Module End Activities

1. Titanic Data Manipulation and Analysis

- Link: kaggle.com/c/titanic
- Description: Contains information about passengers on the Titanic, including survival status, age, sex, ticket class, fare, and more.

Data Manipulation with NumPy:

a. Loading and Viewing Data:
   - Load the dataset into a pandas DataFrame using pd.read_csv().
   - View the first few rows using head().
   - Check the shape of the DataFrame using shape.
   - Get descriptive information about the data using info().

b. Handling Missing Values:
   - Check for missing values using isnull() or isna().
   - Handle missing values using appropriate techniques (e.g., fill with mean, median, mode, or drop rows/columns).

c. Data Cleaning:
   - Correct any inconsistencies or errors in the data.
   - Convert data types if necessary (e.g., change strings to numerical values).

2. Name: House Prices Analysis

- Link: kaggle.com/c/house-prices-advanced-regression-techniques: kaggle.com/c/house-prices-advanced-regression-techniques
- Description: Contains information about house sales in Ames, Iowa, including features like sale price, lot area, number of rooms, year built, and more.

Data Manipulation with NumPy:

1. Loading Data:
   - Load the house prices dataset into a NumPy array using np.loadtxt() or np.genfromtxt().
2. Exploring Data:
   - Check the shape of the array to see the number of rows (houses) and columns (features).
   - Print the minimum and maximum values for the 'SalePrice' column using np.min() and np.max().
3. Handling Missing Values:
   - Identify missing values (NaNs) using np.isnan().
   - Replace missing values in the 'LotFrontage' column with the mean value of that column using np.nan_to_num() and np.mean().
4. Feature Normalization:
   - Normalize the 'GrLivArea' column (living area) to have a mean of 0 and a standard deviation of 1 using np.mean(), np.std(), and element-wise operations.
5. Feature Selection:
   - Select the columns 'SalePrice', 'OverallQual' (overall quality), and 'YearBuilt' using array slicing or indexing.
6. Calculate summary statistics (mean, median, standard deviation, quartiles) for numerical columns using np.mean(), np.median(), np.std(), and np.quantile().
7. Filter rows based on conditions (e.g., select houses with 'OverallQual' > 7).
8. Sort the data based on 'SalePrice' in descending order using np.argsort().
9. Apply mathematical operations to entire columns or subsets of data (e.g., calculate the square footage by multiplying 'LotArea' and 'LotFrontage').