penguin_intel_report

April 11, 2025

[13]:
```
'''
1.        Load the Dataset
        •       Use seaborn.load_dataset('penguins')
        •       Assign it to a DataFrame called penguins_df
2.        Basic Exploration
        •       Display:
        •       .head()
        •       .info()
        •       .describe()
'''
import seaborn as sns
import pandas as pd

penguins_df = sns.load_dataset("penguins")
penguins_df.head()
# Display the first few rows of the DataFrame

penguins_df.info()
# Display information about the DataFrame, including data types and non-null␣
 ↪counts

penguins_df.describe()
# Display summary statistics of the DataFrame
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 344 entries, 0 to 343
Data columns (total 7 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   species            344 non-null    object
 1   island             344 non-null    object
 2   bill_length_mm     342 non-null    float64
 3   bill_depth_mm      342 non-null    float64
 4   flipper_length_mm  342 non-null    float64
 5   body_mass_g        342 non-null    float64
 6   sex                333 non-null    object
dtypes: float64(4), object(3)
memory usage: 18.9+ KB
```

[13]:

	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g
count	342.000000	342.000000	342.000000	342.000000
mean	43.921930	17.151170	200.915205	4201.754386
std	5.459584	1.974793	14.061714	801.954536
min	32.100000	13.100000	172.000000	2700.000000
25%	39.225000	15.600000	190.000000	3550.000000
50%	44.450000	17.300000	197.000000	4050.000000
75%	48.500000	18.700000	213.000000	4750.000000
max	59.600000	21.500000	231.000000	6300.000000

[12]:
```
'''
3.        Write Custom Summary Functions
        •       A function get_column_stats(df, column_name) that returns a␣
 ↪dictionary with:
        •       Min
        •       Max
        •       Mean
        •       Count of missing values
'''

def get_column_stats(df, column_name):
    """
    Returns a dictionary with statistics for a specified column in the␣
 ↪DataFrame.

    Parameters:
        df (pd.DataFrame): The DataFrame containing the data.
        column_name (str): The name of the column to analyze.

    Returns:
        dict: A dictionary with Min, Max, Mean, and Count of missing values.
    """
    stats = {
        "Min": df[column_name].min(),
        "Max": df[column_name].max(),
        "Mean": df[column_name].mean(),
        "Missing Values": df[column_name].isnull().sum()
    }
    return stats
```

[10]:
```
# 4. Investigate a Pattern
#       • Find the heaviest penguin and print its species and island.
#       • Compare average body mass between different species and sex.

# Find the heaviest penguin
heaviest_penguin = penguins_df.loc[penguins_df['body_mass_g'].idxmax()]
```

```
print(f"Heaviest Penguin: Species - {heaviest_penguin['species']}, Island -␣
 ↪{heaviest_penguin['island']}")

# Compare average body mass between different species and sex
average_body_mass = penguins_df.groupby(['species', 'sex'])['body_mass_g'].
 ↪mean()
print("Average Body Mass by Species and Sex:")
print(average_body_mass)
```

```
Heaviest Penguin: Species - Gentoo, Island - Biscoe
Average Body Mass by Species and Sex:
species    sex
Adelie     Female    3368.835616
           Male      4043.493151
Chinstrap  Female    3527.205882
           Male      3938.970588
Gentoo     Female    4679.741379
           Male      5484.836066
Name: body_mass_g, dtype: float64
```

0.0.1 Observations

- The heaviest penguin belongs to the Gentoo species, found on Biscoe Island, with a body mass of 6300 grams. This indicates that Gentoo penguins tend to have a higher body mass compared to other species.
- The average body mass varies significantly between species and sexes. For example, Gentoo males have the highest average body mass (5484.84 grams), while Adelie females have the lowest (3368.84 grams).
- There are missing values in the dataset, particularly in columns like `bill_length_mm`, `bill_depth_mm`, `flipper_length_mm`, `body_mass_g`, and `sex`. This could affect the analysis and may require data cleaning or imputation.