DIGITAL
SKOLA

LEARNING PROGRESS REVIEW "WEEK 6"

BY : DREAM (KELOMPOK 1)

Data Realm Engineers And Maestros

ANGGOTA KELOMPOK

Afroh Fauziah

Althaf Nawadir Taqiyyahh

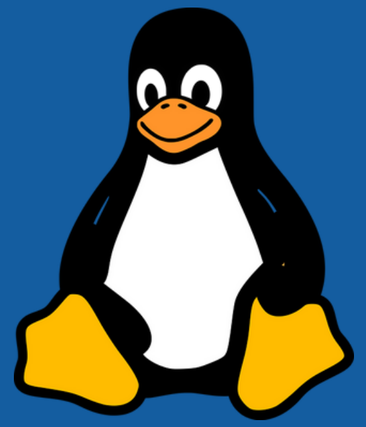
Andi Rosilala

Andrew Bintang Pratama

Andrew Fortino Mahardika Suadnya

DigitalSkola
Uncover The World of Digital Skills with Us

LINUX I



SEJARAH LINUX

Dari Linus Torvalds yang belajar di Universitas Helsinki, dimana ia membuat kernel kecil diluar dari windows yang dulu pernah ada, hingga menghasilkan bentuk UNIX dari projectnya untuk dapat direkayasa dan dikembangkan, juga tidak untuk dikomersialkan. UNIX punya kecenderungan sama dengan sintax-sintaxnya.

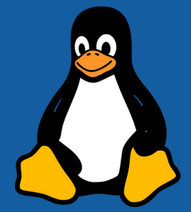
Operasi Linux awalnya non komersial seperti open source, hingga populer setelah ada berbagai macam Distro Linux seperti solaris, fedora, ubuntu dan sebagainya dengan tampilan Desktop ada GNOME dan KDE, juga kelebihan bahwa Linux ini free jadi banyak digunakan di perkantoran.

SISTEM OPERASI LINUX

TYPE USER LINUX

- Root (superuser) > bisa melakukan semua command / file direktori, menggunakannya dengan sudo.
- User biasa > akses tertentu.

LINUX



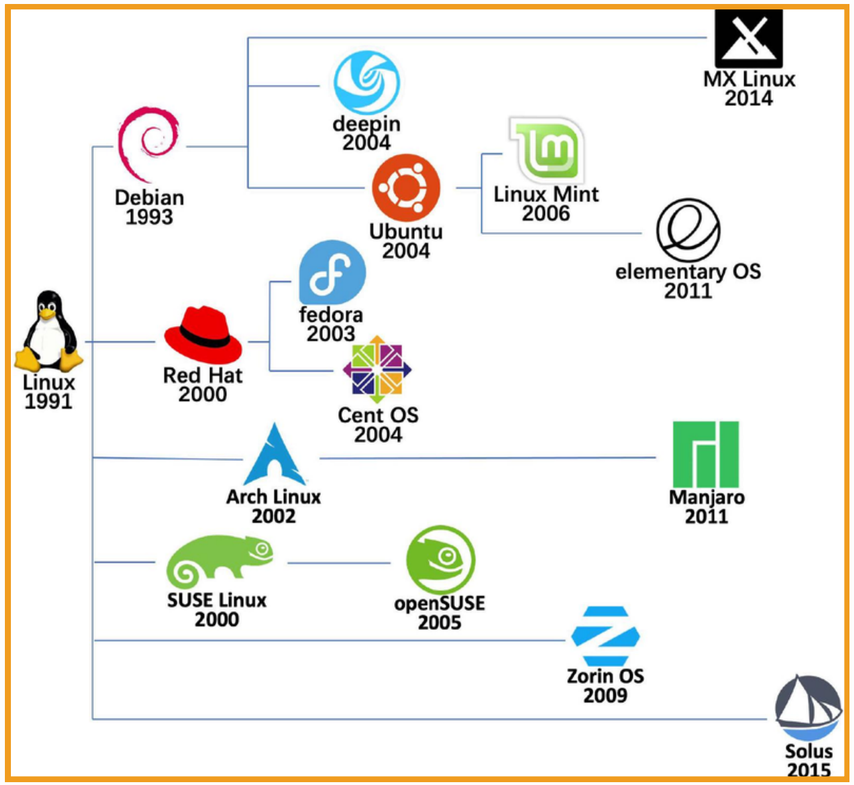
Linux adalah sistem operasi open source yang berjalan pada berbagai macam perangkat keras, mulai dari komputer desktop dan server hingga perangkat mobile dan embedded system. Berbeda dengan sistem operasi proprietary yang kode sumbernya tertutup, kode sumber Linux tersedia secara bebas dan dapat diakses serta dimodifikasi oleh siapa saja.

Karakteristik Utama Linux:

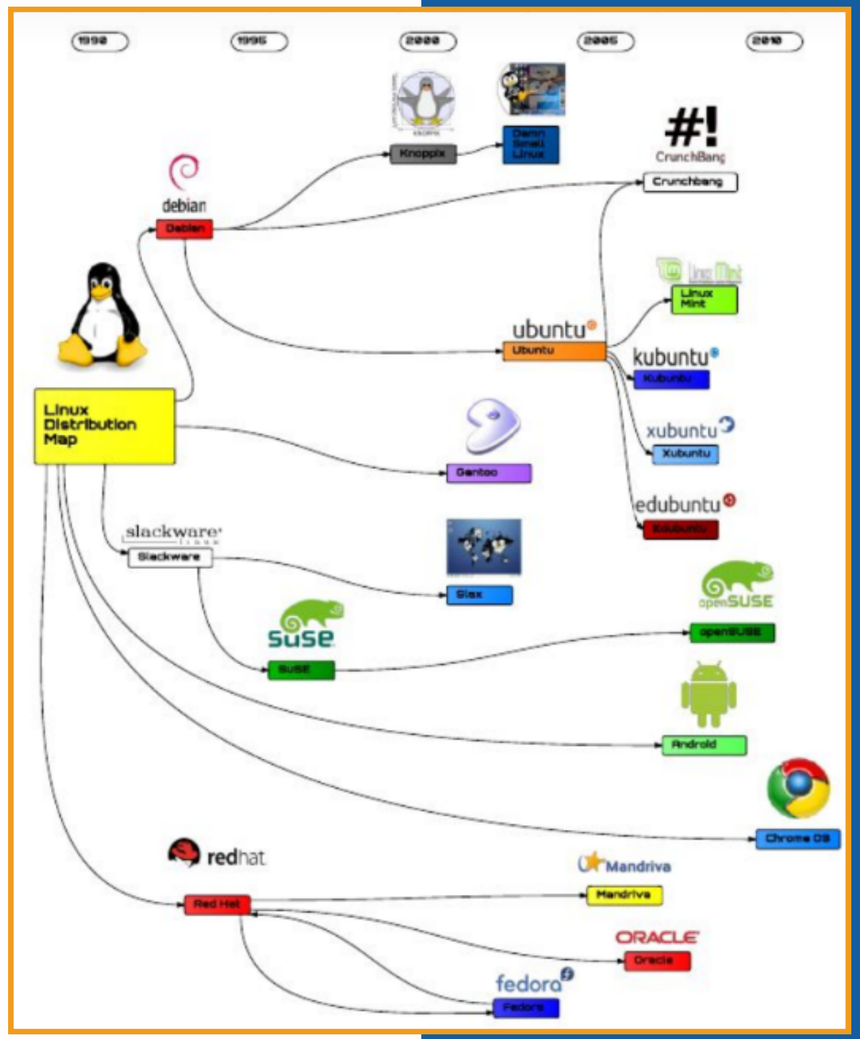
- **Open-source**: Kode sumber Linux tersedia secara bebas untuk semua orang. Ini berarti bahwa siapa saja dapat melihat, memodifikasi, dan mendistribusikan Linux.
- **Gratis**: Linux dapat digunakan dan didistribusikan secara gratis. Tidak ada biaya lisensi yang terkait dengan Linux.
- **Multi-user**: Linux memungkinkan banyak pengguna untuk menggunakan komputer secara bersamaan.
- **Multi-tasking**: Linux memungkinkan beberapa program untuk berjalan secara bersamaan.
- **Stabil**: Linux dikenal sebagai sistem operasi yang stabil dan andal.
- **Aman**: Linux memiliki tingkat keamanan yang tinggi.

LINUX DIXTRO (DISTRIBUTION)

Linux Distro adalah operating system (OS) yang dibuat dari koleksi software berdasarkan Linux kernel.

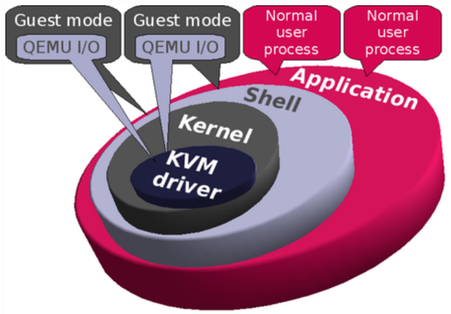


Linux Distro adalah versi atau distribusi Linux yang merupakan kumpulan dari kernel Linux, perangkat lunak aplikasi, dan utilitas yang dikemas bersama dan dibagikan sebagai sebuah sistem operasi lengkap.

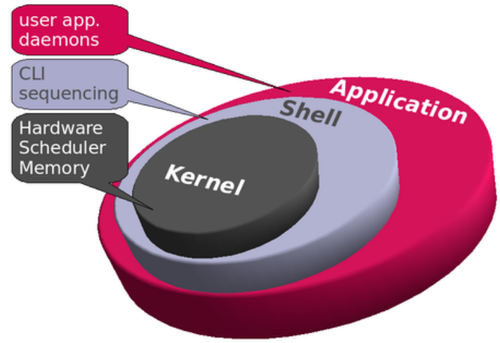


DigitalSkola
Uncover The World of Digital Skills with Us

LINUX KERNEL



- Linux Kernel adalah inti dari Linux itu sendiri, suatu perangkat lunak yang menjadi bagian utama atau terpenting dari sistem operasi yang berinteraksi langsung dengan perangkat keras komputer, tugasnya melakukan operasi atau menjadi penghubung hardware dan software.
- Bagian-bagian penting dalam kernel linux diantaranya memanajemen proses dan memori, hardware device drivers, filesystem drivers, manajemen jaringan dan lain sebagainya.



BOOT LOADER

Boot Loader adalah suatu program yang sudah tertanam pada sistem operasi untuk mem-boot atau memanggil kernel sistem operasi yang ada pada hard disk dan media boot lainnya misal GRUB dan LILO. Boot Loader digunakan untuk memilih sistem operasi yang ada pada hard disk karena memiliki lebih dari 1 sistem operasi. Boot Loader dimuat pada BIOS komputer untuk memulai pada saat dinyalakan. Tugasnya dapat berupa menginisialisasi perangkat keras, memuat kernel sistem operasi ke dalam memori, dan menjalankan kernel tersebut agar sistem operasi dapat mulai berjalan.

TIPE PROSES LINUX

Foreground Process

Proses yang terlihat user, berjalan melalui inisiasi dan dapat dikontrol melalui terminal session dengan nama lain interactive processes. Prosesnya berjalan setelah dijalankan oleh user. Sehingga tidak berjalan secara otomatis.

Background Process

Proses yang tidak terlihat user, tidak dikenali pada terminal session, tetapi begitulah cara jalannya proses ini, berjalan secara independen dari terminal. Sehingga tidak mengharapkan input apapun dari user.

MANAJEMEN PROSES LINUX

TOP

Menampilkan informasi tentang proses yang sedang berjalan di sistem, dapat pakai berbagai opsi untuk mengatur tampilan dan filter informasinya.

PS

Menampilkan daftar proses yang sedang berjalan bersama dengan informasi seperti penggunaan CPU, penggunaan memori, waktu mulai, dan lainnya secara real-time.

Perintah Dasar Linux :

Navigasi:

- **cd**: Berpindah direktori.
- **pwd**: Menampilkan lokasi direktori saat ini.

Pengelolaan File & Direktori:

- **ls**: Menampilkan daftar isi direktori.
- **mkdir**: Membuat direktori baru.
- **rmdir**: Menghapus direktori kosong.
- **rm**: Menghapus file.
- **mv**: Memindahkan/mengubah nama file/direktori.
- **cp**: Menyalin file/direktori.

Informasi Sistem:

- **clear**: Membersihkan layar terminal.
- **help**: Menampilkan informasi bantuan untuk perintah lain.
- **man**: Menampilkan halaman manual untuk perintah tertentu.

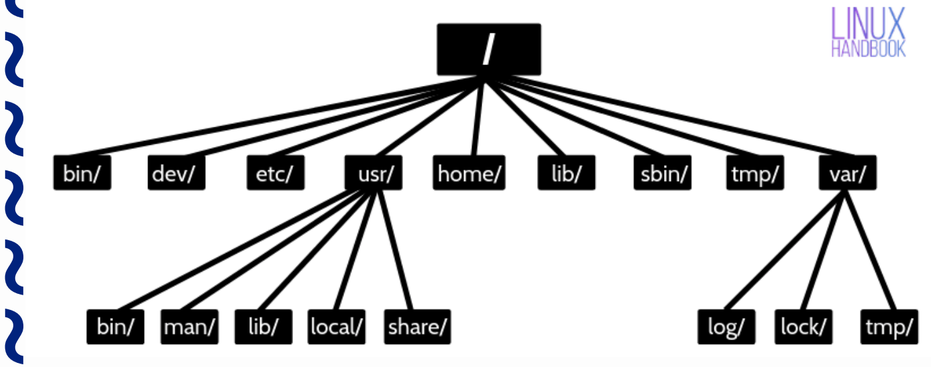
Izin Akses:

- **chmod**: Mengubah izin akses file/direktori.

