**Data Utilization Prework**

1. **Get Rakuten DEV-VPN access to connect to development environment.**

Required for Hands-On exercises

|  |  |
| --- | --- |
| **What is DEV-VPN?** | VPN to access to data center  <https://confluence.rakuten-it.com/confluence/display/VPN/01.+DEV-VPN> |
| **How to apply DEV-VPN access ?** | <https://confluence.rakuten-it.com/confluence/display/VPN/Request+DEV-VPN+Access> |
| **How to use DEV-VPN ?** | <https://confluence.rakuten-it.com/confluence/display/VPN/03.+How+to+use> |
| **Comment** | It takes 2 or 3 days to get permissions on your VPN account after applying (you will get email when they set permissions)  In the DEV – VPN Group, select your department group from the list. |

1. **Data Collection – Environment Setup for exercises**

**Setting up Hue**

After Connecting to DEV-VPN, you should be able to open the HUE URL below.

url: <http://ins-adseda101z.prod.hnd1.bdd.local/accounts/login/?next=/>

Note: Username and Password will be provided during training. At this point, please confirm if you can see the login page of Hue.

**Setting up Jupyter**

After Connecting to DEV-VPN, you should be able to open the JUPYTER URL below.

url: [**http://ins-adseda101z.prod.hnd1.bdd.local/jupyterhub/tree**](http://ins-adseda101z.prod.hnd1.bdd.local/jupyterhub/tree)**?**

Note: Username and Password will be provided during training. At this point, please confirm if you can see the login page of Jupyter.

1. **Cheat Sheet for Hive Exercises**

* Basic SQL skills will be helpful
* Hadoop and Hive knowledge is an advantage

1. **Query data in columns c1 and c2 from a table t in database db.**

select c1, c2 from db.t;

1. **Query all rows and columns from a table t**

select \* from t;

1. **Query n rows from a table**

select c1, c2 from t limit n;

1. **Query distinct rows from table t**

select distinct c1 from t;

1. **Query data and filter rows with condition**

select c1, c2 from t where condition;

1. **Group rows using aggregate function**

select c1, aggregate(c2) from t group by c1;

aggregate can be – count, sum, min, max

1. **Filter groups using having clause**

select c1, aggregate(c2) from t group by c1 having condition;

**Useful queries for the exercise –**

1. describe data\_utilization\_training\_db.red\_basket\_order\_tbl\_masked;
2. select \* from data\_utilization\_training\_db.red\_basket\_order\_tbl\_masked limit 10;
3. select order\_no,count(\*) from data\_utilization\_training\_db.red\_basket\_order\_tbl\_masked group by order\_no having count(\*) > 1;
4. select \* from data\_utilization\_training\_db.red\_basket\_order\_tbl\_masked where easy\_id = 279;
5. select distinct prefecture from data\_utilization\_training\_db.red\_basket\_order\_tbl\_masked;
6. select device\_cd, count(\*) from data\_utilization\_training\_db.red\_basket\_order\_tbl\_masked geoup by device\_cd;
7. select min(reg\_datetime), max(reg\_datetime) from data\_utilization\_training\_db.red\_basket\_order\_tbl\_masked;
8. **Cheat Sheet for Python Exercises**

Please refer python3\_cheatsheet.pdf for basic Python

Please refer Pandas(Python)\_cheatsheet.pdf for Basic Pandas for Data Science