MODI SHRESHTHA(190130111081)

Data Anonymization pipeline

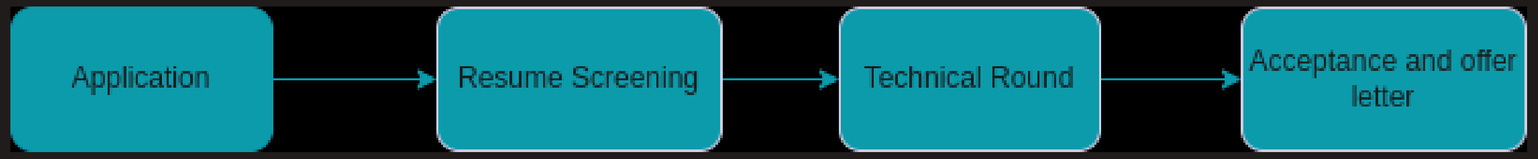
Eternal Soft Solutions ✦ GEC Gandhinagar (EC) ✦ 2023

About eternal

Eternal is a AWS partner web development based consulting company based in Ahmedabad and UK that provides cloud based product design and development solutions for businesses and startups

401 Satyam Mall Nr Mansi Circle, Satellite, Ahmedabad, Gujarat 380051

About My role



A part of the cloud solutions team where i design scalable, compact and cost optimized solutions based on the requirnments of the customer around the whole cloud service area.

working areas:

- Storage and databases
- Computing
- Data Analysis and visualization
- Data Privacy

Tech stack:

- AWS
- Python
- UNIX
- Snowflake
- Apache airflow
- GIT
- SQL

Training

01 Week 1

Linux

02 Week 3

03 Week 2

04 Week 4

aws

Training

05 Week 5

06 Week 6

07 Week 7

Project Introduction and setting up the project enviournment

08 Week 8

Configuring the arx servers and anonymizing the dummy arx data

WEEK AT A GLANCE

05 Week 9

Anonymizing the dummy data

06 Week 11

Automating the etl workflow using airflow

07 Week 10

Snowflake set up,
connecting the db to snowflake

08 Week 12

Chart analysis,
generating a docker image for the same

Week 1

01 Installing python on Linux, virtual environments in python and the code skeleton of python, introduction to vscode intellisense ide and shortcuts for vscode

02 Data types, variables, typecasting

Desk Setup

03 basic and nested loops, switch statements, basic calculator and a mad libs game

04 String, list, dictionary and array methods such as slicing, indexing and appending

05 Functions and modules, global and local scope, basics of understanding the errors and debugging them in python

Shreshtha Modi 190130111081

Taking user input and then building a calculator with the same

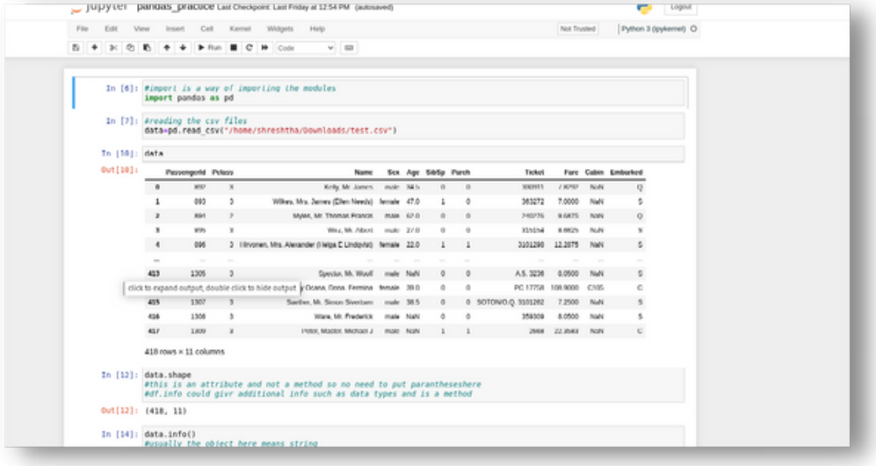
```
var1= input("Please enter the first variable")
var2=input("Please Enter the second variable")
operation=input("Please Enter the operation that you would like to perform with the variable")
def calculator (var1, var2):
     if (type(var1)and type(var2)==int):
        var1,var2,operation=var1,var2,operation
        if (operation != ['*' or '+' or '/' or '-' or '%']):
            print("Enter a valid function")
        else:
            if(operator== '+'):
                print(var1+var2)
            elif(operator == '-'):
                print(var1-var2)
            elif(operator== '*'):
                print(var1*var2)
            elif(operator == '/'):
                print(var1/var2)
            else:
               (var1%var2)
            return
```

```
Please enter the first variable2
Please Enter the second variable3
Please Enter the operation that you would like to perform with the variable'*'
```

