

**Bài 1.** Hoàn thành python class cho việc chuẩn hoá dữ liệu cho mean bằng 0 và standard deviation bằng 1. Với API được mô tả như hình sau.

```
class StandardScaler:
    def __init__(self):
        self.mean = None
        self.std = None

    def fit(self, X):
        # Enter your code here
        return

    def transform(self, X):
        X_tran = "Enter your code herre"
        return X_tran

    def inver_transform(self, X_norm):
        """
        Chuyển ngược từ các features đã được chuẩn hoá đầu
        về features ban
        """
        X = "Enter your code he"
        return X

    def save(self, filename):
        """Save ('serialize') fitted scaler"""

    def load(self, filename):
        """Load ('Deserialize') fitted scaler"""
```

Hoàn thành module với tên là ***'ho\_va\_ten\_standardlization.py'***, và ***nộp lên google classroom***

**Bài 2.** Hoàn thành python class thực hiện linear regression trên một tập dữ liệu bất kỳ với API ược mô tả như sau

```
class LinearRegressionGD:
    def __init__(self):
        self.weights = None
        # For storing the training history:
        # list of values of Loss on training process
        self.train_history = Nonetraining

    def train(self, X, y, n_epoch=50, learning_rate=0.01):
        self.train_history = "For update history in training process"
        self.weights = "Final weights by gradient descent algorithm"

    def predict(self, X):
        y_hat = "Enter your code here"
        return y_hat

    def save(self, filename):
        """Save ('serialize') fitted regressor"""

    def load(self, filename):
        """Load ('Deserialize') fitted regressor"""
```

Hoàn thành module với tên là ***'ho\_va\_ten\_linearregression.py'***, và ***nộp lên google classroom***

**Bài 3.** Viết chương trình ***python lưu vào file ipython notebook 'bavn\_3\_ho\_va\_ten.ipynb'*** (vd: ***'bavn\_3\_pham\_tien\_lam.ipynb'***) và ***xuất ra file pdf (Nộp lại cả 2 file)*** với kết quả thực hiện các việc sau thực hiện các thao tác sau:

1. Import hàm `load_boston` từ thư viện `sklearn` như sau `'from sklearn.datasets import load_boston'`
2. Load data bằng lệnh `'data = load_boston()'`
3. In ra mô tả tập dữ liệu bằng lệnh. `'print(data.DESCR)'`
4. Đọc dữ liệu về thông tin của các ngôi nhà và giá bằng lệnh `'X = data.data'` và `'y=data.target'`.
5. Tính correlation matrix của X, y và in ra
6. Vẽ đồ thị dạng heat map của correlation matrix
7. Sử dụng `StandardScaler` trong bài 1 để chuẩn hoá dữ liệu
8. Sử dụng dữ liệu đã chuẩn hoá và `LinearRegressionGD` trong bài 2 để lập mô hình dự đoán giá nhà ở Boston.