[1] LOAD DATASET HotelBookings AS csv FROM hotel_bookings.csv @ artifact file 93

Console Timing Datasets Charts

hotelbookings (**119390** rows)

Views

	hotel (string)	is_canceled (boolean)	lead_time (short)	arrival_date_year (date)	arrival_date_month (string)	arrival_date_week_number (short)	arrival_date_day_of_month
0	Resort Hotel	false	342	2015-00-01	July	27	1
1	Resort Hotel	false	737	2015-00-01	July	27	1
2	Resort Hotel	false	7	2015-00-01	July	27	1
3	Resort Hotel	false	13	2015-00-01	July	27	1
4	Resort Hotel	false	14	2015-00-01	July	27	1
5	Resort Hotel	false	14	2015-00-01	July	27	1
6	Resort Hotel	false	0	2015-00-01	July	27	1
7	Resort Hotel	false	9	2015-00-01	July	27	1
8	Resort Hotel	true	85	2015-00-01	July	27	1
9	Resort Hotel	true	75	2015-00-01	July	27	1
10	Resort Hotel	true	23	2015-00-01	July	27	1
11	Resort Hotel	false	35	2015-00-01	July	27	1
12	Resort Hotel	false	68	2015-00-01	July	27	1
13	Resort Hotel	false	18	2015-00-01	July	27	1
14	Resort Hotel	false	37	2015-00-01	July	27	1
15	Resort Hotel	false	68	2015-00-01	July	27	1
16	Resort Hotel	false	37	2015-00-01	July	27	1
17	Resort Hotel	false	12	2015-00-01	July	27	1
18	Resort Hotel	false	0	2015-00-01	July	27	1
19	Resort Hotel	false	7	2015-00-01	July	27	1

[2]

```
missing_values =["Undefined"]
ds = vizierdb.get_data_frame("hotelbookings")

print(ds.shape)

print("Null Values by Column")
print(ds.isna().sum())


import pandas as pd

# Create a dictionary of arrival date data
hotel_data = {
    "arrival_date_year": [2023, 2023, 2023, 2024, 2022],
    "arrival_date_month": [10, 11, 12, 1, 3],
    "arrival_date_day_of_month": [1, 5, 25, 10, 15]
}

# Create a pandas DataFrame from the dictionary
ds = pd.DataFrame(hotel_data)

# Convert arrival date to datetime format
ds['arrival_date'] = pd.to_datetime(ds['arrival_date_year'].astype(str) + '/' + ds['arrival_date_month'].astype(str) + '/' + ds['arriv

print(ds['arrival_date'])

import pandas as pd
import numpy as np
```

```python
data = {
    "arrival_date_year": [2023, 2022, 2021],
    "arrival_date_month": [10, 12, 8],
    "arrival_date_day_of_month": [4, None, 12],  # Example column with missing values (use NaN or None)
    "other_column": [5, 6, 7]  # Other columns...
}

hotel = pd.DataFrame(data)

# Calculate the number of missing values in each column
missing_values_per_column = np.sum(hotel.isnull())
print("Number of missing values per column:")
print(missing_values_per_column)
```

Console Timing Datasets Charts

```
(119390, 32)
Null Values by Column
hotel                                0
is_canceled                          0
lead_time                            0
arrival_date_year                    0
arrival_date_month                   0
arrival_date_week_number             0
arrival_date_day_of_month            0
stays_in_weekend_nights              0
stays_in_week_nights                 0
adults                               0
children                             4
babies                               0
meal                                 0
country                              0
market_segment                       0
distribution_channel                 0
is_repeated_guest                    0
previous_cancellations               0
previous_bookings_not_canceled       0
reserved_room_type                   0
assigned_room_type                   0
booking_changes                      0
deposit_type                         0
agent                            16340
company                              0
days_in_waiting_list                 0
customer_type                        0
adr                                  0
required_car_parking_spaces          0
total_of_special_requests            0
reservation_status                   0
reservation_status_date              0
dtype: int64
```

[3]

```python
import numpy as np
ds = vizierdb.get_data_frame("hotelbookings")
missing_values_per_column = np.sum(ds.isnull())
print("Number of missing values per column:")
print(missing_values_per_column)
import pandas as pd

nan_indices = ds[ds['children'].isnull()].index.tolist()

# Display the indices
print("Indices where 'children' column contains NaN values:")
print(nan_indices)

total_rows = ds.shape[0]

for col in ds.columns:
    missing_count = ds[col].isnull().sum()
    if missing_count > (total_rows * 0.7):
        ds.drop(columns=col, inplace=True)

print(ds.shape)

ds.drop(columns=["arrival_date_week_number", "arrival_date_year", "arrival_date_month", "arrival_date_day_of_month"],
           inplace=True, axis=1)

print(ds.shape)

ds.dropna(subset=["agent"], inplace=True)
print(ds.shape)
```

```python
# Fill missing values in 'children' column with the mean
ds["children"].fillna(value=ds["children"].mean(), inplace=True)

# Convert 'children' column values to their floor values
ds["children"] = ds["children"].apply(np.floor)

# Count missing values in 'children' column after filling
missing_children = np.sum(ds['children'].isnull())
print(f"Total missing values in 'children' column after filling: {missing_children}")


columns_to_fill = ['market_segment', 'distribution_channel', 'meal', 'country']

for column in columns_to_fill:
    ds[column].fillna(method='bfill', inplace=True)

print("Number of missing values after backward fill:")
for column in columns_to_fill:
    print(f"{column}: {np.sum(ds[column].isnull())}")

missing_values_per_column = np.sum(ds.isnull())
print("Total missing values in each column:")
print(missing_values_per_column)
```

Console Timing Datasets Charts

```
Number of missing values per column:
hotel                                 0
is_canceled                           0
lead_time                             0
arrival_date_year                     0
arrival_date_month                    0
arrival_date_week_number              0
arrival_date_day_of_month             0
stays_in_weekend_nights               0
stays_in_week_nights                  0
adults                                0
children                              4
babies                                0
meal                                  0
country                               0
market_segment                        0
distribution_channel                  0
is_repeated_guest                     0
previous_cancellations                0
previous_bookings_not_canceled        0
reserved_room_type                    0
assigned_room_type                    0
booking_changes                       0
deposit_type                          0
agent                             16340
company                               0
days_in_waiting_list                  0
customer_type                         0
adr                                   0
required_car_parking_spaces           0
total_of_special_requests             0
reservation_status                    0
reservation_status_date               0
```

```
Indices where 'children' column contains NaN values:
[40600, 40667, 40679, 41160]
(119390, 32)
(119390, 28)
(103050, 28)
Total missing values in 'children' column after filling: 0
Number of missing values after backward fill:
market_segment: 0
distribution_channel: 0
meal: 0
country: 0
Total missing values in each column:
hotel                             0
is_canceled                       0
lead_time                         0
stays_in_weekend_nights           0
stays_in_week_nights              0
adults                            0
children                          0
babies                            0
meal                              0
country                           0
market_segment                    0
distribution_channel              0
is_repeated_guest                 0
previous_cancellations            0
previous_bookings_not_canceled    0
reserved_room_type                0
assigned_room_type                0
booking_changes                   0
deposit_type                      0
agent                             0
company                           0
days_in_waiting_list              0
customer_type                     0
```