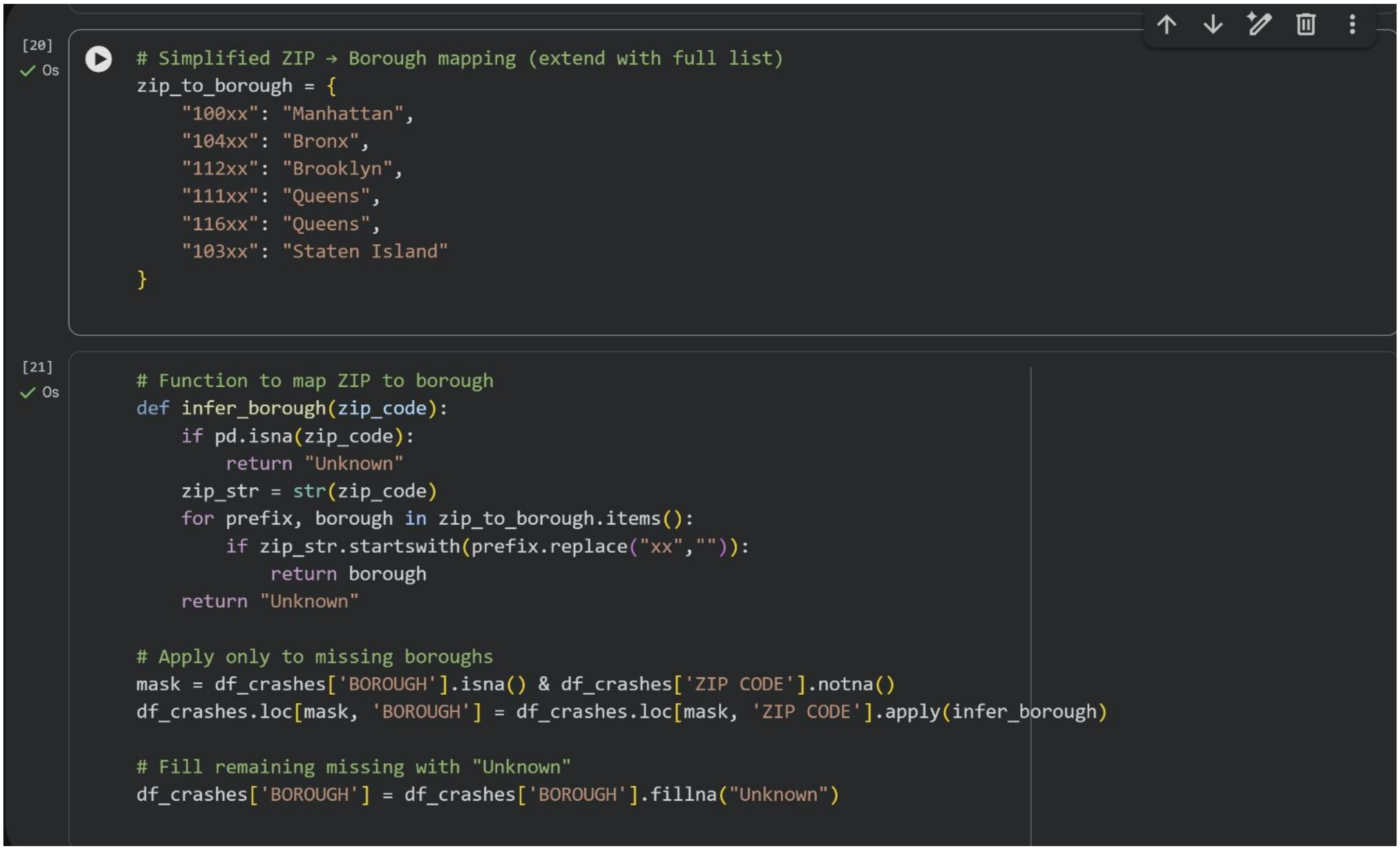
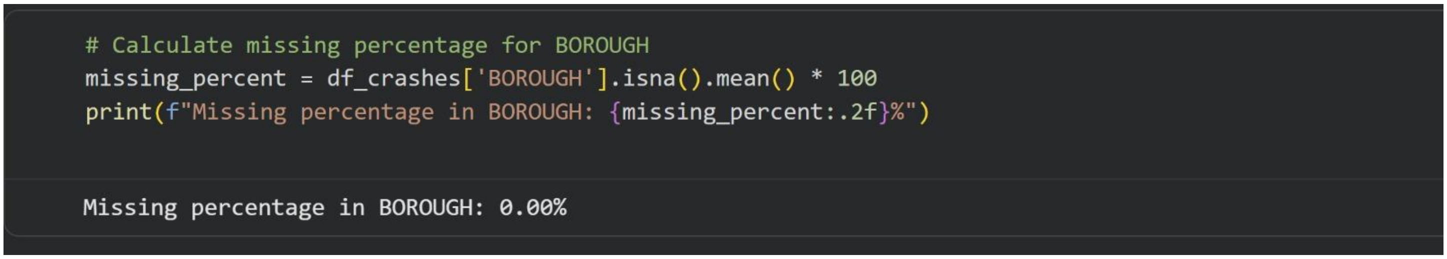
Imputation of Borough



```python
# Simplified ZIP → Borough mapping (extend with full list)
zip_to_borough = {
    "100xx": "Manhattan",
    "104xx": "Bronx",
    "112xx": "Brooklyn",
    "111xx": "Queens",
    "116xx": "Queens",
    "103xx": "Staten Island"
}
```