Week 2

01 Classes and inheritance and understanding of scope

02 Modules and various types of modules, introduction to the pandas and numpy modules



03 Global and local variables and iterators

04 Pandas (importing the csv data, understanding the csv data, getting the columns and other methods), working with csv data

05 API and apis using python, pip and python file handling

Shreshtha Modi 190130111081

```python
script_name = sys.argv[0]

res = {
    "total_lines":"",
    "total_characters":"",
    "total_words":"",
    "unique_words":"",
    "special_characters":""
}

try:
    textfile = sys.argv[1]
    with open(textfile, "r", encoding = "utf_8") as f:

        data = f.read()
        res["total_lines"] = data.count(os.linesep)
        res["total_characters"] = len(data.replace(" ","")) - res["total_lines"]
        counter = collections.Counter(data.split())
        d = counter.most_common()
        res["total_words"] = sum([i[1] for i in d])
        res["unique_words"] = len([i[0] for i in d])
        special_chars = string.punctuation
```

Week 3

01 Introduction to web scraping using beautiful soup using selenium

03 Python revision and breif reading for further resources

02 What is cloud computing, advantages of cloud as compared to on prem and major cloud providers, models of cloud computing, aws free tier, storage and networking

04 Introduction to aws and the aws free tier, advantages of aws over other servies

05 AWS storage and compute services such as ec2, s3 and elastic file sharing

Shreshtha Modi 190130111081

```python
from selenium import webdriver
import csv
import time

items=[]
driver=webdriver.Chrome(r"C:/Users/hp/Anaconda3/chromedriver.exe")

driver.get('https://www.youtube.com/watch?v=iFPMz36std4')

driver.execute_script('window.scrollTo(1, 500);')

#now wait let load the comments
time.sleep(5)

driver.execute_script('window.scrollTo(1, 3000);')


username_elems = driver.find_elements_by_xpath('//*[@id="author-text"]')
comment_elems = driver.find_elements_by_xpath('//*[@id="content-text"]')
for username, comment in zip(username_elems, comment_elems):
    item = {}
    item['Author'] = username.text
    item['Comment'] = comment.text
    items.append(item)
filename = 'C:/Users/hp/Desktop/commentlist.csv'
with open(filename, 'w', newline='', encoding='utf-8') as f:
    w = csv.DictWriter(f,['Author','Comment'])
    w.writeheader()
    for item in items:
        w.writerow(item)
```

Learnings so far

01 **Asking the users for input in python**

02 **How the python interpreters work**

03 **Looping and branching to give structure to your code**

04 **Making code reusable with functions and modules**

05 **Components of clean code and how to write clean code**

06 **Understanding complex open source code and how to write code that others will understand**

07 **Getting data from the web and analyzing it**

08 **Importance of modules in python and how modules make a developers life easier**

09 **How to ask for help and talk in technical terms**

10 **Analyzing data**

Week 4

01 Understanding the databases in aws and understanding how to deploy a static website on aws+ advanced s3

02 Introduction to fully managed services in aws such as lambda and fargate

03 aws billing and pricing understanding along with exploring services such cloudwatch and budgets and analytics

04 AWS free tier introduction and introduction to free tier of various services and how to stay in free tier

05 Introduction to the shared responsibility model of the aws, aws IAM and roles

Shreshtha Modi 190130111081

Date...../...../.....

~~Storing variables~~

stores as buckets

S3 : storage, backup, hybrid storage, and analytic.

- Bucket ~~na~~ created at region, no uppercase/underscore, not IP,

S3 obj = have a key i.e. the full path.

key = prefix + obj name

↳ has no option of directories in bucket.

- keys → long names w/ slashes

Obj value = content of body

↳ metadata, tags, version ID

S3 presigned url - verifies person making requests

Date...../...../.....

~~Security~~

Security - resource

↳ user based

IAM principle can access it if user permiss. or allow it / resource allows it or no explicit deny.

bucket policies - principal - account or user to apply the policy to. Actions - set of API to allow / deny

Static website - specify the index.html

↳ upload

S3 versioning: bucket level

↳ good practice

any file befor version will be null.