HANDS-ON

```
Last login: Mon Mar 25 13:30:55 2024 from 103.162.237.23
root@ubuntu-s-2vcpu-4gb-amd-sfo3-01:~# ls
aisa  fauzan  ferdian  hans  mahda  septi  snap  syarif  zita  zyah
root@ubuntu-s-2vcpu-4gb-amd-sfo3-01:~# mkdir afroh
root@ubuntu-s-2vcpu-4gb-amd-sfo3-01:~# ls -ltr
total 84
drwx------ 3 root root 4096 Mar 25 13:14  snap
drwxr-xr-x 2 root root 4096 Mar 25 13:31  mahda
drwxr-xr-x 2 root root 4096 Mar 25 13:31  zyah
drwxr-xr-x 2 root root 4096 Mar 25 13:31  fauzan
drwxr-xr-x 2 root root 4096 Mar 25 13:31  ferdian
drwxr-xr-x 2 root root 4096 Mar 25 13:31  syarif
drwxr-xr-x 2 root root 4096 Mar 25 13:31  hans
drwxr-xr-x 2 root root 4096 Mar 25 13:31  septi
drwxr-xr-x 2 root root 4096 Mar 25 13:31  zita
drwxr-xr-x 2 root root 4096 Mar 25 13:31  aisa
drwxr-xr-x 2 root root 4096 Mar 25 13:31  zidan
drwxr-xr-x 2 root root 4096 Mar 25 13:31  oliver
drwxr-xr-x 2 root root 4096 Mar 25 13:31  nisaokt
drwxr-xr-x 2 root root 4096 Mar 25 13:31 'tina^C'
drwxr-xr-x 2 root root 4096 Mar 25 13:31  nicholas
drwxr-xr-x 2 root root 4096 Mar 25 13:31  tina
drwxr-xr-x 2 root root 4096 Mar 25 13:31  anisa
drwxr-xr-x 2 root root 4096 Mar 25 13:32  afroh
drwxr-xr-x 2 root root 4096 Mar 25 13:32  gustian
drwxr-xr-x 2 root root 4096 Mar 25 13:32  Ramdani
drwxr-xr-x 2 root root 4096 Mar 25 13:32  althaf
root@ubuntu-s-2vcpu-4gb-amd-sfo3-01:~# cd afroh
root@ubuntu-s-2vcpu-4gb-amd-sfo3-01:~/afroh#
```

men list lebih detail

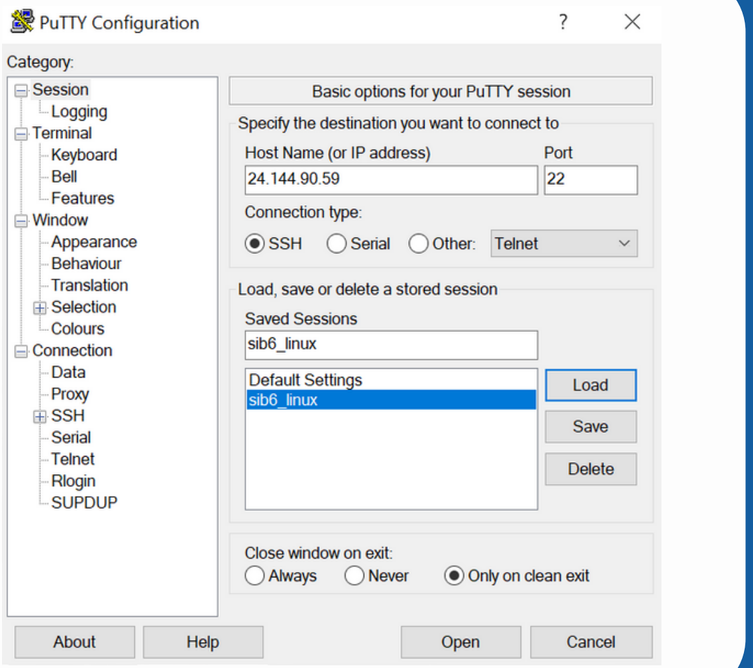


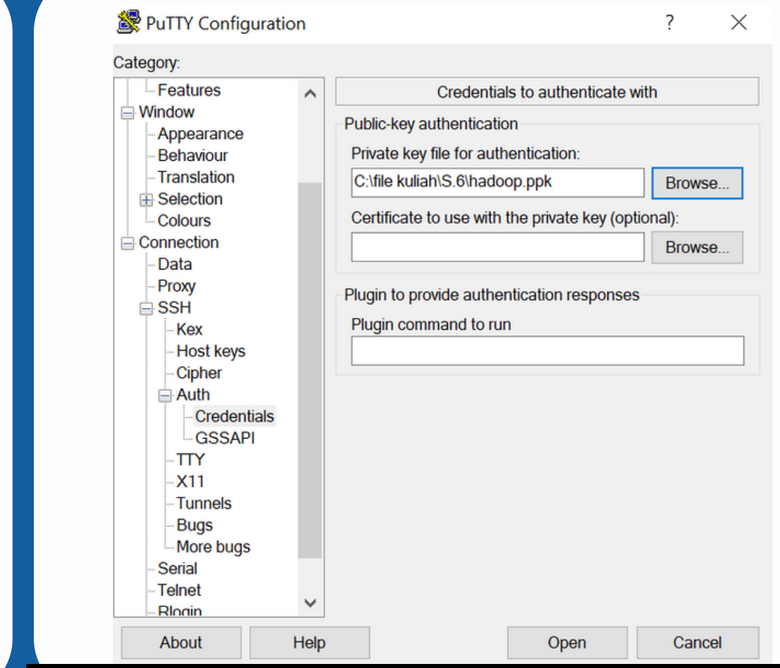
Linux Basic Commands : Directory

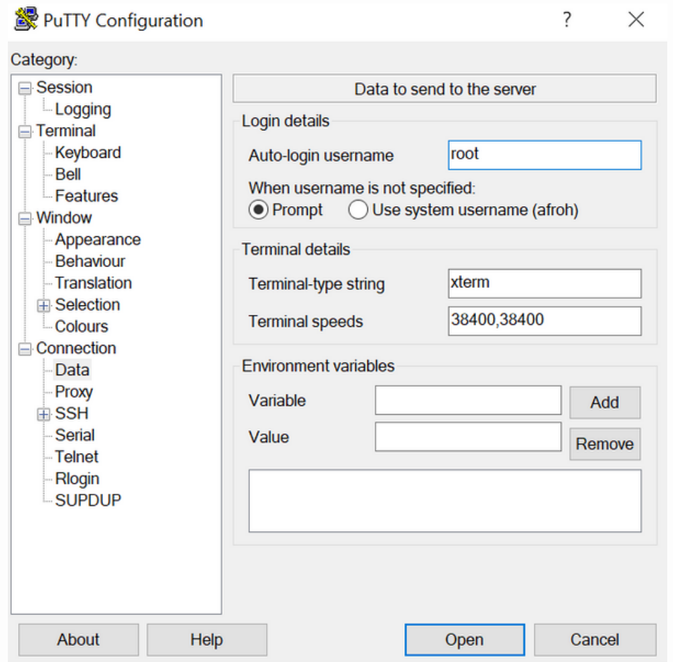
- ls : Perintah untuk melihat isi / list dari dir yang dibuat.
- mkdir : Perintah untuk membuat folder teruntuk nama masing-masing.
- ls -l : Berdasarkan abjad.
- ls -lt : Berdasarkan waktu pembuatan bawah ke atas.
- ls -ltr : Dibalikkan waktu dibuat dipaling atas.
- cd : Perintah untuk berpindah direktori / masuk ke dir masing-masing.

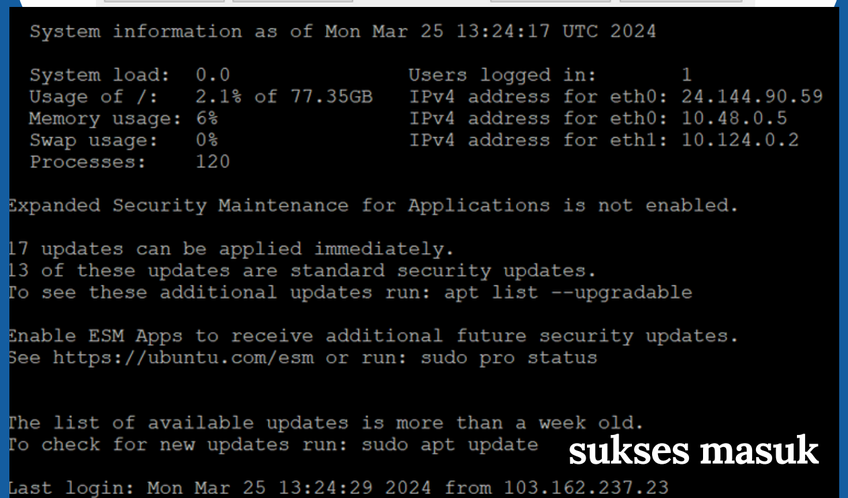
HANDS-ON Menggunakan PuTTY

Buka aplikasi PuTTY lalu masukkan IP address, nama sessions, hadoop, dan username saat login.









sukses masuk

HANDS-ON

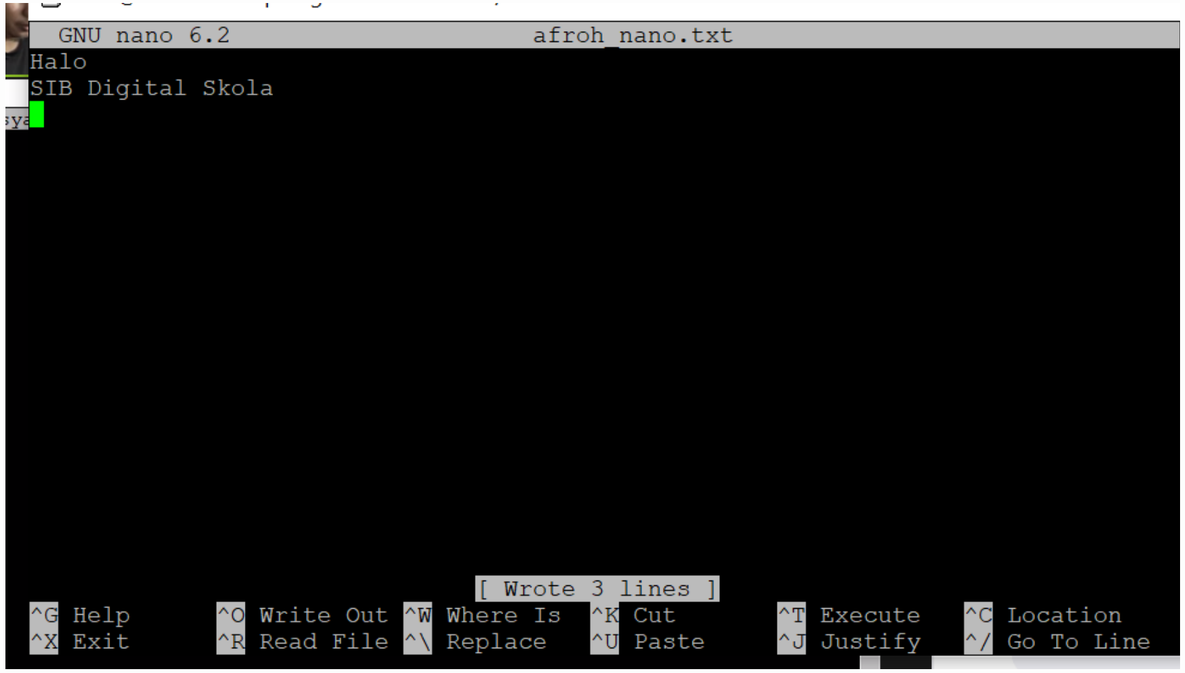
Linux Basic Commands: File

touch : Perintah untuk membuat file dengan touch_nama.txt
Misal touch_afroh.txt
Lalu lihat listnya menggunakan perintah ls -ltr */*

```
oot@ubuntu-s-2vcpu-4gb-amd-sfo3-01:~# ls -ltr */*
rw-r--r-- 1 root root    0 Mar 25 13:46 mahda/mahda_file.txt
rw-r--r-- 1 root root    0 Mar 25 13:46 nicholas/nicholas_file.txt
rw-r--r-- 1 root root    0 Mar 25 13:46 fauzan/fauzan_file.txt
rw-r--r-- 1 root root    0 Mar 25 13:46 septi/septi_file.txt
rw-r--r-- 1 root root    0 Mar 25 13:46 hafizh/hafizh_file.txt
rw-r--r-- 1 root root    0 Mar 25 13:46 afroh/afroh_file.txt
rw-r--r-- 1 root root    0 Mar 25 13:46 ferdian/ferdian_file.txt
rw-r--r-- 1 root root    0 Mar 25 13:46 angel/angel_file.txt
rw-r--r-- 1 root root    3 Mar 25 13:47 althaf/althaf_file.txt
rw-r--r-- 1 root root    0 Mar 25 13:47 Ramdani/Ramdani_file.txt
rw-r--r-- 1 root root    0 Mar 25 13:47 syarif/syarif_file.txt
rw-r--r-- 1 root root    0 Mar 25 13:47 nisaokt/nisaokt_file.txt
rw-r--r-- 1 root root    0 Mar 25 13:47 zita/zita_file.txt
rw-r--r-- 1 root root    0 Mar 25 13:47 anisa/anisa_file.txt
rw-r--r-- 1 root root    0 Mar 25 13:48 zidan/zidan_file.txt
```

*Cara copy paste : block lalu klik kanan

nano : Text editor dengan nano nama_nano.txt
Misal nano afroh_nano.txt
Lalu hasilnya akan masuk ke bentuk edit textnya



Edit text lalu save dengan Ctrl+O lalu dienter dan kembali ke halaman utama dengan Ctrl+X

DigitalSkola

HANDS-ON

Text editor menggunakan vi yakni “vi nama_vi.txt”
Misal “vi afroh_vi.txt”

Untuk memulai ketik i sampai muncul mode INSERT lalu ketikan text

```
Halo
FROM VI
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --                                                    3,8
```

Untuk save harus kembali ke cursor dengan Ctrl+C lalu :w lalu enter sampai muncul gambar dibawah ini

```
~
"afroh_vi.txt" [New] 3L, 14B written                          3,7           All
```

Untuk save dan keluar bisa dengan :wq (write & quit)

Untuk memastikan text sudah ada, bisa dengan cat (perintah untuk membaca file)
misal cat afroh_vi.txt & cat afroh_nano.txt

```
root@ubuntu-s-2vcpu-4gb-amd-sfo3-01:~/afroh# vi afroh_vi.txt
root@ubuntu-s-2vcpu-4gb-amd-sfo3-01:~/afroh# cat afroh_vi.txt

Halo
FROM VI
root@ubuntu-s-2vcpu-4gb-amd-sfo3-01:~/afroh# cat afroh_nano.txt
Halo
SIB Digital Skola
```

DigitalSkola

HANDS-ON

DigitalSkola

Mencoba dengan text lebih panjang di nano dengan “nano test.txt”

```
 GNU nano 6.2                                                    test.txt *
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Venenatis>

Tincidunt arcu non sodales neque sodales ut etiam sit amet. Id neque aliquam vestibulum morbi blandit cursus risus at. In eu mi biben>

Nullam vehicula ipsum a arcu cursus vitae congue. Risus sed vulputate odio ut. Porttitor massa id neque aliquam vestibulum. Pellentes>

Posuere lorem ipsum dolor sit amet consectetur. Lobortis feugiat vivamus at augue eget arcu dictum. Enim diam vulputate ut pharetra s>

Quis blandit. Proin fermentum leo vel orci porta non. Ac turpis egestas sed tempus. Lorem sed risus ultricies tristique nulla.
```

*Perintah lain untuk membaca file selain cat yakni ada less, head, & tail.

```
root@ubuntu-s-2vcpu-4gb-amd-sfo3-01:~/afroh# cp afroh_nano.txt nano_afroh.txt
```

Mengcopy file

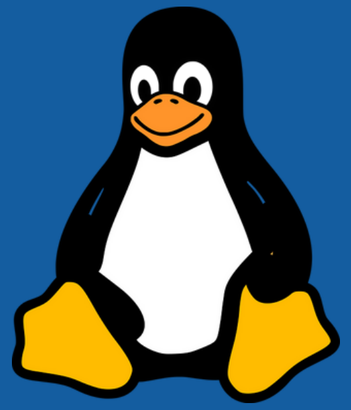
Hasil list file yang dibuat

```
root@ubuntu-s-2vcpu-4gb-amd-sfo3-01:~/afroh# ls -ltr
total 20
-rw-r--r-- 1 root root    0 Mar 25 13:46 afroh_file.txt
-rw-r--r-- 1 root root   24 Mar 25 13:51 afroh_nano.txt
-rw-r--r-- 1 root root   14 Mar 25 14:02 afroh_vi.txt
-rw-r--r-- 1 root root 4442 Mar 25 14:12 test.txt
-rw-r--r-- 1 root root   24 Mar 25 14:19 nano_afroh.txt
```

