Sql Query Equivalent in Python/Panads And R

Huseyin Onder Bektas
Data Scientist at DenizBank

March 3, 2019

Introduction

- If you are familiar with the Sql, I believe the basic way to learn pandas or R is to compered it to sql query.
- In this presentation, I give the basic sql query equivalent for pandas and R.
- Let's us assume a table called "Personal" in the database and have four columns : people identity number "Id", their city "City", their income "Income" and their work definition "Work"
- I give all of the example based on these parameters and equivalent.

Select Statement

Select statement:

Sql Query

```
Select Top 10 * From Personel
```

Pandas Equivalent

```
Personal.head(10)
```

R Equivalent

```
head(Personal, n=10L)
```

Selection Columns

Selection Columns,

Sql Query

```
Select Top 10 Id, City
From Personel
```

Pandas Equivalent

```
Personal[['Id', 'City']].head(10)
```

R Equivalent

```
library(dplyr)
head(select(Personal,Id,City), n=10L)
```

Select statement in R require "dplyr" library.

Ordering The Columns

Ordering The Columns,

Sql Query

```
Select Top 10 Id, City
From Personel
Order By Income ASC
```

Pandas Equivalent

```
Personal[['Id', 'City']].sort_value(by=['Income']).head(10)
```

R Equivalent

```
library(dplyr)
head(select(arrange(Personel,Income),Id,City), n=10L)
```

Select statement in R require "dplyr" library.

Filtering The Data

Filtering The Data (Where Statement in SQL)

Sql Query

```
Select Top 10 Id, City
From Personel
Where Income >= 2500
```

Pandas Equivalent

```
Personal[Personal['Income'] >= 2500][['Id','City']].head(10)
```

R Equivalent

```
library(dplyr)
head(select(filter(Personal, Income >=2500),Id,City), n=10L)
```

Grouping The Data

Grouping The Data

Sql Query

Select City, sum(Income)
From Personel
Group By City

Pandas Equivalent

Personal.groupby(['City'])['Income'].sum()

R Equivalent

library(dplyr)
Personal %>% group_by(City) %>% summarise(Income = sum(Income))

Conclusion

- There are various kind of sql queries like group by after filtering or row numbers or creating subqueries etc.
- I can't give all of the example here. However my idea is to give brief introduction to people who is not familiar R or Python/pandas.
- You can create your own queries using the basic ones.
- Rest is yours.