Week 5

01 Project breif and aws cdn services and relaiblity

02 Setting up docker on the local instance, docker components, architecture and docker basics

03 Misc aws services such as sns, inspector and macie for threat detection, Introduction to docker

04 Frequently used docker commands and building a local docker instance from an image

05 What is version control, various types of version control systems and the difference between git and github, setting up your own github account

Shreshtha Modi 190130111081

```
sudo apt-get install docker-ce docker-ce-cli containerd.io
docker-buildx-plugin docker-compose-plugin
```

The docker is now installed. Can verify the install using hello world.
Issues faced:
Dependencies installation error
(had to reinstall curl and sudo)
My system had virtualiation and kvm not installed hence had to reinstall it

After getting the docker set up and running, we can use:

https://github.com/navikt/arxaas to install the docker image on the local server:

Before running the instance, we have to login as root first and that can be done using:

```
Sudo -s
```

Learnings so far

01 **Learning about where our data goes from the website and how it is all stored**

02 **Advantages of cloud as compared to on prem**

04 **Types of cloud providers and the cloud race**

05 **Amazon Wbe services and why they are dominating the industry**

07 **Various aspects and services of aws**

08 **IHow cloud can stack up bills quickly and how to save money on them**

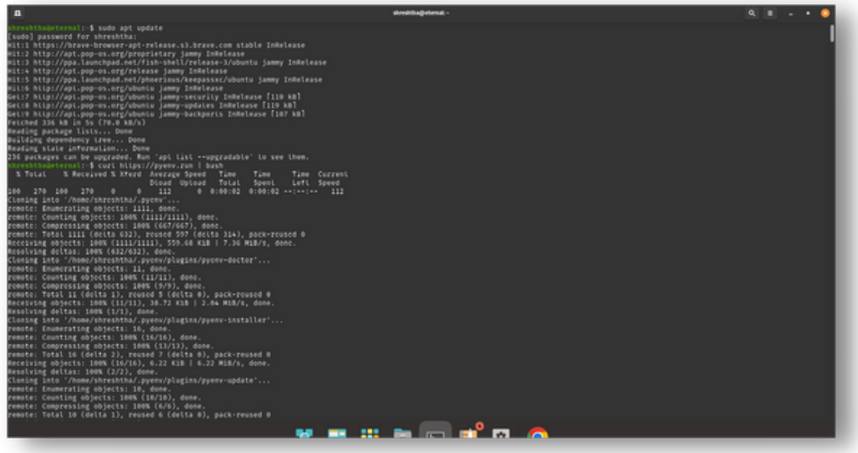
09 **Some use cases are better suited for cloud and someon prem**

10 **Integrating various parts of the project with cloud**

Week 6

01 Frequently used github commands to clone a repo, push and pull

02 Introduction to Operating Systems and kernels



03 Introduction to linux and it's various distributions

04 Installing ubuntu 22.04 on a local machine and virtual machine and explanation about various components of the linux os

05 working with files, directories and linux file structures, open ssh and numerical permissions