Cek proses yang sedang berjalan bisa dengan perintah htop

```
  0[||                        2.0%]   Tasks: 103, 37 thr; 1 running
  1[||                        1.3%]   Load average: 0.01 0.01 0.00
Mem[||||||||           385M/3.83G]   Uptime: 01:02:17
Swp[                     0K/0K]

    PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%▽  TIME+  Command
    728 root       20   0  998M 48024 20276 S  0.0  1.2  0:02.13 /usr/lib/snapd/
    816 root       20   0  998M 48024 20276 S  0.0  1.2  0:00.09 /usr/lib/snapd/
    817 root       20   0  998M 48024 20276 S  0.0  1.2  0:00.19 /usr/lib/snapd/
    818 root       20   0  998M 48024 20276 S  0.0  1.2  0:00.00 /usr/lib/snapd/
    819 root       20   0  998M 48024 20276 S  0.0  1.2  0:00.00 /usr/lib/snapd/
    820 root       20   0  998M 48024 20276 S  0.0  1.2  0:00.23 /usr/lib/snapd/
    821 root       20   0  998M 48024 20276 S  0.0  1.2  0:00.11 /usr/lib/snapd/
    824 root       20   0  998M 48024 20276 S  0.0  1.2  0:00.10 /usr/lib/snapd/
    843 root       20   0  998M 48024 20276 S  0.0  1.2  0:00.00 /usr/lib/snapd/
    844 root       20   0  998M 48024 20276 S  0.0  1.2  0:00.12 /usr/lib/snapd/
    866 root       20   0  998M 48024 20276 S  0.0  1.2  0:00.01 /usr/lib/snapd/
    868 root       20   0  998M 48024 20276 S  0.0  1.2  0:00.11 /usr/lib/snapd/
    869 root       20   0  998M 48024 20276 S  0.0  1.2  0:00.11 /usr/lib/snapd/
    391 root       RT   0  282M 27100  9072 S  0.0  0.7  0:00.59 /sbin/multipath
    396 root       20   0  282M 27100  9072 S  0.0  0.7  0:00.00 /sbin/multipath
    397 root       RT   0  282M 27100  9072 S  0.0  0.7  0:00.00 /sbin/multipath
F1Help  F2Setup  F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9Kill  F10Quit
```

LINUX II



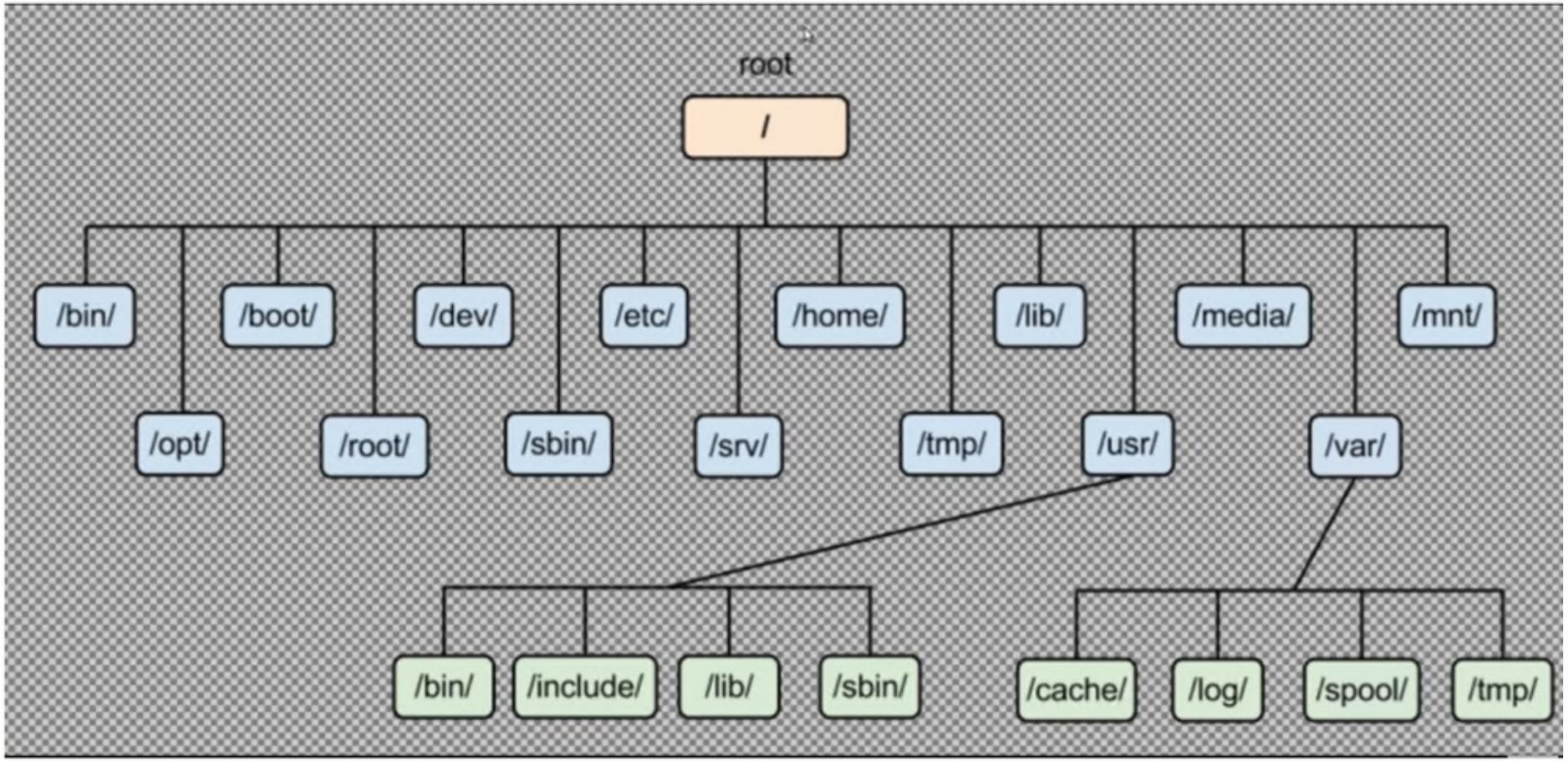
User Environment in Linux

Lingkungan pengguna (User Environment) dalam Linux dan Unix adalah kumpulan pengaturan dan konfigurasi yang mempengaruhi bagaimana pengguna berinteraksi dengan sistem

beberapa perintah untuk mengatur environment variable

- env: Perintah ini memungkinkan kita untuk menjalankan program lain di lingkungan khusus tanpa mengubah yang sekarang.
- printenv: Print environment variables yang ingin dilihat.
- set: Memasukan nilai ke dalam sebuaah environment variable.
- unset: Hapus environment variable.
- export: Memasukan nilai ke dalam sebuah environment variable.

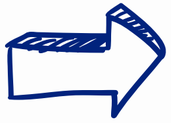
Linux Directory Structure



Linux Directory Structure

Struktur direktori Linux, juga dikenal sebagai hierarki sistem file, adalah organisasi standar file dan direktori dalam sistem Linux. Berikut beberapa direktori utama dalam struktur direktori Linux:

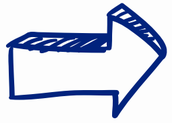
- / (root): Ini adalah direktori tingkat teratas dari hierarki sistem file.
- /bin/: Direktori ini berisi program-program penting yang dapat dieksekusi yang digunakan oleh administrator sistem dan pengguna.
- /boot/: Direktori ini berisi file yang diperlukan untuk mem-boot sistem, seperti kernel Linux dan disk RAM awal.
- /dev/: Direktori ini berisi file perangkat, yang merupakan file khusus yang mewakili perangkat keras atau antarmuka perangkat lunak ke perangkat tersebut.
- /etc/: Direktori ini berisi file konfigurasi untuk sistem dan aplikasinya.
- /home/: Direktori ini berisi direktori home untuk pengguna biasa.
- /lib/: Direktori ini berisi file perpustakaan bersama yang diperlukan oleh program yang dapat dieksekusi.



Linux Directory Structure

Struktur direktori Linux, juga dikenal sebagai hierarki sistem file, adalah organisasi standar file dan direktori dalam sistem Linux. Berikut beberapa direktori utama dalam struktur direktori Linux:

- /media/: Direktori ini digunakan sebagai titik pemasangan untuk media yang dapat dipindahkan, seperti drive USB dan CD-ROM.
- /mnt/: Direktori ini digunakan sebagai titik pemasangan sementara untuk sistem file.
- /opt/: Direktori ini berisi paket perangkat lunak aplikasi opsional.
- /root/: Ini adalah direktori home untuk pengguna administrator sistem (root).
- /sbin/: Direktori ini berisi program sistem yang dapat dijalankan yang digunakan oleh administrator sistem.
- /srv/: Direktori ini berisi data layanan yang disediakan oleh sistem.
- /tmp/: Direktori ini berisi file-file sementara yang dibuat oleh berbagai program.
- /usr/: Direktori ini berisi sebagian besar utilitas dan aplikasi pengguna, serta perpustakaan dan dokumentasi terkait.
- /var/: Direktori ini berisi file variabel, seperti file log, email sistem, dan direktori spool.



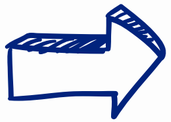
User Environment

Membuat user environment dengan mengatur environment variables. Berikut ini contohnya:

```
// load semua variabel env dari file .env
require('dotenv').config();

// mencetak variabel env
console.log("HOST: " + process.env.DB_HOST);
console.log("USER: " + process.env.DB_USER);
console.log("PASS: " + process.env.DB_PASS);
console.log("NAME: " + process.env.DB_NAME);
```

Modul dotenv untuk memuat variabel lingkungan dari file .env. Berikut tampilan file .env:

```
DB_HOST=localhost
DB_USER=dian
DB_PASS=petanikode
DB_NAME=blog_nodejs
```



User Environment

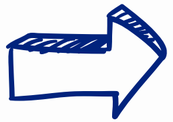
Modul dotenv diperlukan untuk menjalankan kode dari kode javascript :

Menginstal modul dotenv terlebih dahulu:

```
1  npm install dotenv
```

Dijalankan dengan menggunakan node.js

```
1  node dotenv.js
```



Shell Script Programming

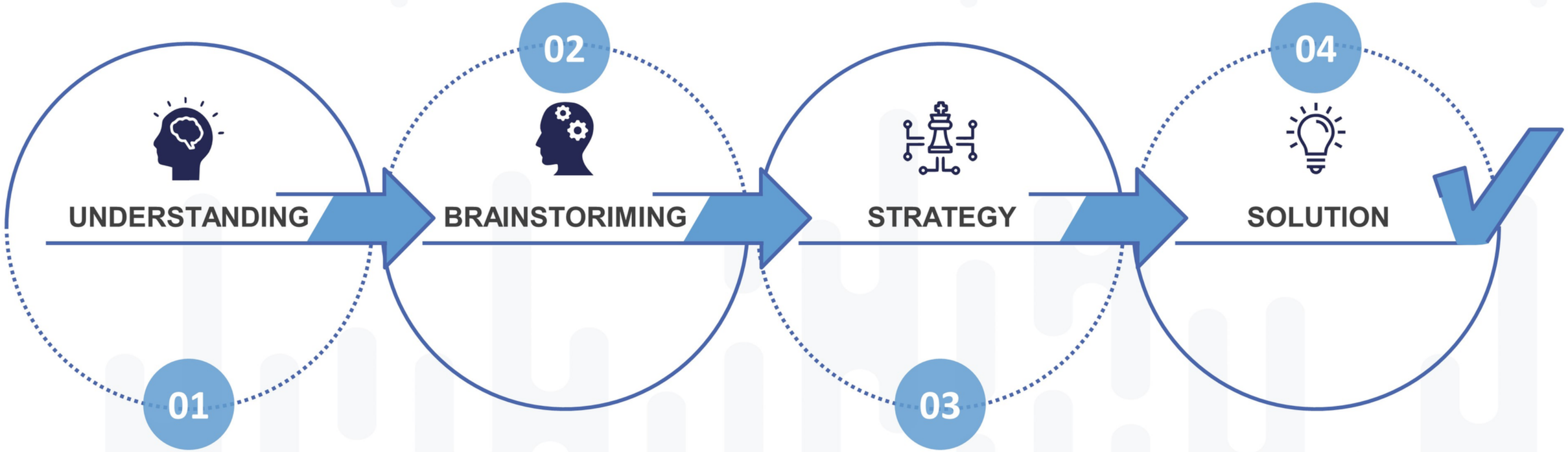
Skrip shell yang menggunakan perintah echo untuk mencetak nilai variabel:

```
#!/bin/bash

# define a variable using command substitution
CURRENT_DATE=$(date)

# print the value of the variable using echo
echo "Today's date is: $CURRENT_DATE"
```

- Kita mendefinisikan variabel bernama NAMA dengan nilai "Petanikode". Kami kemudian menggunakan perintah echo untuk mencetak string "Halo, $NAME!". Simbol $ digunakan untuk menunjukkan bahwa kita ingin mencetak nilai variabel NAME, bukan string literal "$NAME".
- Untuk menjalankan skrip ini, simpan ke file (misalnya hello.sh), buat agar dapat dieksekusi (chmod +x hello.sh), lalu jalankan (./hello.sh).

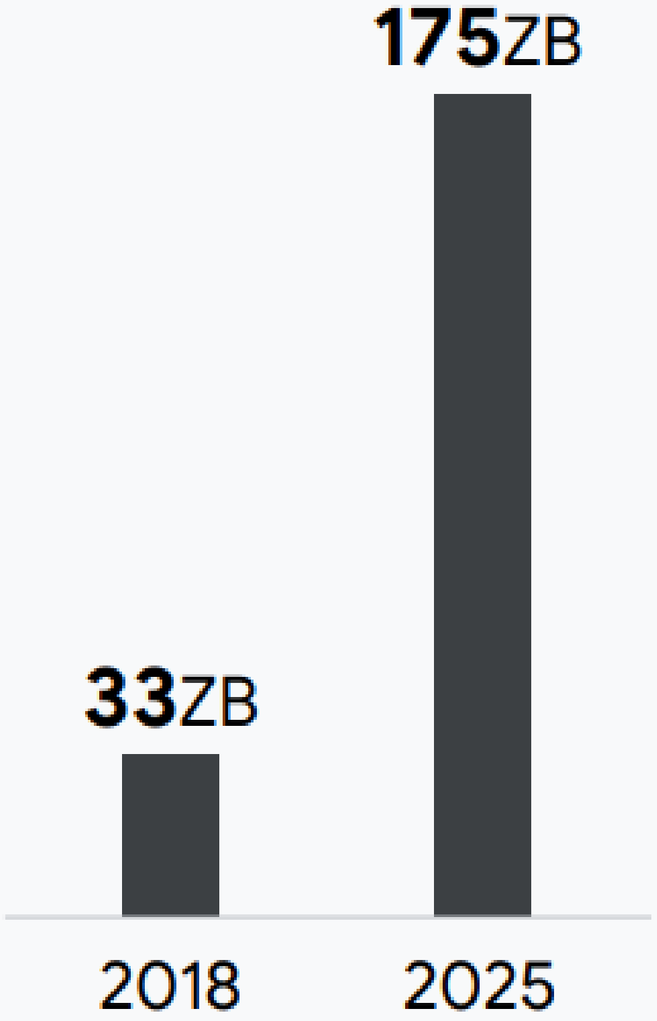
PROBLEM SOLVING

Problem Solving Stages



The world is generating more data than ever

By 2025, the world datasphere will be **175 zettabytes**.