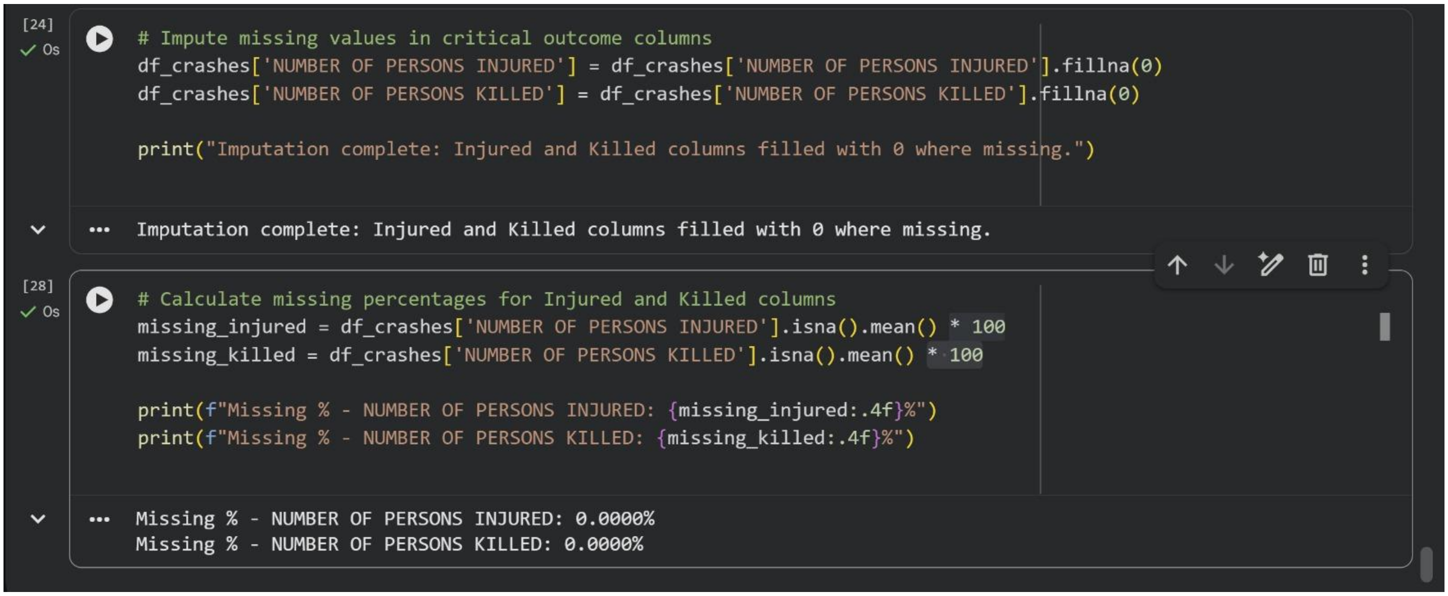
```python
# Function to map ZIP to borough
def infer_borough(zip_code):
    if pd.isna(zip_code):
        return "Unknown"
    zip_str = str(zip_code)
    for prefix, borough in zip_to_borough.items():
        if zip_str.startswith(prefix.replace("xx","")):
            return borough
    return "Unknown"

# Apply only to missing boroughs
mask = df_crashes['BOROUGH'].isna() & df_crashes['ZIP CODE'].notna()
df_crashes.loc[mask, 'BOROUGH'] = df_crashes.loc[mask, 'ZIP CODE'].apply(infer_borough)

# Fill remaining missing with "Unknown"
df_crashes['BOROUGH'] = df_crashes['BOROUGH'].fillna("Unknown")
```



```python
# Calculate missing percentage for BOROUGH
missing_percent = df_crashes['BOROUGH'].isna().mean() * 100
print(f"Missing percentage in BOROUGH: {missing_percent:.2f}%")
```

```
Missing percentage in BOROUGH: 0.00%
```

