データベース演習 （Advanced Database Exercise）

４．Pythonのデータフレーム，集計・集約，ソート

URL: http://www.kkaneko.jp/cc/dbenshu/index.html

**概要 Abstract**

この演習では、**Python でのデータ処理機能**を演習する。MySQL Employees Sample データベースを使用する

　Today's class is python's data processing function. MySQL Exmployees Sample is used as a sample dataset.

**■　Python の pandas パッケージ (Python pandas パッケージ)**

Python の pandas パッケージは、次の機能を持つ．(Python's pandas package has the followin functions)

　・データフレームと呼ばれる表形式データの処理 (DataFrame data structure is a table style data structure)

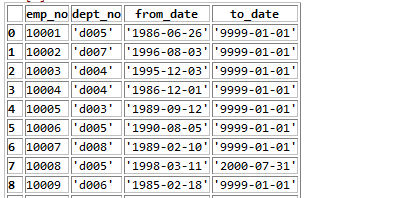
　・時系列データの処理 (Time series data processin)

　・集計、集約 (grouping, aggregating)

　・CSVファイル、Excel2003 ファイルなどの読み書き機能 (Input/Output data files, CSV, Excel2003 etc)

　・リレーショナルデータベースとの連携機能 (Relational database)

など (etc.)



Python のデータフレームの例

■ SQL の GLOB

SQL の GLOB は、文字列のパターンマッチを行う (SQL's GLOB is string pattern matching)

\* 任意の文字列とマッチ (any string)

? 任意の１文字とマッチ (any character)

[a-z] a,b,…,zのどれか１文字とマッチ (a, b, …, z)

　　　　など

例えば、次のSQL プログラムでは、列 A に文字列 9999 を含む行を得る

The evaluation result of the following SQL program is a set of lines that the arribute value of A contains 9999.

select \* from R where A GLOB '\*9999\*'

**演習 4 (Exercises 4)**

前準備 (preparation)

・MySQL Empolyees Sample データベース (MySQL Empolyees Sample Database)

MySQL employees sample データベースの出典と著作権表示

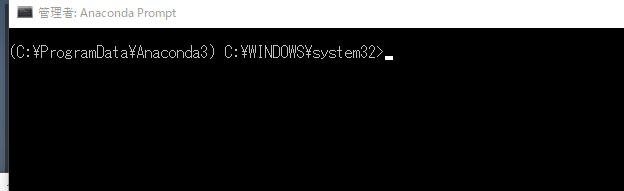
出典：MySQL employees sample database https://dev.mysql.com/doc/employee/en/   
===  
This work is licensed under the Creative Commons Attribution-Share Alike 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/3.0/> or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.  
Please contact <http://www.mysql.com/about/contact/> for more information.

**■ ステップ1 (Step 1)**

**python のパッケージをインストールする。次回以降の授業でも使う**

(Install python packages for future use)

①　スタートメニューの「**Anaconda3 (64bit)**」の下の「**Anaconda Prompt**」を管理者として実行する．



2. Anaconda Promptで，**次の Python プログラムを実行しなさい**

**pip install bottle**

**pip install numpy**

**pip install scipy**

**pip install pandas**

**pip install matplotlib**

**pip install scikit-learn**

**pip install seaborn**

　※ ここまで。上の操作をすべて実行してください。

* **ステップ2 (Step 2)**

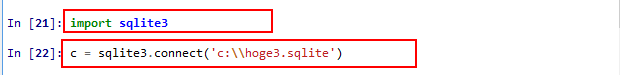
**IPython コンソール・ウインドウ**で，**次の Python プログラムを実行させてみなさい**

**SQLite3データベースに接続する**

(connect to a SQLite3 database using IPython console)

**◆** 今日の演習ではSQLite3 データベースファイル名は **c:\hoge3.sqlite**

(SQLite3 database file name is '**c:\hoge3.sqlite**')



* **ステップ３ (Step 3)**

**pandasの機能を使い、リレーショナルデータベースのテーブルと、Pythonのデータフレーム**

**(Data Frame)に読み込む**

(Read a relational table to a python data frame using pandas package)