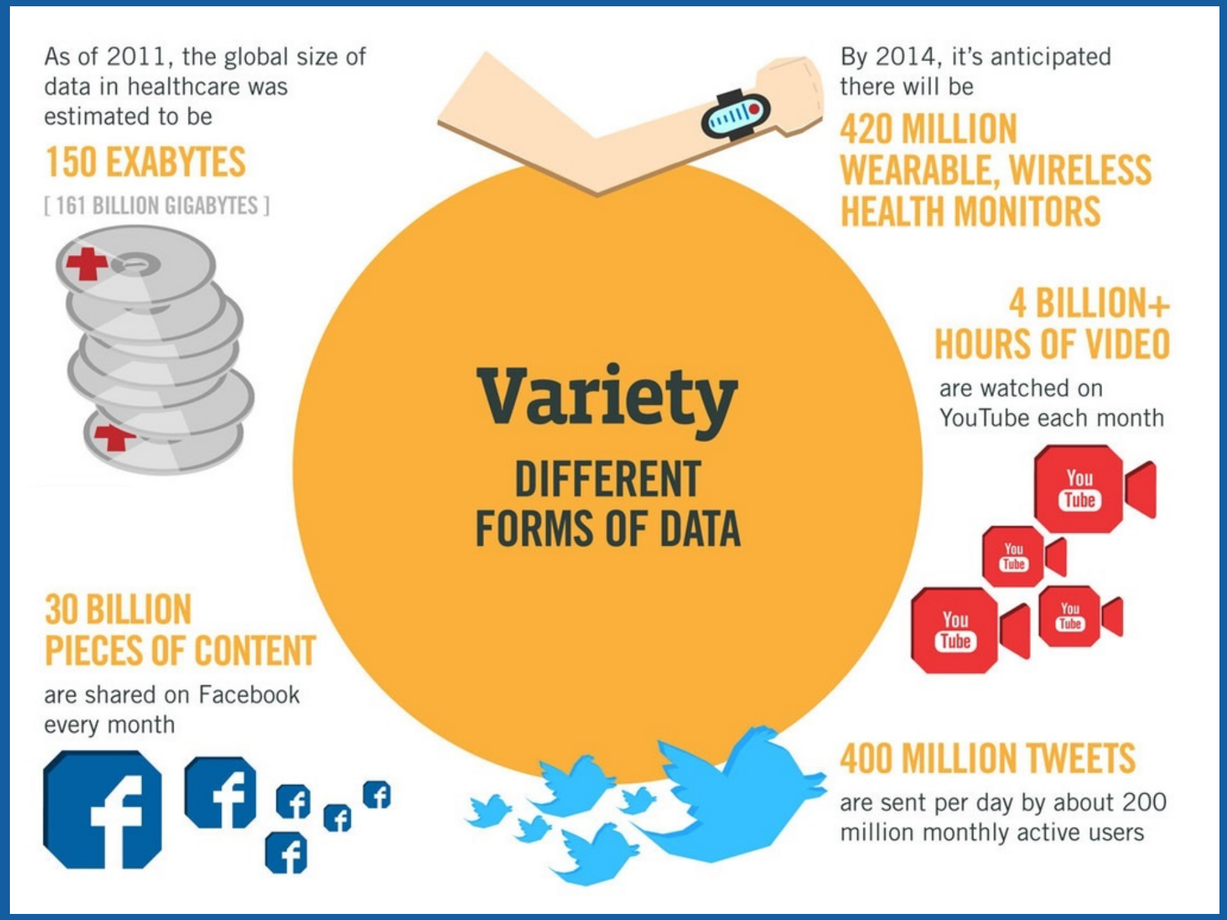
It all started with curiosity

About human spending **behavior**

4V of BIG DATA

Volume

Variety



4V of BIG DATA

Velocity

The New York Stock Exchange captures

1 TB OF TRADE INFORMATION

during each trading session

Modern cars have close to

100 SENSORS

that monitor items such as fuel level and tire pressure

Velocity

ANALYSIS OF STREAMING DATA

By 2016, it is projected there will be

18.9 BILLION NETWORK CONNECTIONS

– almost 2.5 connections per person on earth

Veracity

1 IN 3 BUSINESS LEADERS

don't trust the information they use to make decisions

Poor data quality costs the US economy around

$3.1 TRILLION A YEAR

in one survey were unsure of how much of their data was inaccurate

Veracity

UNCERTAINTY OF DATA

Technical Skills required for the Data Engineer role

Basic:

- Database architectures
- SQL-based technologies (e.g. PostgreSQL and MySQL)
- Data modeling tools (e.g. ERWin, Enterprise Architect and Visio)
- Extract Transform and Load (ETL) proficiency
- Python, C/C++ Java, Perl
- Data warehousing solutions

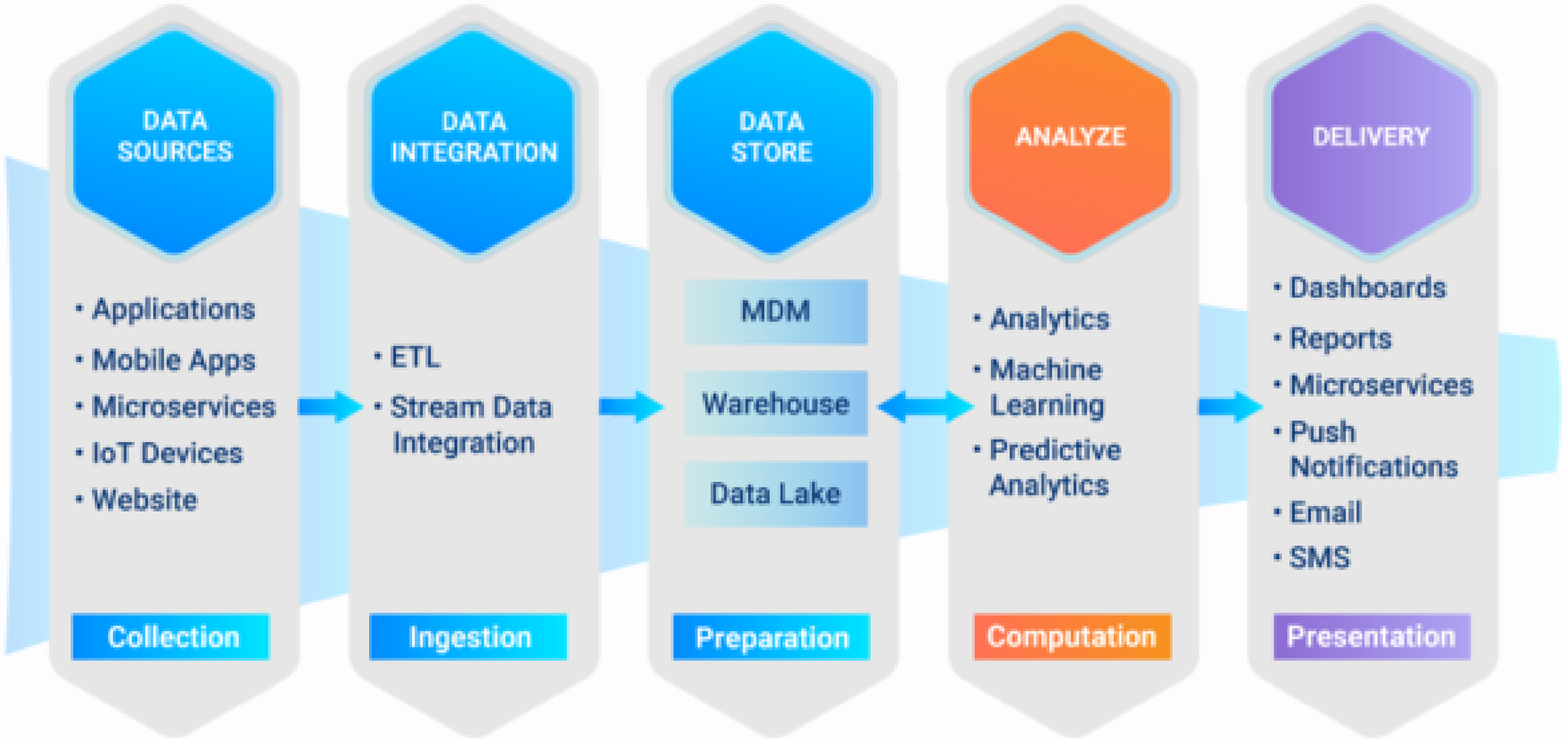
Advanced:

- NoSQL technologies (e.g. Cassandra and MongoDB)
- Hadoop-based technologies (e.g. MapReduce, Hive and Pig)
- Data mining
- Machine learning

Business Skills required for the Data Engineer role

- Creative Problem-Solving: Approaching data organization challenges with a clear eye on what is important; employing the right approach/methods to make the maximum use of time and human resources.
- Effective Collaboration: Carefully listening to management, data scientists and data architects to establish their needs.
- Intellectual Curiosity: Exploring new territories and finding creative and unusual ways to solve data management problems.
- Industry Knowledge: Understanding the way your chosen industry functions and how data can be collected, analyzed and utilized; maintaining flexibility in the face of big data developments.

Data Pipeline Automation: End-to-End Orchestration



DigitalSkola

What Big Data can achieve?

- Supports 1 million motorcycle drivers with rapid access to riders and optimized routes
- Enables demand forecasting and pricing adjustments
- Manages up to 5TB of data per day

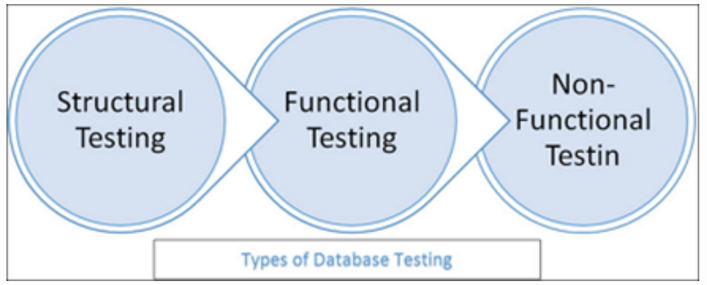
What Big Data can achieve?

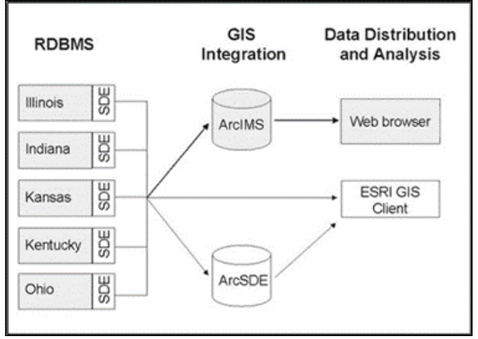
- Extends access to data warehousing and analysis to all relevant team members
- Delivers the seamless, cost-effective scalability needed to support business growth
- Analyzes customer ecommerce behavior

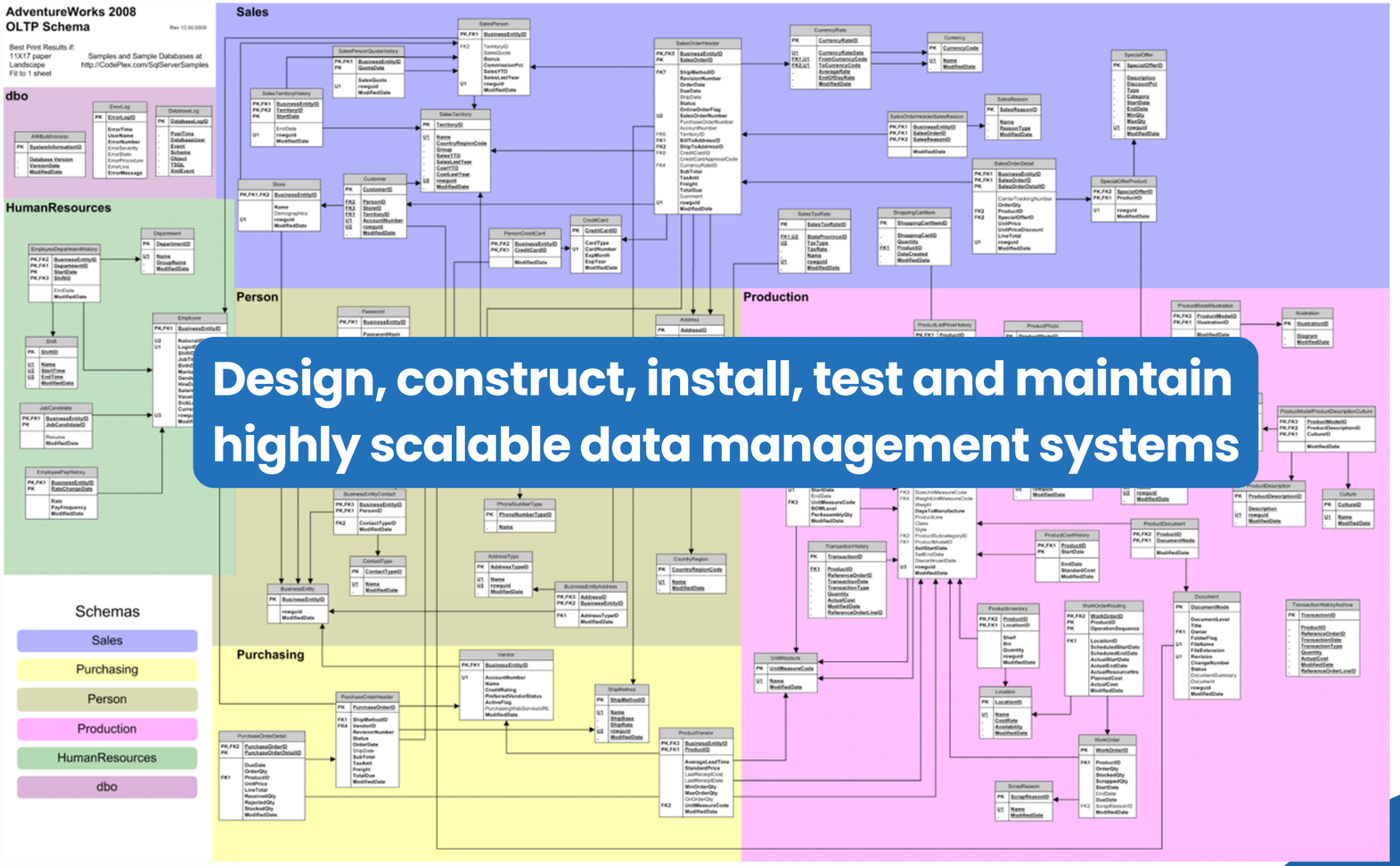
What Big Data can achieve?