Shreshtha Modi 190130111081

Workspaces Applications Mar 2 2:29 PM

root@eternal: /home/shreshtha

```
lib/systemd/system/qemu-kvm.service.
Setting up qemu-system-x86 (1:6.2+dfsg-2ubuntu6.6) ...
Setting up libpmemobj1:amd64 (1.11.1-3build1) ...
Setting up librbd1 (17.2.0-0ubuntu0.22.04.2) ...
Setting up qemu-utils (1:6.2+dfsg-2ubuntu6.6) ...
Setting up libiscsi7:amd64 (1.19.0-3build2) ...
Setting up libgfrpc0:amd64 (10.1-1) ...
Setting up qemu-system-gui (1:6.2+dfsg-2ubuntu6.6) ...
Setting up libgfapi0:amd64 (10.1-1) ...
Setting up qemu-block-extra (1:6.2+dfsg-2ubuntu6.6) ...
Created symlink /etc/systemd/system/multi-user.target.wants/run-qemu.mount → /li
b/systemd/system/run-qemu.mount.
Processing triggers for libc-bin (2.35-0ubuntu3.1) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
root@eternal:/home/shreshtha# sudo apt install libvirt-clients libvirt-daemon-system libvirt-daemon virtinst bridge-utils qemu qemu-kvm #installing kvm since we need it to use docker
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'qemu-system-x86' instead of 'qemu-kvm'
bridge-utils is already the newest version (1.7-1ubuntu3).
bridge-utils set to manually installed.
qemu-system-x86 is already the newest version (1:6.2+dfsg-2ubuntu6.6).
The following packages were automatically installed and are no longer required:
  appmenu-gtk-module-common appmenu-gtk2-module appmenu-gtk3-module gir1.2-gweather-4.0 libappmenu-gtk2-parser0 libappmenu-gtk3-parser0 libgl1-amber-dri libllvm13 libopenaptx0
  linux-headers-5.16.19-76051619 linux-headers-5.16.19-76051619-generic linux-headers-5.17.5-76051705 linux-headers-5.17.5-76051705-generic linux-image-5.16.19-76051619-generic
  linux-image-5.17.5-76051705-generic linux-modules-5.16.19-76051619-generic linux-modules-5.17.5-76051705-generic pipewire-audio-client-libraries
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  gir1.2-libosinfo-1.0 jq libburn4 libgovirt-common libgovirt2 libgtk-vnc-2.0-0 libgvnc-1.0-0 libisoburn1 libisofs6 libjq1 libjte2 libnss-mymachines libonig5 libosinfo-1.0-0
  libphodav-2.0-0 libphodav-2.0-common libspice-client-glib-2.0-8 libspice-client-gtk-3.0-5 libtpms0 libusbredirhost1 libvirt-daemon-config-network libvirt-daemon-config-nwfilter
  libvirt-daemon-driver-qemu libvirt-daemon-system-systemd libvirt-glib-1.0-0 libvirt-glib-1.0-data libvirt0 mdevctl osinfo-db python3-libvirt python3-libxml2
  spice-client-glib-usb-acl-helper swtpm swtpm-tools systemd-container virt-viewer xorriso
Suggested packages:
  libosinfo-l10n gstreamer1.0-plugins-bad libvirt-login-shell libvirt-daemon-driver-storage-gluster libvirt-daemon-driver-storage-iscsi-direct libvirt-daemon-driver-storage-rbd
  libvirt-daemon-driver-storage-zfs libvirt-daemon-driver-lxc libvirt-daemon-driver-vbox libvirt-daemon-driver-xen numad auditd nfs-common open-iscsi pm-utils systemtap zfsutils trousers
  python3-argcomplete xorriso-tcltk jigit cdck
The following NEW packages will be installed:
  gir1.2-libosinfo-1.0 jq libburn4 libgovirt-common libgovirt2 libgtk-vnc-2.0-0 libgvnc-1.0-0 libisoburn1 libisofs6 libjq1 libjte2 libnss-mymachines libonig5 libosinfo-1.0-0
  libphodav-2.0-0 libphodav-2.0-common libspice-client-glib-2.0-8 libspice-client-gtk-3.0-5 libtpms0 libusbredirhost1 libvirt-clients libvirt-daemon libvirt-daemon-config-network
  libvirt-daemon-config-nwfilter libvirt-daemon-driver-qemu libvirt-daemon-system libvirt-daemon-system-systemd libvirt-glib-1.0-0 libvirt-glib-1.0-data libvirt0 mdevctl osinfo-db
  python3-libvirt python3-libxml2 qemu spice-client-glib-usb-acl-helper swtpm swtpm-tools systemd-container virt-viewer virtinst xorriso
0 upgraded, 42 newly installed, 0 to remove and 232 not upgraded.
Need to get 8,837 kB of archives.
After this operation, 35.1 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Week 7

01 shell training to learn more about installing and updating the packages, debugging dependency errors and using regex in shell to automate things, package manager

02 Project breif and project setup with the practical use case

03 IManaging system units and logs, and managing users, scp and rsync

04 Understanding the scope of the project and reading about various types of anonymization techniques in aws

05 Installing arx

Shreshtha Modi 190130111081