Imputation of Number of Persons Injuried/Killed



```
[24]
# Impute missing values in critical outcome columns
df_crashes['NUMBER OF PERSONS INJURED'] = df_crashes['NUMBER OF PERSONS INJURED'].fillna(0)
df_crashes['NUMBER OF PERSONS KILLED'] = df_crashes['NUMBER OF PERSONS KILLED'].fillna(0)

print("Imputation complete: Injured and Killed columns filled with 0 where missing.")
```

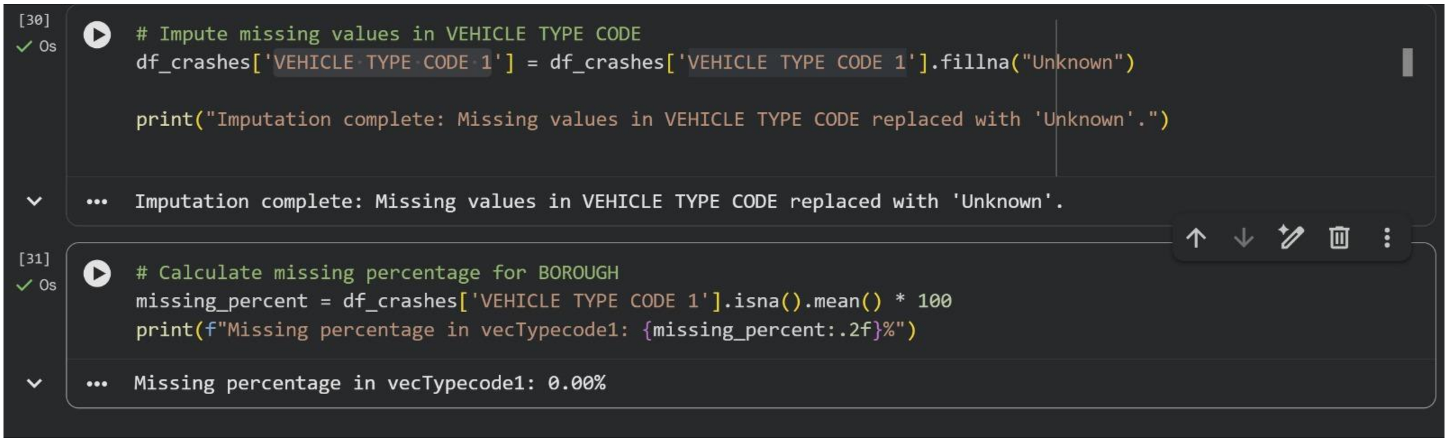
```
Imputation complete: Injured and Killed columns filled with 0 where missing.
```

```
[28]
# Calculate missing percentages for Injured and Killed columns
missing_injured = df_crashes['NUMBER OF PERSONS INJURED'].isna().mean() * 100
missing_killed = df_crashes['NUMBER OF PERSONS KILLED'].isna().mean() * 100

print(f"Missing % - NUMBER OF PERSONS INJURED: {missing_injured:.4f}%")
print(f"Missing % - NUMBER OF PERSONS KILLED: {missing_killed:.4f}%")
```

```
Missing % - NUMBER OF PERSONS INJURED: 0.0000%
Missing % - NUMBER OF PERSONS KILLED: 0.0000%
```

We imputed these columns because they are critical outcome variables in crash analysis. Missing values would distort injury and fatality statistics if left as NaN. Filling with 0 is logical since unreported cases reasonably mean no injuries or deaths occurred.