- Processing 10 million records per day
- Established 3 critical dashboards within 24 hours to mitigate risks of COVID pandemic
- Prevented potential losses of 7.3 million transactions per week at each of the bank’s branches
- Reduced loan qualifications from 5 days to one day

Data Engineer Responsibilities (part 1)







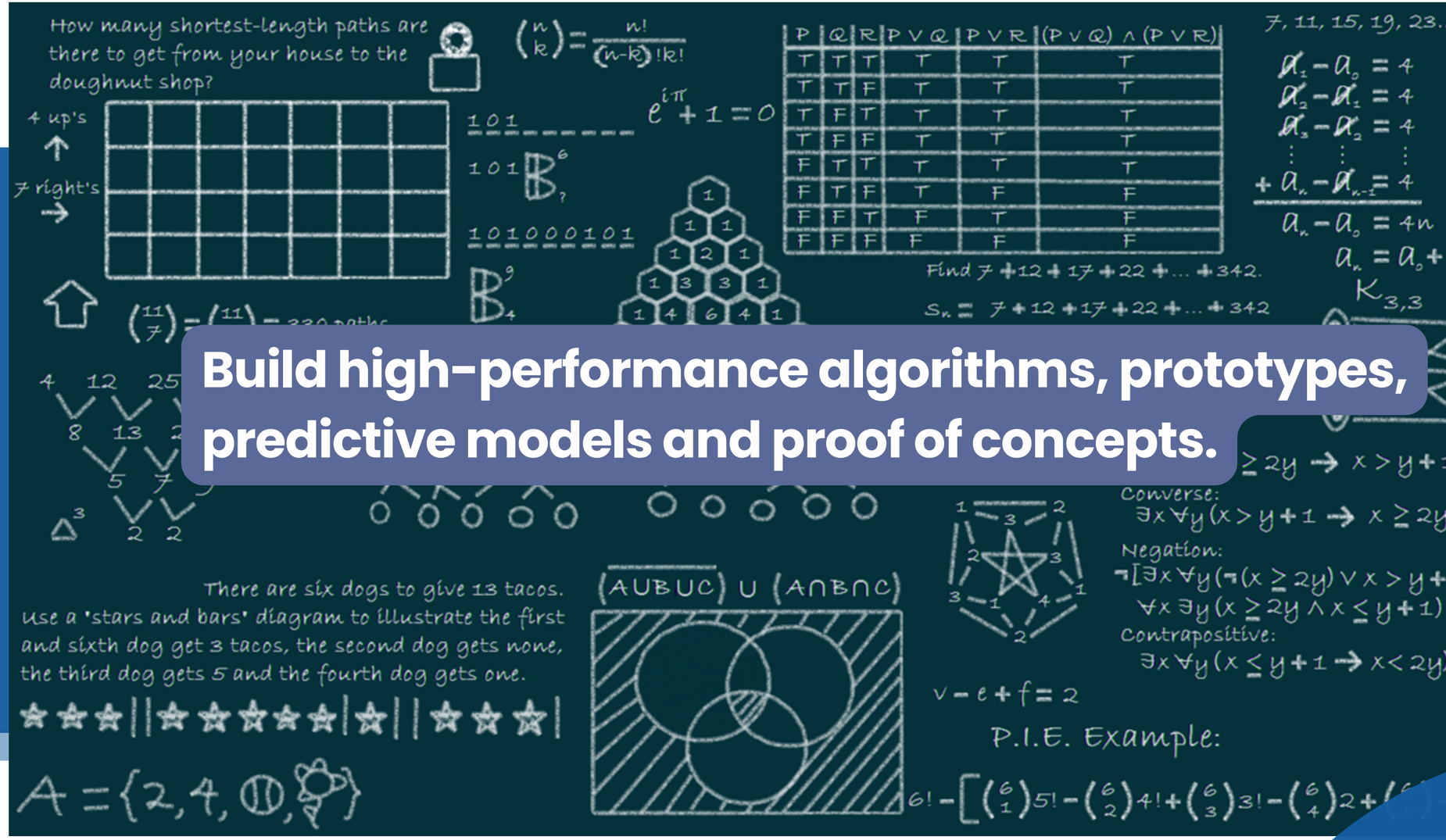
Design, construct, install, test and maintain highly scalable data management systems

Data Engineer Responsibilities (part 1)

Ensure systems meet business requirements and industry practices

DigitalSkola
Uncover The World of Digital Skills with Us

Data Engineer Responsibilities (part 1)

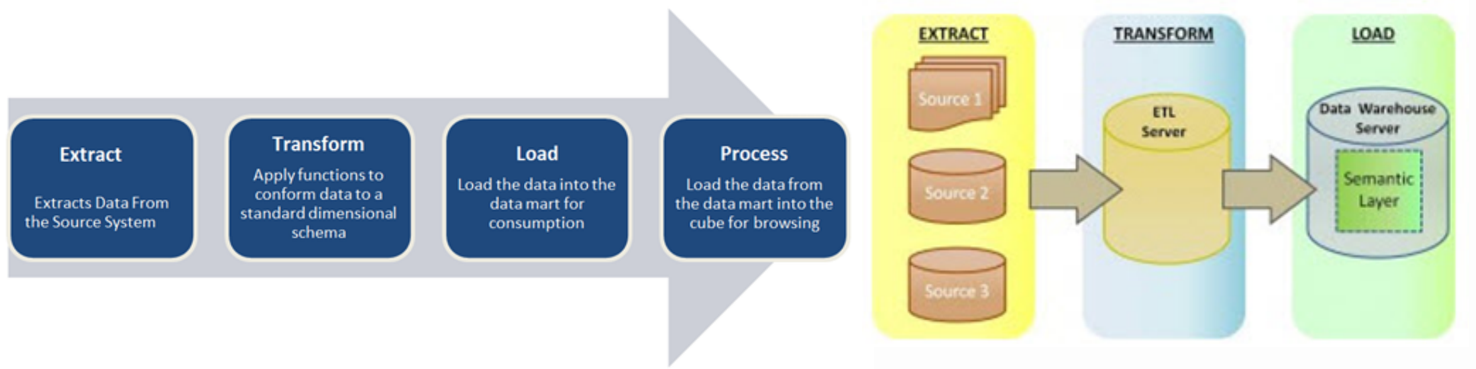


Build high-performance algorithms, prototypes, predictive models and proof of concepts.

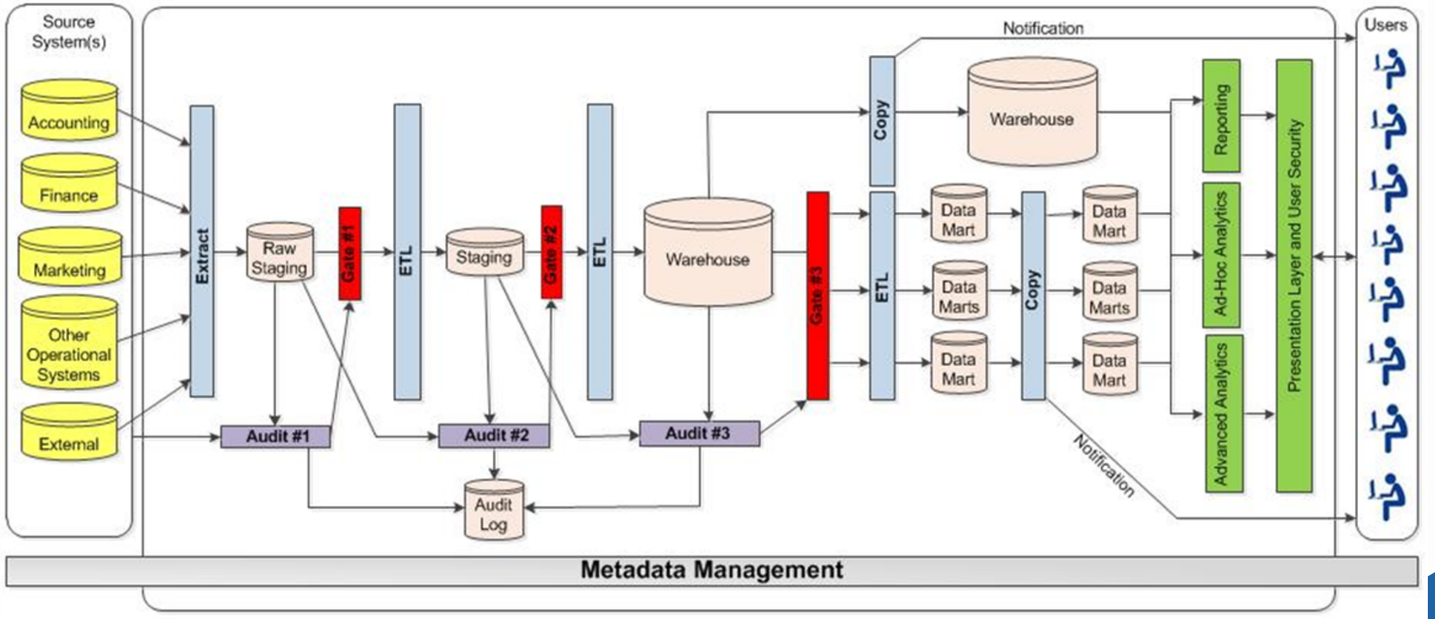
Data Engineer Responsibilities (part 1)

Research opportunities for data acquisition and new uses for existing data.

Data Engineer Responsibilities (part 2)



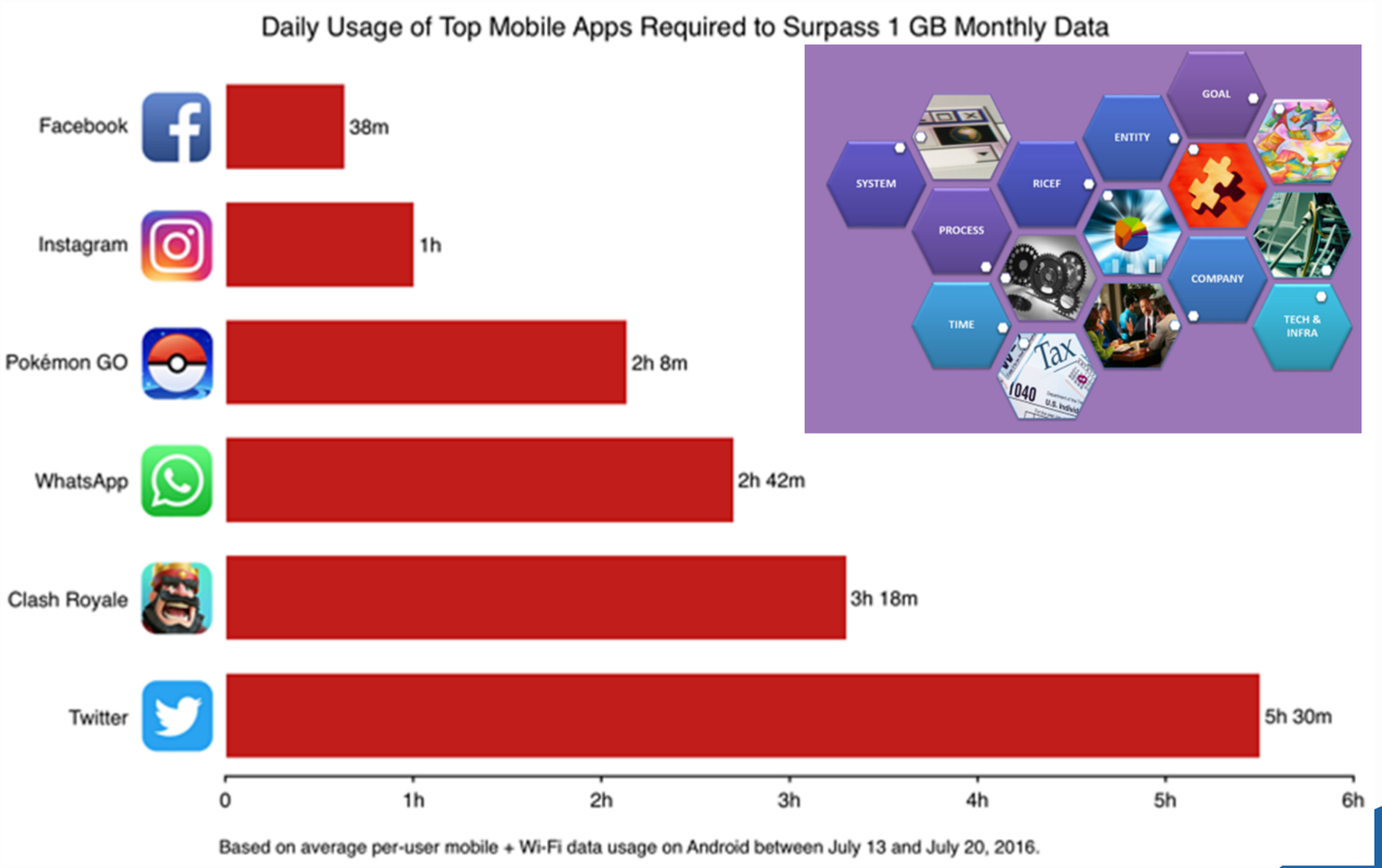
Develop data set processes for data modeling, mining and production



DigitalSkola
Uncover The World of Digital Skills with Us

Data Engineer Responsibilities (part 2)

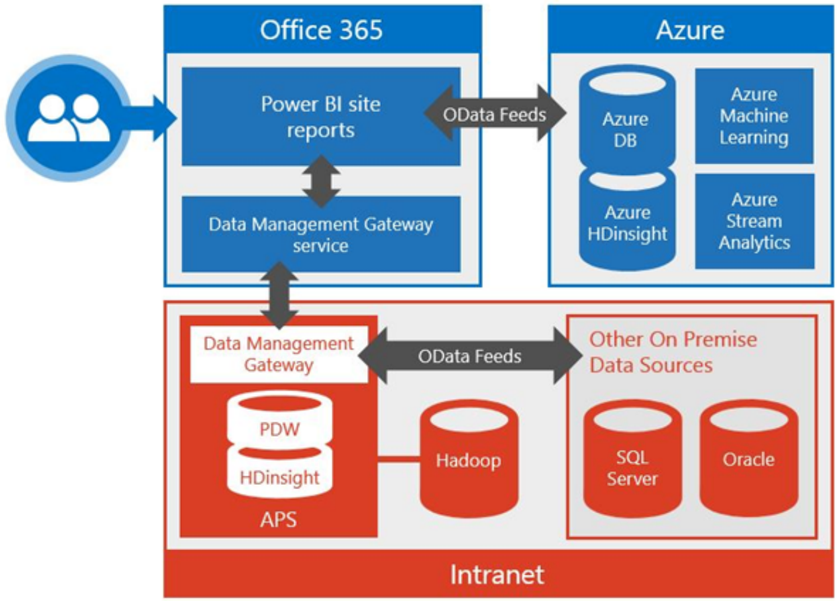
Integrate new data management technologies and software engineering tools into existing structures



DigitalSkola
Uncover The World of Digital Skills with Us

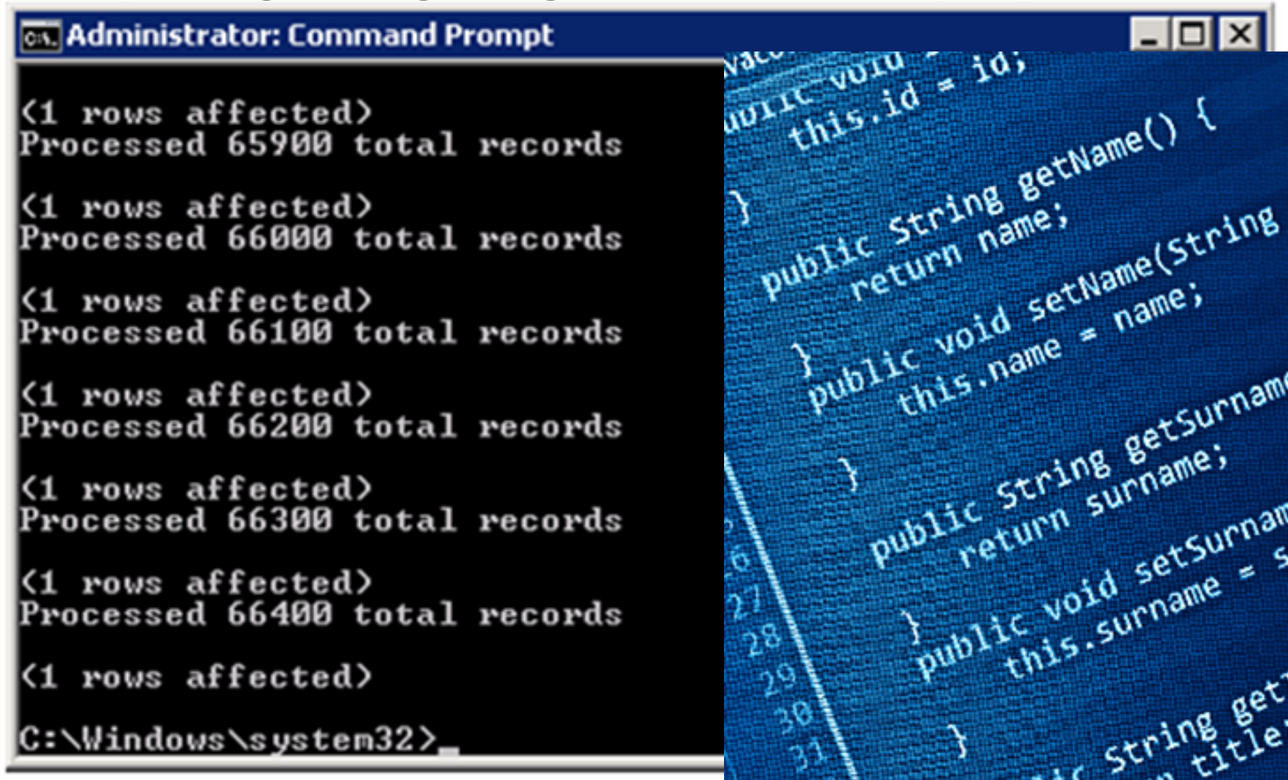
Data Engineer Responsibilities (part 2)

