

Data Loading & Initial Inspection

- Step: Load SpaceX Dataset
 - Key Phrase: `df=pd.read_csv("dataset_part_1.csv")`
- Step: Identify & Calculate Percentage of Missing Values
 - Key Phrase: `df.isnull().sum()/len(df)*100`
 - Callout: Focus on `LandingPod` (28.89% missing)
- Step: Identify Column Data Types
 - Key Phrase: `df.dtypes`
 - Callout: Differentiate Numerical (e.g., `int64`, `float64`) and Categorical (e.g., `object`, `bool`)



Data Loading & Initial Inspection

- Step: Load SpaceX Dataset
 - Key Phrase: `df=pd.read_csv("dataset_part_1.csv")`
- Step: Identify & Calculate Percentage of Missing Values
 - Key Phrase: `df.isnull().sum()/len(df)*100`
 - Callout: Focus on `LandingPod` (28.89% missing)
- Step: Identify Column Data Types
 - Key Phrase: `df.dtypes`
 - Callout: Differentiate Numerical (e.g., `int64`, `float64`) and Categorical (e.g., `object`, `bool`)



Task 3: Determine Number & Occurrence of Mission Outcomes

- Key Phrase: `landing_outcomes = df['Outcome'].value_counts()`
- Callout: Outcome Types: `True ASDS`, `None None`, `True RTLS`, `False ASDS`, `True Ocean`, `False Ocean`, `None ASDS`, `False RTLS`
- Step: Define 'Bad Outcomes' Set
 - Key Phrase: `bad_outcomes = set(landing_outcomes.keys()[1,3,5,6,7])`
 - Callout: Examples: `{'False ASDS', 'None None', 'False Ocean', 'None ASDS', 'False RTLS'}`



Data Analysis Tasks

- Task 1: Calculate Number of Launches per Site
 - Key Phrase: `df['LaunchSite'].value_counts()`
 - Callout: Launch Sites: `CCAFS SLC 40`, `KSC LC 39A`, `VAFB SLC 4E`
- Task 2: Calculate Number & Occurrence of Each Orbit Type
 - Key Phrase: `df['Orbit'].value_counts()`
 - Callout: Orbit Types: `GTO`, `ISS`, `VLEO`, `PO`, `LEO`, `SSO`, `MEO`, `HEO`, `ES-L1`, `SO`, `GEO`



Training Label Creation

- Task 4: Create 'Class' Column (Landing Outcome Label)
 - Process: Iterate through `Outcome` column
 - Decision: Is current `Outcome` in `bad_outcomes` set?
 - If YES: Append `0` (Unsuccessful Landing) to `landing_class`
 - If NO: Append `1` (Successful Landing) to `landing_class`
 - Key Phrase: `df['Class'] = landing_class`
- Step: Calculate Overall Success Rate
 - Key Phrase: `df["Class"].mean()`



End

- Step: Export Processed Data to CSV
 - Key Phrase: `df.to_csv("dataset_part_2.csv", index=False)`