

BÀI TẬP ÔN TẬP

A. Hướng đối tượng:

Viết chương trình quản lý sinh viên trong Python. Mỗi đối tượng sinh viên có các thuộc tính sau: id, tên, giới tính, tuổi, điểm toán, điểm lý, điểm hóa, điểm trung bình và học lực.

Id là mã sinh viên tự động tăng.

Điểm trung bình là giá trị trung bình của 3 môn toán, lý và hóa.

Học lực được tính như sau:

- Giỏi: nếu điểm trung bình lớn hơn hoặc bằng 8.
- Khá: nếu điểm trung bình nhỏ hơn 8 và lớn hơn hoặc bằng 6.5.
- Trung Bình: nếu điểm trung bình nhỏ hơn 6.5 và lớn hơn hoặc bằng 5.
- Yếu: nếu điểm trung bình nhỏ hơn 5.

Yêu cầu: tạo ra một menu với các chức năng sau:

1. Thêm sinh viên.
2. Cập nhật thông tin sinh viên bởi ID.
3. Xóa sinh viên bởi ID.
4. Tìm kiếm sinh viên theo tên.
5. Sắp xếp sinh viên theo điểm trung bình (GPA).
6. Sắp xếp sinh viên theo tên.
6. Sắp xếp sinh viên theo ID.
8. Hiện thị danh sách sinh viên.

B. Thao tác xử lý dữ liệu

This data was originally a part of UCI Machine Learning Repository and has been removed now. This data also ships with the scikit-learn library. There are 506 samples and 13 feature variables in this data-set. The objective is to predict the value of prices of the house using the given features.

#

The description of all the features is given below:

CRIM: Per capita crime rate by town

ZN: Proportion of residential land zoned for lots over 25,000 sq. ft

INDUS: Proportion of non-retail business acres per town

CHAS: Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)

NOX: Nitric oxide concentration (parts per 10 million)

RM: Average number of rooms per dwelling

AGE: Proportion of owner-occupied units built prior to 1940

DIS: Weighted distances to five Boston employment centers

RAD: Index of accessibility to radial highways

TAX: Full-value property tax rate per \$10,000

B: $1000(B_k - 0.63)^2$, where B_k is the proportion of [people of African American descent] by town

LSTAT: Percentage of lower status of the population

MEDV: Median value of owner-occupied homes in \$1000s

Yêu cầu:

1. Đọc dữ liệu và lấy 5 dòng đầu tiên
2. Thống kê các cột chứa giá trị null
3. Tạo bảng thống kê để về dữ liệu
4. Truy cập theo slice index: Theo cách này chúng ta chỉ cần truyền vào index của dòng và cột và sử dụng hàm `df.iloc[rows_slice, columns_slice]` để trích xuất ra các dòng và cột tương ứng. Cách lấy slice cho rows và columns hoàn toàn tương tự như truy cập slice index trong list. Lấy 5 dòng, 5 cột đầu tiên
5. Lựa chọn 5 dòng từ 5:10 và 2 cột từ 2:4
6. Truy cập các dòng theo row index của dataframe thông qua câu lệnh `df.loc[]`.
7. Truy cập theo column names: Đây là cách được sử dụng phổ biến vì nó tường minh hơn. Theo cách này chúng ta sẽ truy cập các trường của dataframe bằng cách khai báo list `column_names` của chúng. Ví dụ bên dưới chúng ta cần lấy ra các trường ['crim', 'tax', 'rad'] từ bảng dữ liệu.
8. # Kết hợp cả hai cách: Chúng ta có thể truy cập dataframe bằng cách kết hợp cả hai cách theo hướng sử dụng column names đối với cột và slice index đối với dòng. Lấy ra các dòng từ 10:15 của các trường 'crim', 'tax', 'rad'

9. #Lọc ra các thị trấn mà có số phòng ở trung bình trên căn hộ là trên 7
10. Lọc ra các thị trấn mà có số phòng ở trung bình trên căn hộ là trên 7
Và lọc thêm điều kiện thuế suất trên 250
11. In ra # Min, max, mean, meadian, sum của Tax
12. In ra # Min, max, mean, meadian, sum theo từng chas
13. Vẽ biểu đồ phân bố dữ liệu