Create custom software components (e.g. specialized UDFs) and analytics applications

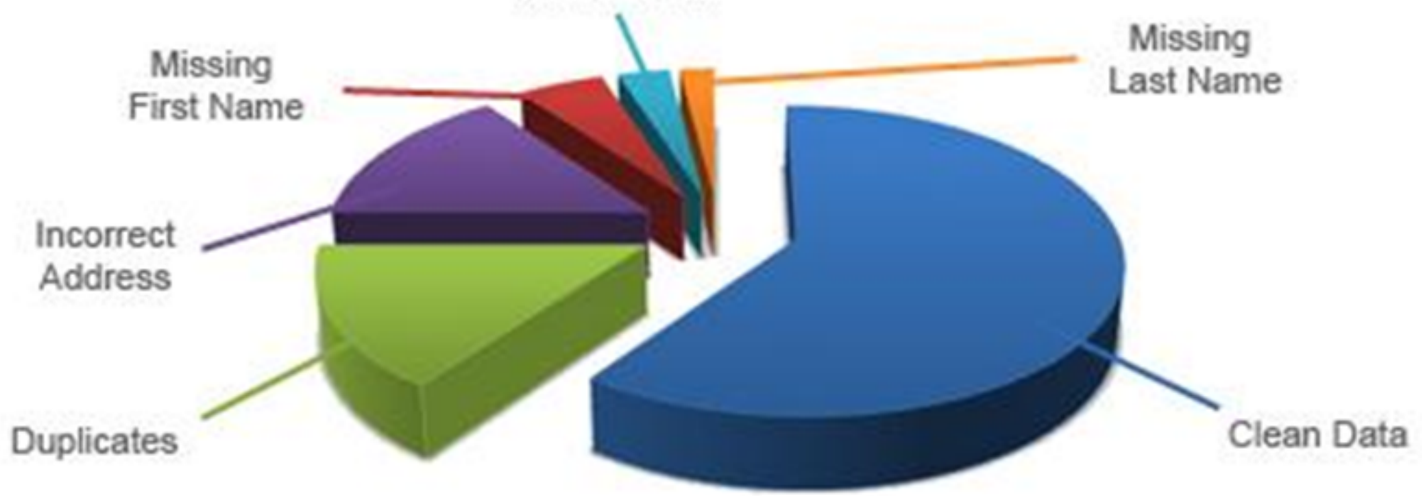


Data Engineer Responsibilities (part 2)

Employ a variety of languages and tools (e.g. scripting languages) to marry systems together



Data Engineer Responsibilities (part 3)

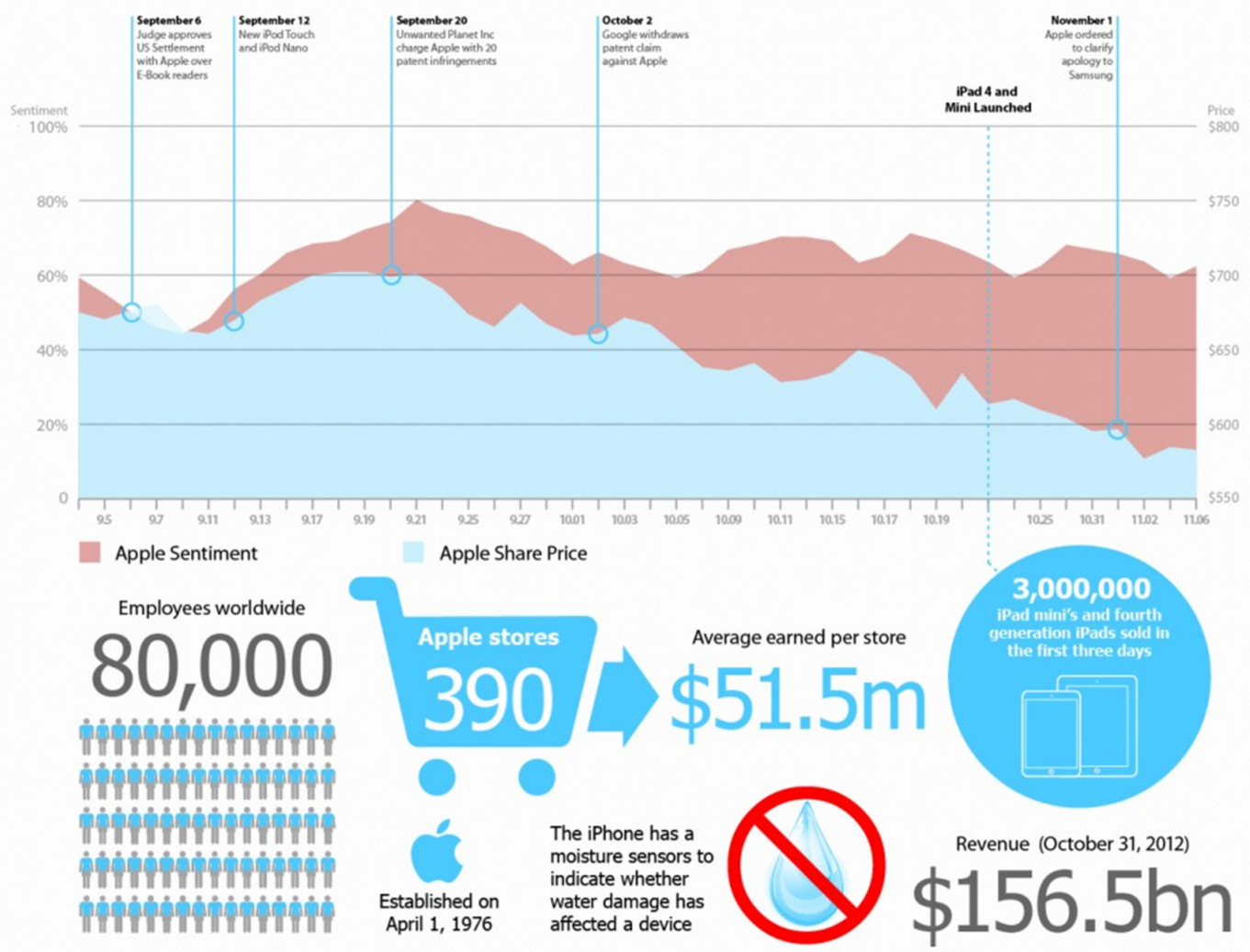


Clean and prune data to discard irrelevant information.

DigitalSkola
Uncover The World of Digital Skills with Us

Data Engineer Responsibilities (part 3)

Analyze and interpret results using standard statistical tools and techniques



DigitalSkola
Uncover The World of Digital Skills with Us

Data Engineer Responsibilities (part 3)

Recommend ways to improve data reliability, efficiency and quality

Data Engineer Responsibilities (part 3)

Collaborate with data architects, modelers and IT team members on project goals

DigitalSkola
Uncover The World of Digital Skills with Us

PROJECT 3

Target Pencapaian

- Bisa membuat ETL sederhana.
- Mengasah skill di sesi-sesi sebelumnya seperti Git, Python, Database dan Docker.

Tools yang harus di install:

- Python
- Vscode/pycharm
- Postgres SQL
- GUI SQL (Dbeaver, Pgadmin, etc)
- Git
- Sourcetree (Optional)

Requirements:

- Python ≥ 3.7
- Packages:
  - Psycopg2-binary==2.9.3
  - SQLAlchemy==1.4.40
  - Sqlparse==0.4.2
  - Pandas==1.4.3

Bahan Project:

- https://drive.google.com/drive/folders/1leg8hIQw6t2qjLxFmKSZcK1XIW5DRIXZ?usp=sharing.
- Siapkan Repository di Github.
- Postgres SQL (disiapkan oleh tutor)

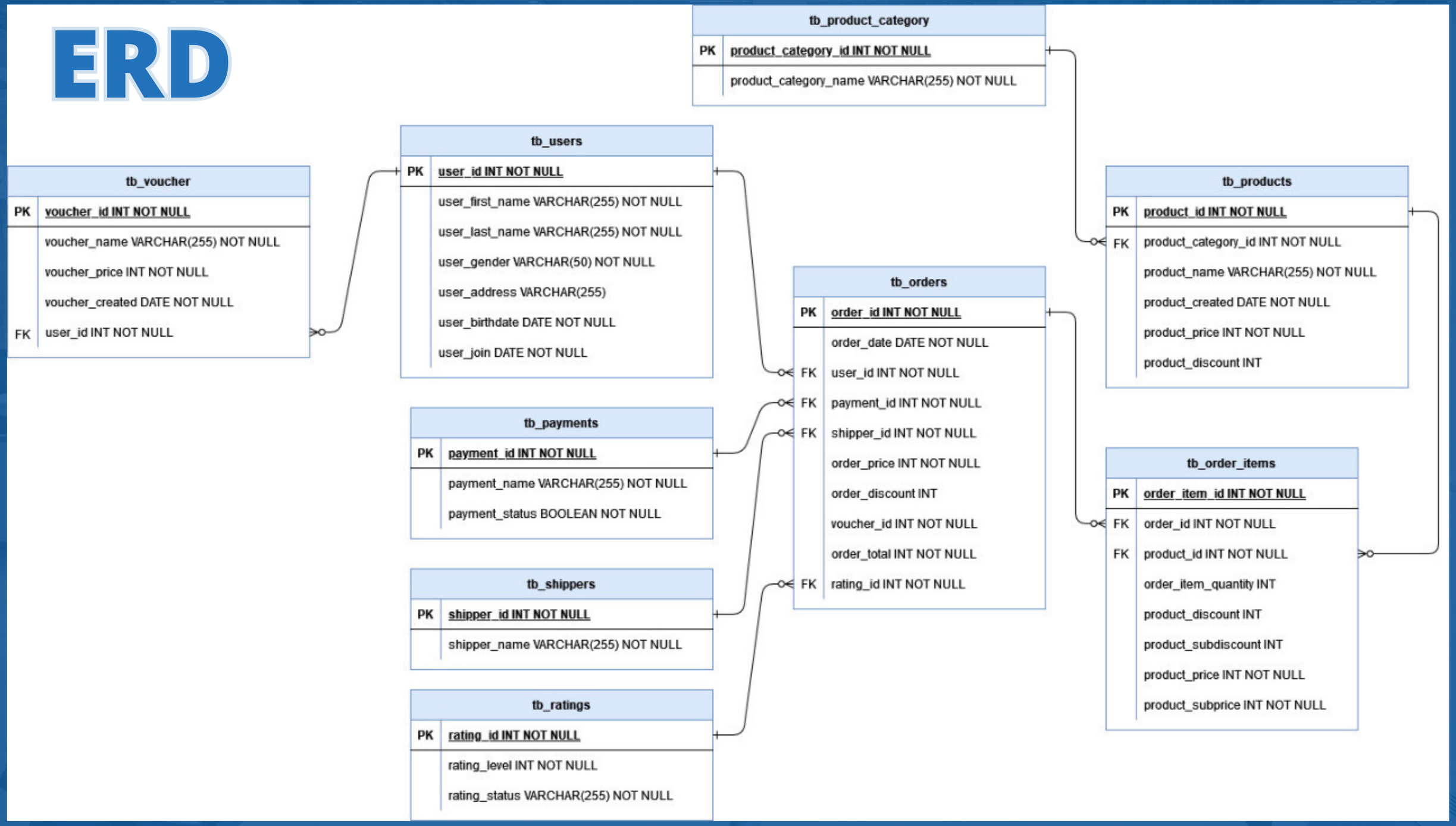
DigitalSkola
Uncover The World of Digital Skills with Us

Background Project

Tim data analyst membutuhkan tabel untuk membuat dashboard terkait detail order dari data source yang ada di database production marketplace, Anda sebagai data engineer di minta untuk membuat script data migrasi dari tabel source tersebut ke data warehouse, sehingga tim data analyst bisa menggunakan table dari data warehouse tanpa membebankan database production marketplace.