```python
#importing the neccesary modules and functions
import numpy as np
import pandas as pd
import pyarxaas
from pyarxaas import ARXaaS
from pyarxaas.hierarchy import IntervalHierarchyBuilder, OrderHierarchyBuilder
from pyarxaas import AttributeType
from pyarxaas.privacy_models import KAnonymity
from pyarxaas import Dataset
#creating a local arx instance, can be created using running the docker image locally
arxaas = ARXaaS("http://localhost:8080/")
#print(data)
#loading the data which contains the zipcodes
data=pd.read_csv("/home/shreshtha/Documents/zipcode.csv",header=None,usecols=[0],names=['zipcode'])
dataset = Dataset.from_pandas(data)
#loading the zipcode hierarchies present in the csv files, if dont want to define manually then can be done automatically with arx
zipcode_hierarchy=pd.read_csv("/home/shreshtha/Documents/zipcode.csv",header=None,usecols=[1,2,3,4,5])
#print(zipcode_hierarchy.head)
#setting the attribute types for the dataset column
dataset.set_attribute_type(AttributeType.QUASIIDENTIFYING)
#creating a risk profile
risk_profile = arxaas.risk_profile(dataset)
#reidentification risk states how much the data is at the risk of reidentifying
#print(risk_profile.re_identification_risk)
#attacker success rate tells you how much the data can be attacked by which model of attacking
#print(risk_profile.attacker_success_rate)
print(risk_profile.population_model)
```

Learnings so far

01 **Various types of operating system and computer architecture**

02 **What is linux**

03 **The beauty of open source**

04 **How to learn tough concepts**

05 **Moving away from windows and getting used to the cli**

06 **Understanding how linux works and the flexiblity it provides**

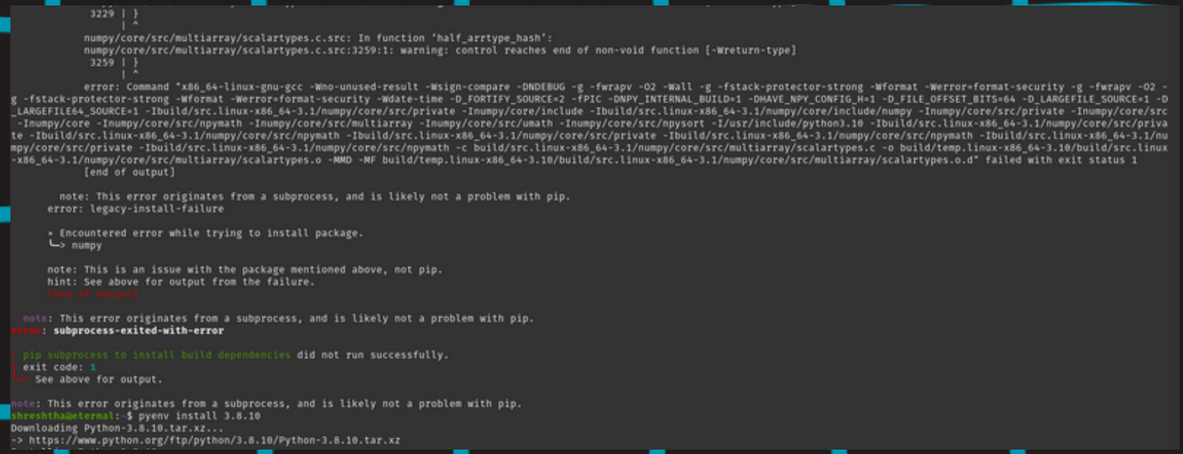
07 **IDocker and the importance of container technology**

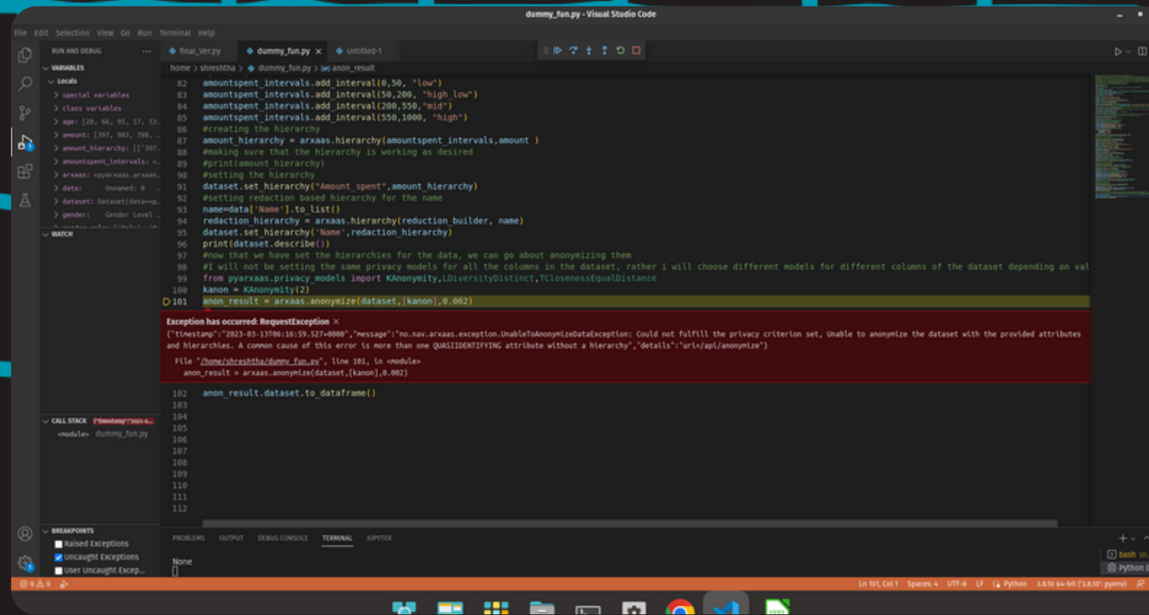
08 **Importance of version control in the life of a developer**

09 **If it works, dont touch it**

Challanges