データフレームを格納するためのオブジェクト **df** を作っている

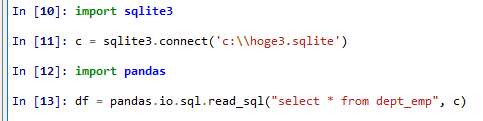
(Create a varible df to store a data frame)

**import sqlite3**

**c = sqlite3.connect('c:\\hoge3.sqlite')**

**import pandas**

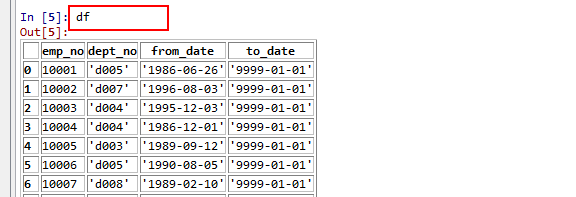
**df = pandas.io.sql.read\_sql("select \* from dept\_emp", c)**



* **ステップ４ (Step 4)**

**読み込んだデータフレームの確認と分析**

(Example the data frame)

1. 確認する (listing the data frame)

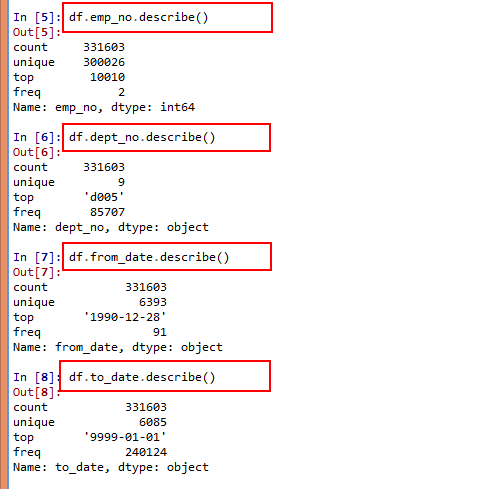
・count: 行数 (number of lines)

・unique: 異なる値の数 (the number of different values)

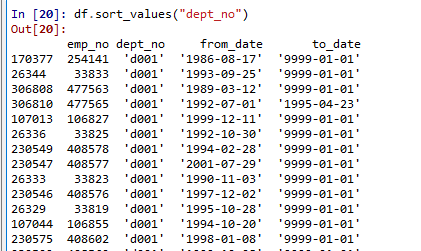
・top: 先頭行の値

・freq: 同じ値が最大何回あるか

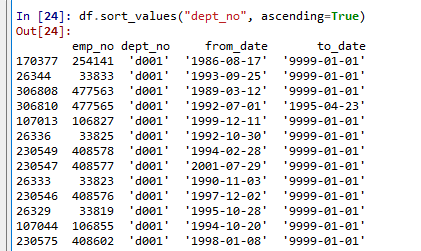
(the maximum number of the same values)

2. 集計

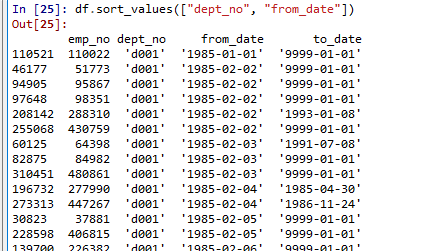
3. ソート



　 昇順 (ascending)

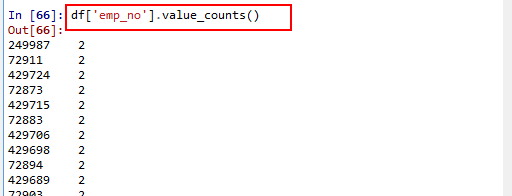


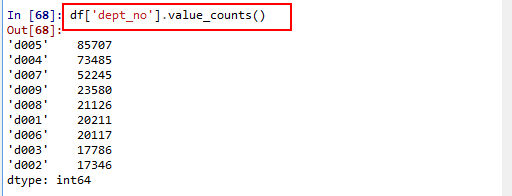
降順 (descnding)

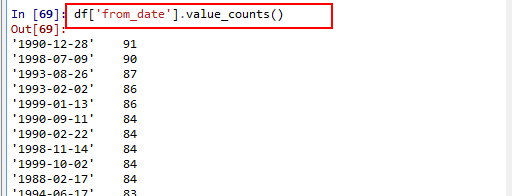


多値 (multi attribute)

4. 頻度分布 histgramming







5. 集約 grouping