DigitalSkola

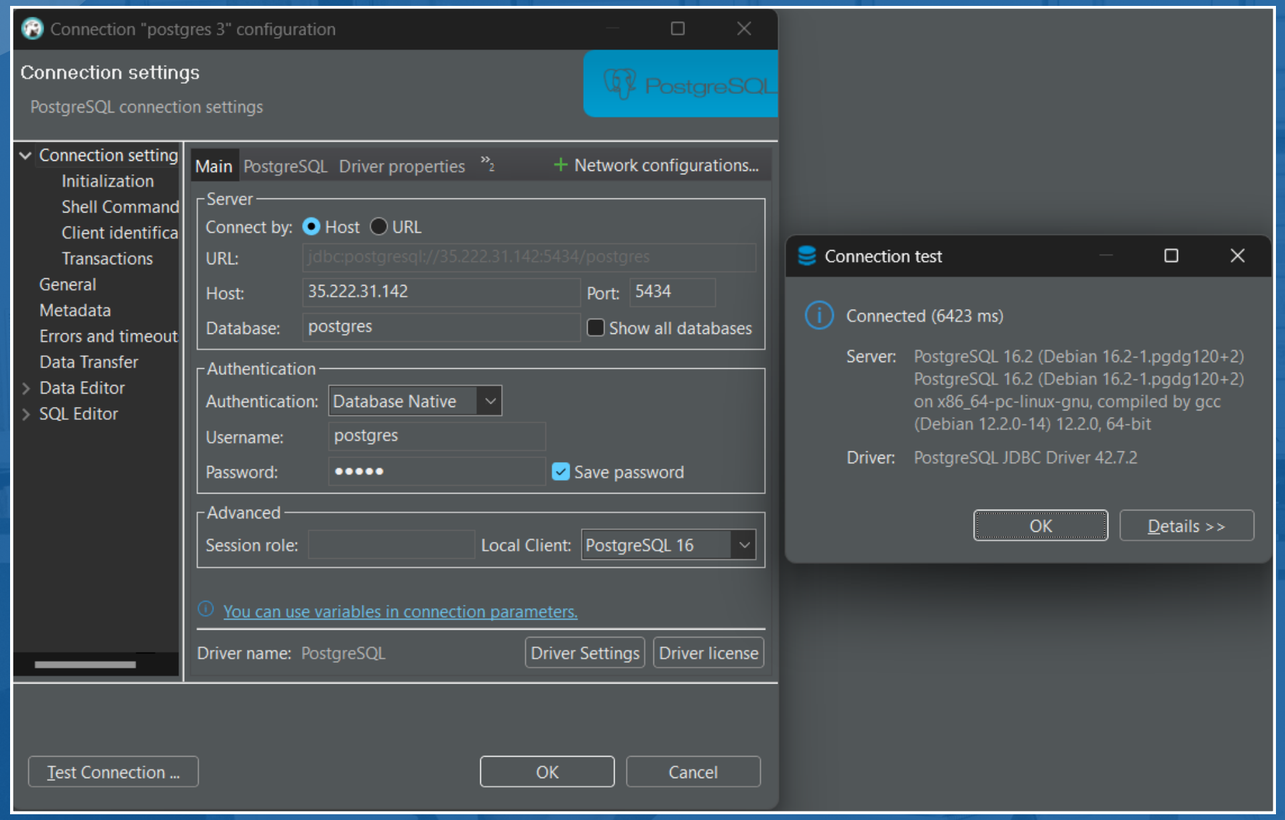
ERD


tb_product_category
PK product_category_id INT NOT NULL
product_category_name VARCHAR(255) NOT NULL
tb_users
PK user_id INT NOT NULL
user_first_name VARCHAR(255) NOT NULL
user_last_name VARCHAR(255) NOT NULL
user_gender VARCHAR(50) NOT NULL
user_address VARCHAR(255)
user_birthdate DATE NOT NULL
user_join DATE NOT NULL
tb_voucher
PK voucher_id INT NOT NULL
voucher_name VARCHAR(255) NOT NULL
voucher_price INT NOT NULL
voucher_created DATE NOT NULL
FK user_id INT NOT NULL
tb_products
PK product_id INT NOT NULL
FK product_category_id INT NOT NULL
product_name VARCHAR(255) NOT NULL
product_created DATE NOT NULL
product_price INT NOT NULL
product_discount INT
tb_orders
PK order_id INT NOT NULL
order_date DATE NOT NULL
FK user_id INT NOT NULL
FK payment_id INT NOT NULL
FK shipper_id INT NOT NULL
order_price INT NOT NULL
order_discount INT
voucher_id INT NOT NULL
order_total INT NOT NULL
FK rating_id INT NOT NULL
tb_payments
PK payment_id INT NOT NULL
payment_name VARCHAR(255) NOT NULL
payment_status BOOLEAN NOT NULL
tb_order_items
PK order_item_id INT NOT NULL
FK order_id INT NOT NULL
FK product_id INT NOT NULL
order_item_quantity INT
product_discount INT
product_subdiscount INT
product_price INT NOT NULL
product_subprice INT NOT NULL
tb_shippers
PK shipper_id INT NOT NULL
shipper_name VARCHAR(255) NOT NULL
tb_ratings
PK rating_id INT NOT NULL
rating_level INT NOT NULL
rating_status VARCHAR(255) NOT NULL

Requirement Kolom - Kolom yang Dibutuhkan oleh Tim Data Analyst.

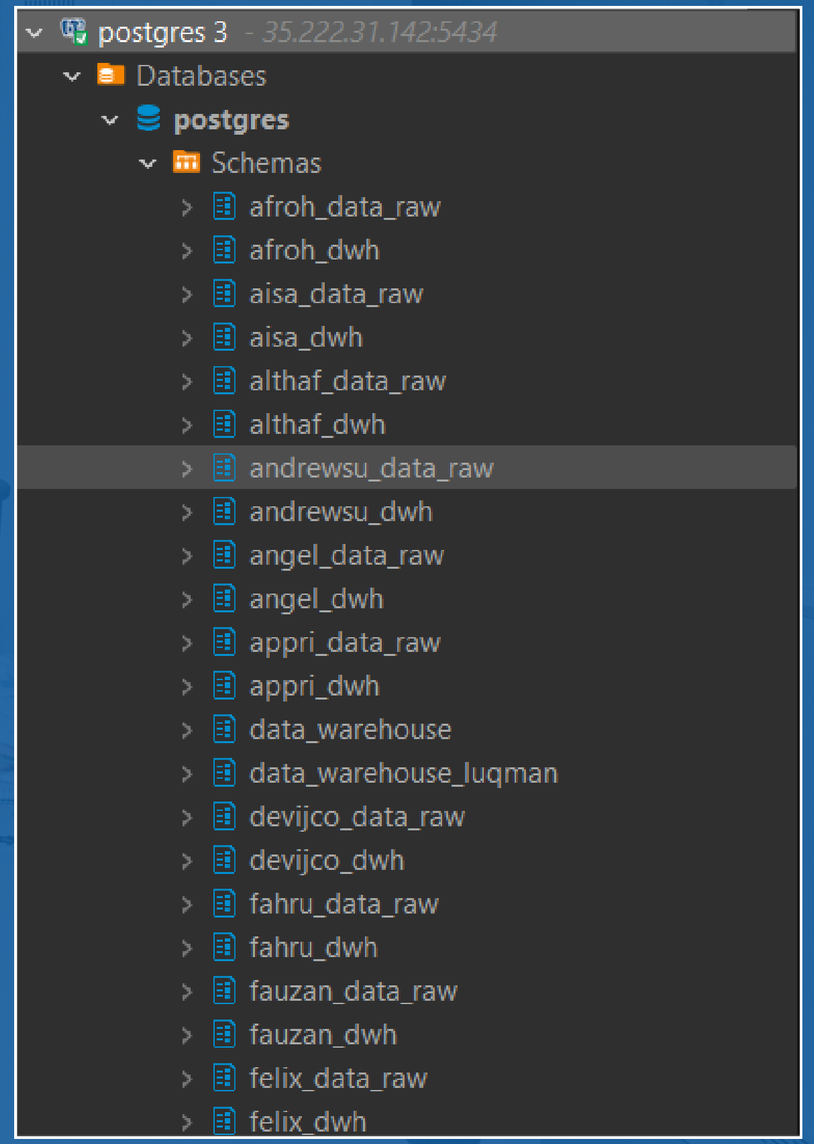
```
order_id INT NOT NULL,
order_date DATE NOT NULL,
user_id INT NOT NULL,
payment_name VARCHAR(255),
shipper_name VARCHAR(255),
order_price INT,
order_discount INT,
voucher_name VARCHAR(255),
voucher_price INT,
order_total INT,
rating_status VARCHAR(255)
```

Step 0: Connect PostgresSQL Menggunakan DBeaver



Akses PostgresSQL:
host: 35.222.31.142
username: postgres
password: admin
port: 5434
database: postgres

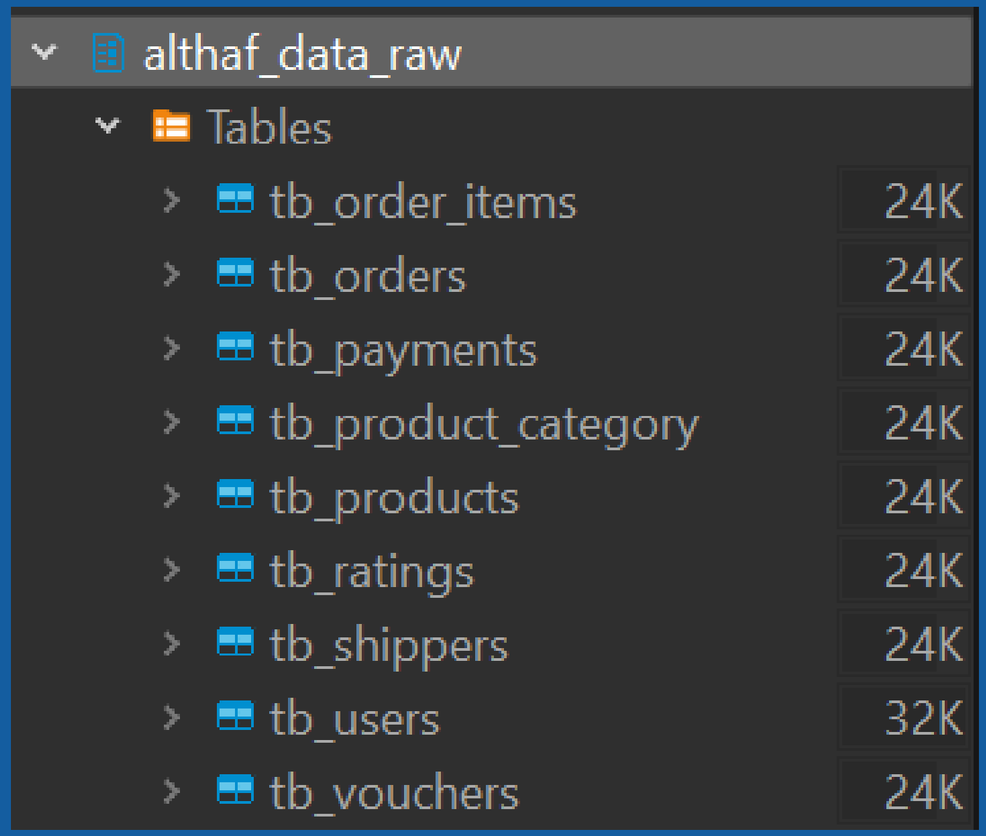
Buat Schema Baru:
- **nama_data_raw**
- **nama_dwh**



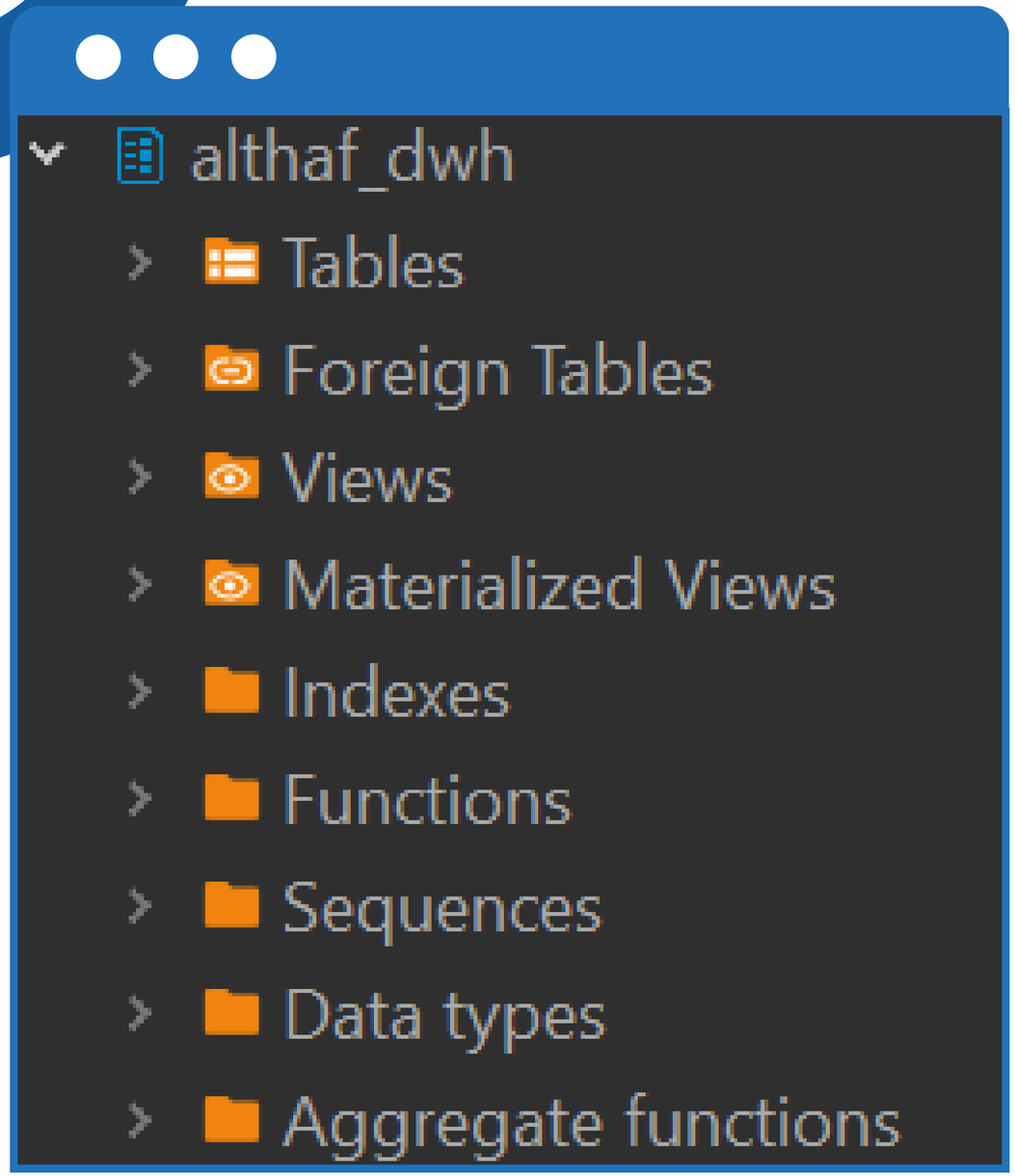
Step 1: Buat schema pada Postgres SQL sebagai raw data

```sql
CREATE TABLE tb_users (
    user_id INT NOT NULL,
    user_first_name VARCHAR(255) NOT NULL,
    user_last_name VARCHAR(255) NOT NULL,
    user_gender VARCHAR(50) NOT NULL,
    user_address VARCHAR(255),
    user_birthday DATE NOT NULL,
    user_join DATE NOT NULL,
    PRIMARY KEY (user_id)
);

CREATE TABLE tb_payments (
    payment_id INT NOT NULL,
    payment_name VARCHAR(255) NOT NULL,
    payment_status BOOLEAN NOT NULL,
    PRIMARY KEY (payment_id)
);

CREATE TABLE tb_shippers (
    shipper_id INT NOT NULL,
    shipper_name VARCHAR(255) NOT NULL,
    PRIMARY KEY (shipper_id)
);

CREATE TABLE tb_ratings (
    rating_id INT NOT NULL,
    rating_level INT NOT NULL,
    rating_status VARCHAR(255) NOT NULL,
    PRIMARY KEY (rating_id)
);

CREATE TABLE tb_product_category (
    product_category_id INT NOT NULL,
    product_category_name VARCHAR(255) NOT NULL,
    PRIMARY KEY (product_category_id)
);

CREATE TABLE tb_vouchers (
    voucher_id INT NOT NULL,
    voucher_name VARCHAR(255) NOT NULL,
    voucher_price INT,
    voucher_created DATE NOT NULL,
    user_id INT NOT NULL,
    PRIMARY KEY (voucher_id),
    CONSTRAINT fk_user_id FOREIGN KEY (user_id) REFERENCES tb_users (user_id)
);
```

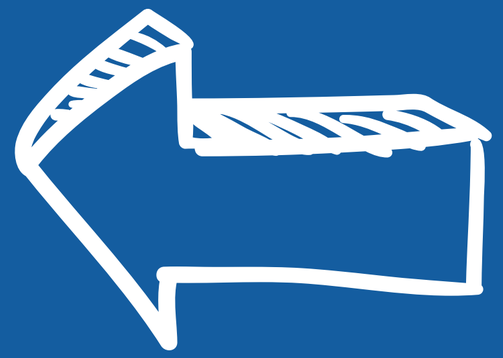
Query to create table on postgres.