Although facing challanges is a common part of programming and it more often than not indicates that you are going in the right direction, sometimes the challanges can often be a roadblock to the further progress. One of the biggest challanges i faced while working for internship till date was trying to configure the pyarxaas module. The module has been outdated with no support for the latest python version or the bugs. Having a system which runs on python 3.10 i could not downgrade the python version globally as that would mean dependency error for a lot of the system default modules, i had to figure out how to have multiple versions of python in the same memory limited system along with making sure that the code and the dependency are not broken





Project Abstract

Internet has an approximate of 4.6 billion users worldwide and they are growing at an astronomical rate. With the rise in users, privacy concerns also pop up. Data privacy is a complex area which happens to be a vital part of the internet ecosystem. This project aims at protecting the privacy of the user data by anonymizing the data using various statistical tools and techniques and proposing an end to end solution of the anonymized data The proposed solution's schema can be modified according to the need of the user and the type of the data with ease

Why data anonymization

2006 AOL data search leak. Here are some interesting statistics about the data leak:

- data of approximately 650,000 users along with 20 Million search results were leaked
- The AOL did not identify the users in the data as the names of the users were not explicitly mentioned in the data
- However, a popular newspaper magazine called New York times were able to identify the users
- Netflix data breach where the users were identified after cross examination even when the user info was removed

- anyone these days can use just about anything to get your personal info hence need to make sure it is secured

Identifiers	Quasi-Identifiers			Confidential Attributes	
SSN	Gender	Age	Zip Code	Hourly Wage	Political Affiliation
