

In [1]:

```
import numpy as np

def impute_mean(data, column):
    mean = np.mean(data[column])
    data[column].fillna(value=mean, inplace=True)
    return data

def impute_median(data, column):
    median = np.median(data[column])
    data[column].fillna(value=median, inplace=True)
    return data

def impute_mode(data, column):
    mode = data[column].mode().values[0]
    data[column].fillna(value=mode, inplace=True)
    return data
```

In [2]:

```
import random

def impute_random(data, column):
    random_sample = data[column].dropna().sample(data[column].isna().sum(),
                                                  random_state=0)
    random_sample.index = data[data[column].isna()].index
    data[column].fillna(value=random_sample, inplace=True)
    return data
```

In [3]:

```
import numpy as np
from sklearn.impute import KNNImputer

def impute_knn(data, column):
    knn_imputer = KNNImputer(n_neighbors=2)
    data[column] = knn_imputer.fit_transform(data[column].values.reshape(-1, 1))
    return data
```

In [4]:

```
import pandas as pd
import numpy as np
from sklearn.linear_model import LinearRegression
from sklearn.impute import SimpleImputer

def regression_imputation(df, target_column):
    # Split the data set into features (X) and target (y)
    X = df.drop(target_column, axis=1)
    y = df[target_column]

    # Impute missing values in the feature data using mean imputation
    mean_imputer = SimpleImputer(strategy="mean")
    X_imputed = mean_imputer.fit_transform(X)

    # Train a Linear regression model on the imputed feature data
    regression_model = LinearRegression().fit(X_imputed, y)

    # Use the regression model to impute missing values in the target data
    y_imputed = regression_model.predict(X_imputed)
    y_imputed = np.where(np.isnan(y), y_imputed, y)

    # Replace the original target data with the imputed target data
    df[target_column] = y_imputed

    return df
```

In [5]:

```
import pandas as pd
import numpy as np
from sklearn.linear_model import LinearRegression
from sklearn.impute import SimpleImputer
from sklearn.ensemble import RandomForestRegressor
from statsmodels.imputation.mice import MICEData

def multiple_imputation(df):
    # Impute missing values in the data using multiple imputation
    mice = MICEData(df)
    df_imputed = mice.data.copy()

    return df_imputed
```