Imputation of VEHICLE TYPE CODE 1



```python
# Impute missing values in VEHICLE TYPE CODE
df_crashes['VEHICLE TYPE CODE 1'] = df_crashes['VEHICLE TYPE CODE 1'].fillna("Unknown")

print("Imputation complete: Missing values in VEHICLE TYPE CODE replaced with 'Unknown'.")
```

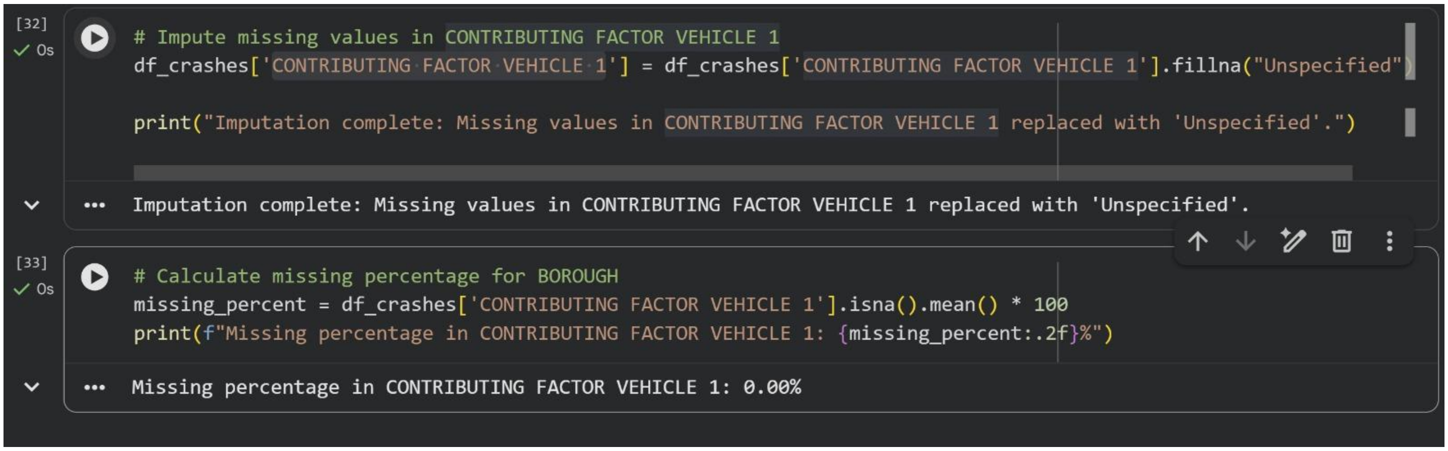
```
Imputation complete: Missing values in VEHICLE TYPE CODE replaced with 'Unknown'.
```

```python
# Calculate missing percentage for BOROUGH
missing_percent = df_crashes['VEHICLE TYPE CODE 1'].isna().mean() * 100
print(f"Missing percentage in vecTypecode1: {missing_percent:.2f}%")
```

```
Missing percentage in vecTypecode1: 0.00%
```

We imputed VEHICLE TYPE CODE 1 to avoid missing values that break consistency in crash records. Unreported vehicle types are treated as "Unknown" instead of leaving them blank. This ensures analysis and aggregation can proceed without bias or dropped rows.

Imputation of Contributing Factor Vehicle 1



```python
# Impute missing values in CONTRIBUTING FACTOR VEHICLE 1
df_crashes['CONTRIBUTING FACTOR VEHICLE 1'] = df_crashes['CONTRIBUTING FACTOR VEHICLE 1'].fillna("Unspecified")

print("Imputation complete: Missing values in CONTRIBUTING FACTOR VEHICLE 1 replaced with 'Unspecified'.")
```