432-55-1356	M	22	94024	$34	Democrat
123-70-4351	F	26	94305	$42	Republican
471-65-3560	M	24	94024	$18	Republican
351-34-2819	M	40	90210	$40	Democrat
241-41-7632	F	38	90210	$41	Independent
501-33-2094	F	42	90213	$37	Democrat

Perturbed Quasi-Identifiers			Confidential Attributes	
Gender	Age	Zip code	Hourly Wage	Political Affiliation
M	24	94***	$34	Democrat
M	24	94***	$42	Republican
M	24	94***	$18	Republican
F	40	90***	$40	Democrat
F	40	90***	$41	Independent
F	40	90***	$37	Democrat

k-Anonymized Records

Week 8

01 Understanding arx and types of anonymization

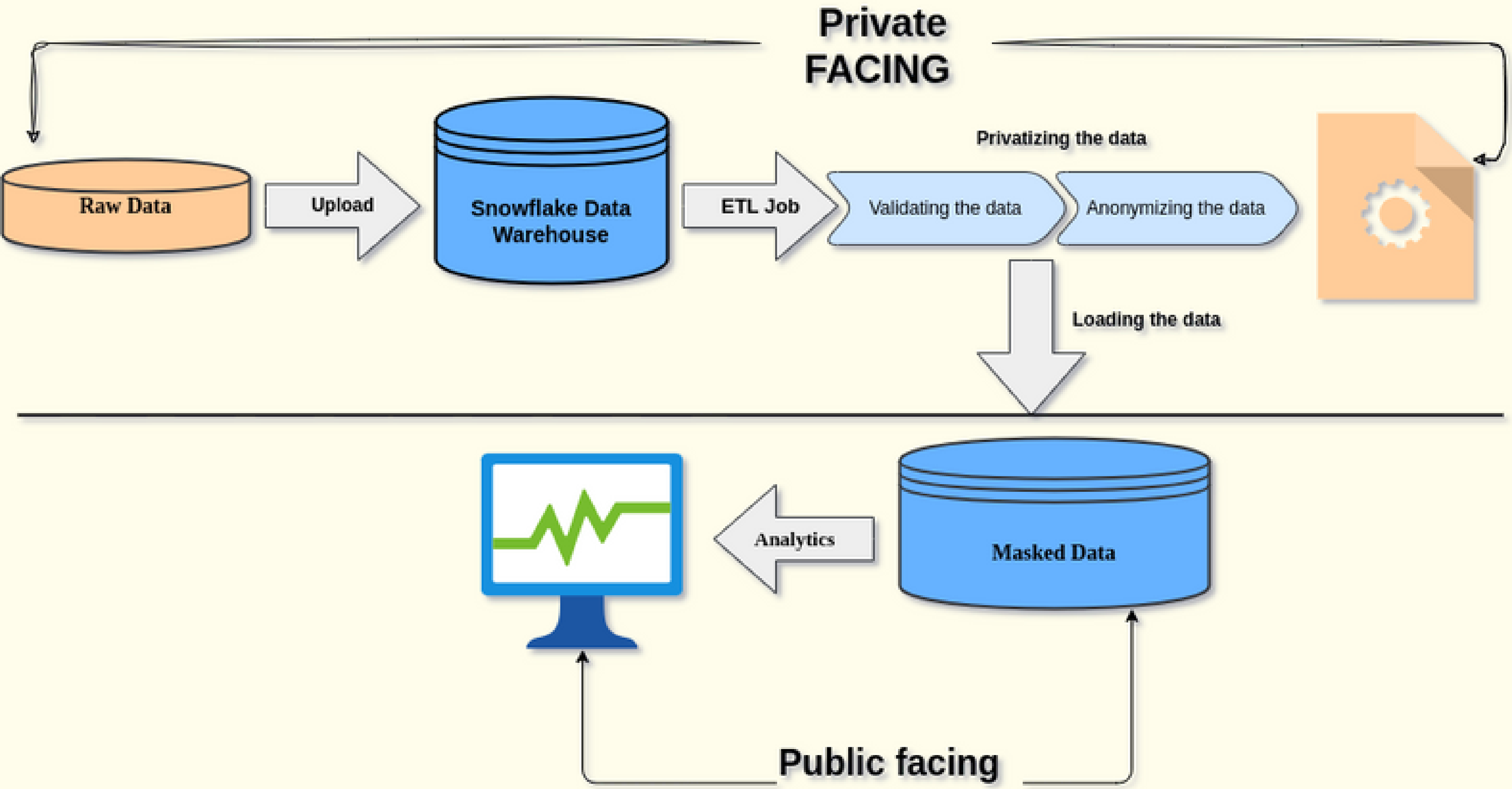
02 Installing arx from source locally and pyarxaas

03 IGetting familiar with pyarxaas on the titanic dataset

04 Setting hierarchies for the titanic dataset

05 Debugging errors regarding pyarxaas version and trying to install pyarxaas on conda

Shreshtha Modi 190130111081


Private
FACING
Raw Data
Upload
Snowflake Data Warehouse
ETL Job
Privatizing the data
Validating the data
Anonymizing the data
Loading the data
Analytics
Masked Data
Public facing

Project Components

- Python (Pandas)
- Python(Numpy)
- Python(Pyarxaas)
- ARXAAS(Docker)
- Snowflake
- VS CODE
- Python(Faker)
- Python(Matplotlib)
- Apache airflow

Week 9

01 Installing the faker library and creating the dummy data

02 DUmmy data creation and validation using hypothesis testing

03 Generating hierarchies for the dummy data and setting hierarchies for the same

04 Anonymizing the final data and risk ananlysis

05 Modularizing function and documentation

Shreshtha Modi 190130111081

Week 10

01 Learning about snowflake and common snowflake commands

02 Loading the dataset into the snowflake and accessing the snowflake instance via the local python script

03 Debugging the errors and anonymizing and setting hierarchies for the data

04 Anonymizing the final data and risk ananlysis

05 Modularizing function and documentation

Shreshtha Modi 190130111081

Shreshtha Modi 190130111081

Thank You So Much!

Eternal soft solutions ✦ GEC Gandhinagar(EC) ✦ 2023