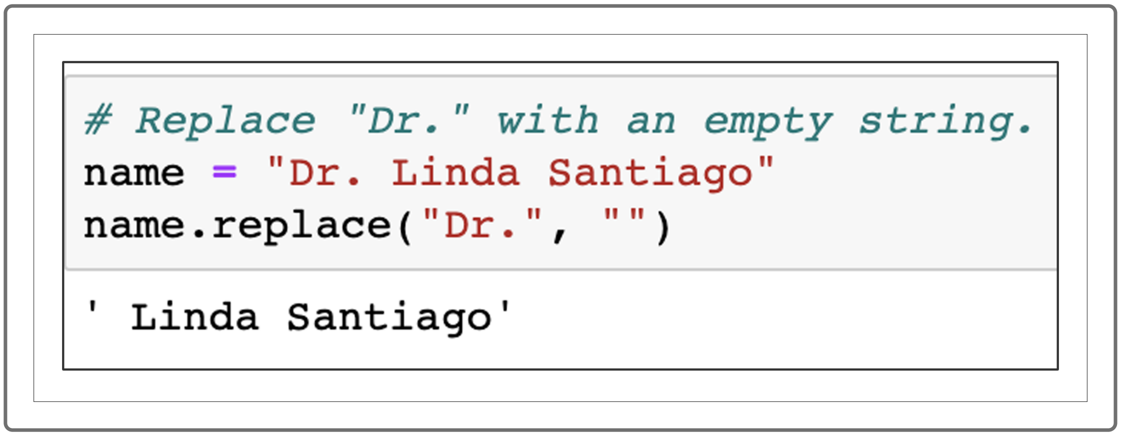
Retake

To change "Mrs. Linda Santiago" to "Dr. Linda Santiago" you would write `name.replace("Mrs.","Dr.")`.

Now let's replace "Dr." in "Dr. Linda Santiago" with an empty string. Add the following code to a new cell and run the cell.

```
# Replace "Dr." with an empty string.
name = "Dr. Linda Santiago"
name.replace("Dr.", "")
```

The output after running this cell will be the following:



```
# Replace "Dr." with an empty string.
name = "Dr. Linda Santiago"
name.replace("Dr.", "")
```

```
' Linda Santiago'
```

You may have noticed in the output that there is a white space before "Linda Santiago." We can remove this white space by adding it to "Dr." while we are using the `replace()` method, like this:

`name.replace("Dr.", "")`.

NOTE

> For more information, read the **Python documentation on the replace() method (https://docs.python.org/3/library/stdtypes.html#string-methods)**.