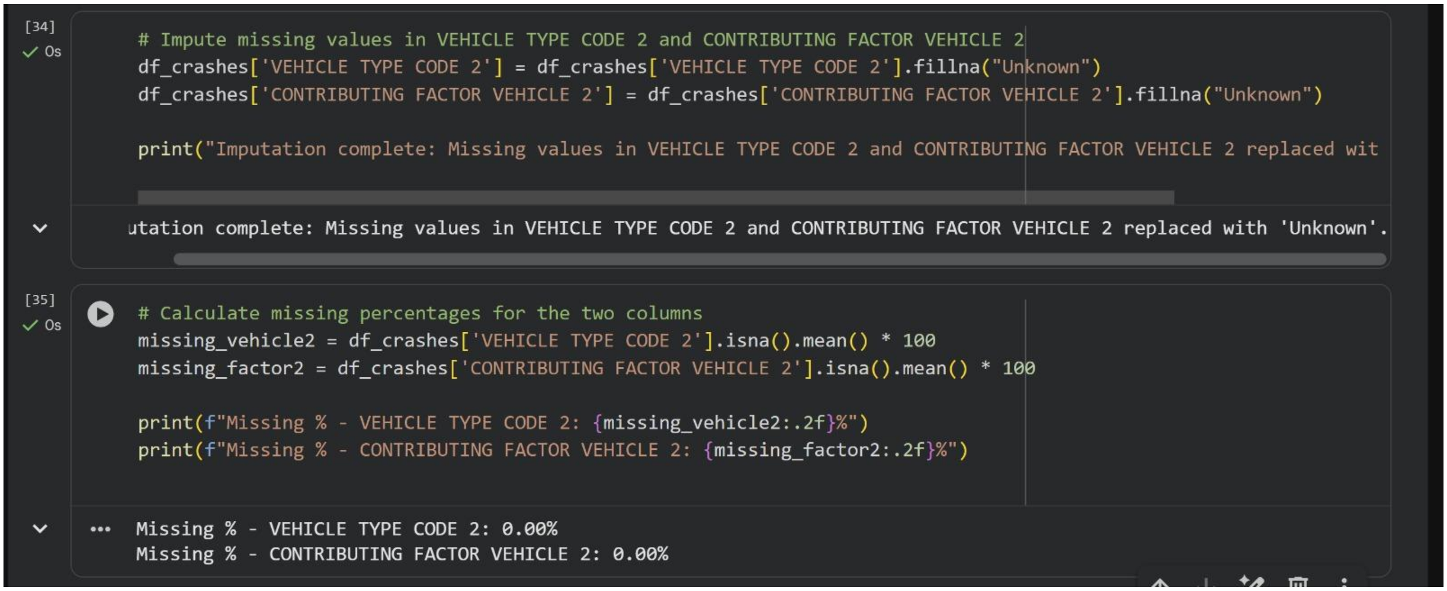
```
Imputation complete: Missing values in CONTRIBUTING FACTOR VEHICLE 1 replaced with 'Unspecified'.
```

```python
# Calculate missing percentage for BOROUGH
missing_percent = df_crashes['CONTRIBUTING FACTOR VEHICLE 1'].isna().mean() * 100
print(f"Missing percentage in CONTRIBUTING FACTOR VEHICLE 1: {missing_percent:.2f}%")
```

```
Missing percentage in CONTRIBUTING FACTOR VEHICLE 1: 0.00%
```

Filling with "Unspecified" preserves completeness while clearly marking unknown factors.

Imputation of Vehicle type code 2 & contributing factor vehicle 2



[34]

```
# Impute missing values in VEHICLE TYPE CODE 2 and CONTRIBUTING FACTOR VEHICLE 2
df_crashes['VEHICLE TYPE CODE 2'] = df_crashes['VEHICLE TYPE CODE 2'].fillna("Unknown")
df_crashes['CONTRIBUTING FACTOR VEHICLE 2'] = df_crashes['CONTRIBUTING FACTOR VEHICLE 2'].fillna("Unknown")

print("Imputation complete: Missing values in VEHICLE TYPE CODE 2 and CONTRIBUTING FACTOR VEHICLE 2 replaced wit
```

```
utation complete: Missing values in VEHICLE TYPE CODE 2 and CONTRIBUTING FACTOR VEHICLE 2 replaced with 'Unknown'.
```

[35]

```
# Calculate missing percentages for the two columns
missing_vehicle2 = df_crashes['VEHICLE TYPE CODE 2'].isna().mean() * 100
missing_factor2 = df_crashes['CONTRIBUTING FACTOR VEHICLE 2'].isna().mean() * 100

print(f"Missing % - VEHICLE TYPE CODE 2: {missing_vehicle2:.2f}%")
print(f"Missing % - CONTRIBUTING FACTOR VEHICLE 2: {missing_factor2:.2f}%")
```

```
Missing % - VEHICLE TYPE CODE 2: 0.00%
Missing % - CONTRIBUTING FACTOR VEHICLE 2: 0.00%
```

values would weaken analysis of secondary vehicles and their crash causes. Filling with "Unknown" ensures consistency while clearly marking unreported information.