Terdapat 9 tabel pada data raw tersebut, yaitu order_items, orders, payments, product_category, products, ratings, shippers, users, dan vouchers.

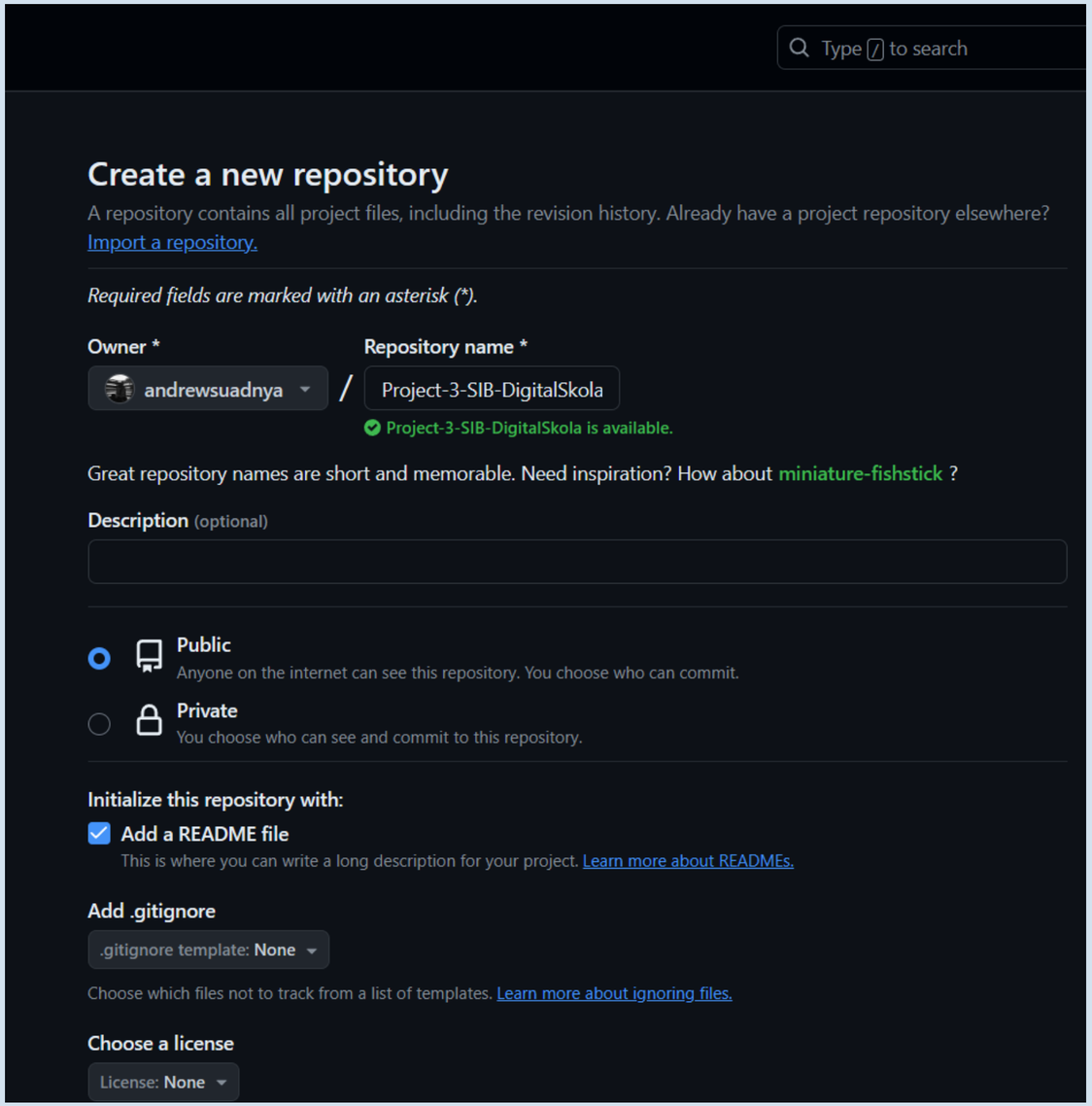


Step 2: Siapkan Schema Untuk Data Warehouse



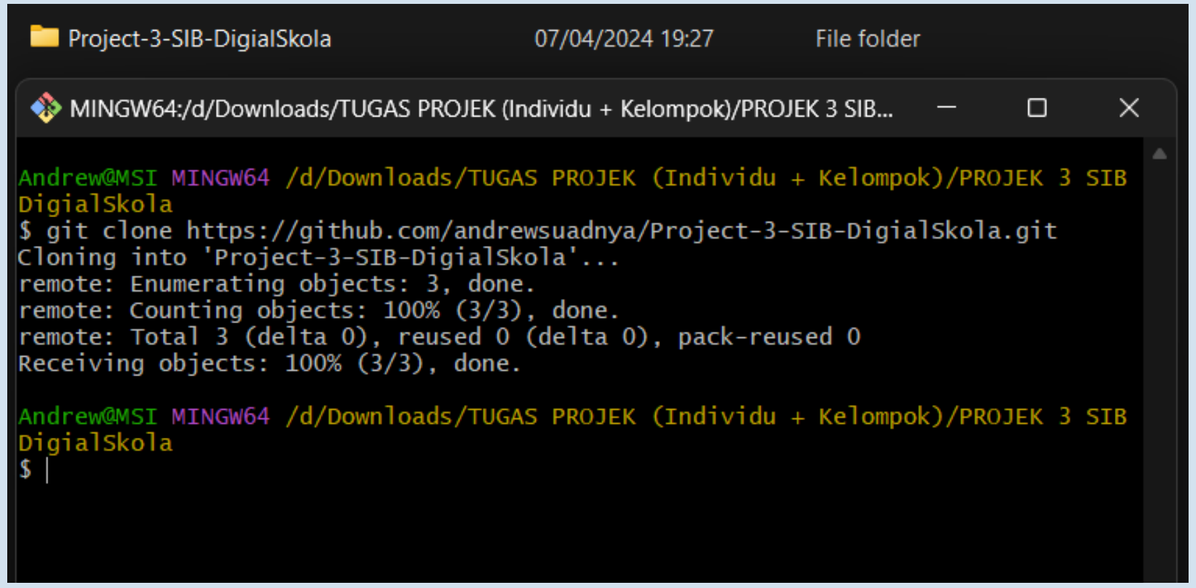
Step 3: Buat Repository GitHub

GITHUB REPOSITORY



Membuat repository GitHub dengan nama “Project-3-SIB-DigialSkola” dengan men-set ke “Public”.

Kemudian Gitnya di clone ke lokal path menggunakan Git Bash dengan perintah “git clone (link repo GitHub)” dan akan muncul folder baru dengan nama “Project-3-SIB-DigialSkola”.



Step 4: Install requirements yang diperlukan menggunakan VS Code

Dengan membuka folder clone dari GitHub repo sebelumnya.

```
psycopg2-binary==2.9.3
SQLAlchemy==1.4.40
sqlparse==0.4.2
pandas==1.4.3
```

Buat file “requirements.txt” kemudian Install requirements dengan command pip.

```
pip install -r requirements.txt
```

Step 5: Set Up Connection ke Postgres

{} config.json > {} dwh > abc port

```
1   {
2       "marketplace_prod": {
3           "host": "35.222.31.142",
4           "db": "postgres",
5           "user": "postgres",
6           "password": "admin",
7           "port": "5434"
8       },
9       "dwh": {
10          "host": "35.222.31.142",
11          "db": "postgres",
12          "user": "postgres",
13          "password": "admin",
14          "port": "5434"
15      }
16  }
```

config.json

Buat file “.gitignore” kemudian masukan “*config.json” agar file config.json tidak ter push ke repo GitHub.

File config.json digunakan untuk konfigurasi database marketplace_prod (data raw) dan data warehouse.

```python
import os
import json
import psycopg2
from sqlalchemy import create_engine

def config(connection_db):
    path = os.getcwd()
    with open(path+'/config.json') as file:
        conf = json.load(file)[connection_db]
    return conf

def get_conn(conf, name_conn):
    try:
        conn = psycopg2.connect(
            host=conf['host'],
            database=conf['db'],
            user=conf['user'],
            password=conf['password'],
            port=conf['port']
        )
        print(f'[INFO] success connect to postgress {name_conn}')
        engine = create_engine(f"postgresql+psycopg2://{conf['user']}:{co
        return conn, engine
    except Exception as e:
        print(f'[INFO] cannot connect to postgress {name_conn}')
        print(str(e))
```

Step 6: Set Up Connection ke Postgres

File connection.py digunakan untuk mengatur koneksi ke database PostgreSQL. File ini berisi dua fungsi utama, config dan get_conn.

Fungsi config membaca file config.json dari direktori kerja saat ini dan mengambil konfigurasi database berdasarkan parameter connection_db. Fungsi get_conn mencoba membuat koneksi ke database PostgreSQL menggunakan konfigurasi yang diberikan.

```sql
DROP TABLE IF EXISTs althaf_dwh.dim_orders;
CREATE TABLE althaf_dwh.dim_orders (
    order_id INT NOT NULL,
    order_date DATE NOT NULL,
    user_id INT NOT NULL,
    payment_name VARCHAR(255),
    shipper_name VARCHAR(255),
    order_price INT,
    order_discount INT,
    voucher_name VARCHAR(255),
        voucher_price INT,
    order_total INT,
        rating_status VARCHAR(255)
);
```

Step 7: Buat Query SQL Untuk Design Data Warehouse dan Query untuk Mengambil Data dari Schema Data Raw.

Query sql untuk schema data warehouse.

Step 8: Buat Query SQL Untuk Design Data Warehouse dan Query untuk Mengambil Data dari Schema Data Raw.

```sql
select order_id,
            order_date,
            a.user_id,
            b.payment_name,
            c.shipper_name,
            order_price,
            order_discount,
            d.voucher_name,
            d.voucher_price,
            order_total,
            e.rating_status
from althaf_data_raw.tb_orders a
        left join althaf_data_raw.tb_payments b on a.payment_id = b.payment_id
        left join althaf_data_raw.tb_shippers c on a.shipper_id = c.shipper_id
        left join althaf_data_raw.tb_vouchers d on a.voucher_id = d.voucher_id
        left join althaf_data_raw.tb_ratings e on a.rating_id = e.rating_id ;
```

Query SQL untuk mengambil data dari schema data raw.

main.py

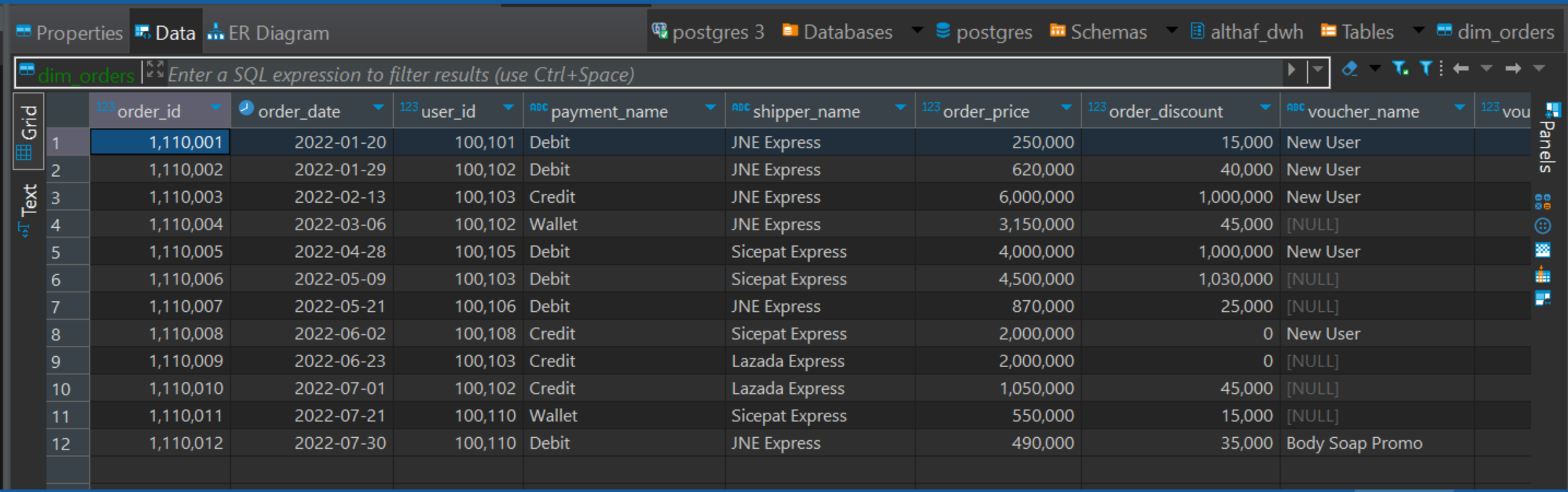
```python
import connection
import os
import sqlparse
import pandas as pd

if __name__ == '__main__':
    #connection data source
    conf = connection.config('marketplace_prod')
    conn, engine = connection.get_conn(conf, 'DataSource')
    cursor = conn.cursor()

    #connection data source
    conf = connection.config('dwh')
    conn_dwh, engine_dwh = connection.get_conn(conf, 'DataWarehouse')
    cursor_dwh = conn_dwh.cursor()

    #get query string
    path_query = os.getcwd()+'/query/'
    query = sqlparse.format(
        open(path_query+'query.sql', 'r').read(), strip_comments=True
    ).strip()

    #get schema dwh design
    dwh_design = sqlparse.format(
        open(path_query+'dwh_design.sql', 'r').read(), strip_comments=True
    ).strip()
```

Step 9: Melakukan Proses ETL dari Sumber Data ke Data Warehouse (DWH).

Script main.py ini mencakup pembuatan koneksi ke sumber data dan DWH, membaca query, ekstraksi data, pembuatan schema DWH, pemuatan data ke DWH, serta penanganan Error.

```python
try:
    #get data
    print('[INFO] service ETL is running...')
    df = pd.read_sql(query, engine)

    #create schema dwh
    cursor_dwh.execute(dwh_design)
    conn_dwh.commit()

    #ingest data to dwh
    df.to_sql('dim_orders', engine_dwh, schema='althaf_dwh', if_exists='append', index=False)
    print('[INFO] service ETL is success..')
except Exception as e:
    print('[INFO] service ETL is failes')
    print(str(e))
```

Tampilan Tabel `dim_orders` Pada Schema Data Warehouse



Properties | Data | ER Diagram | postgres 3 | Databases | postgres | Schemas | althaf_dwh | Tables | dim_orders

dim_orders | Enter a SQL expression to filter results (use Ctrl+Space)

Grid | Text | Panels

	order_id	order_date	user_id	payment_name	shipper_name	order_price	order_discount	voucher_name	vou
1	1,110,001	2022-01-20	100,101	Debit	JNE Express	250,000	15,000	New User	
2	1,110,002	2022-01-29	100,102	Debit	JNE Express	620,000	40,000	New User	
3	1,110,003	2022-02-13	100,103	Credit	JNE Express	6,000,000	1,000,000	New User	
4	1,110,004	2022-03-06	100,102	Wallet	JNE Express	3,150,000	45,000	[NULL]	
5	1,110,005	2022-04-28	100,105	Debit	Sicepat Express	4,000,000	1,000,000	New User	
6	1,110,006	2022-05-09	100,103	Debit	Sicepat Express	4,500,000	1,030,000	[NULL]	
7	1,110,007	2022-05-21	100,106	Debit	JNE Express	870,000	25,000	[NULL]	
8	1,110,008	2022-06-02	100,108	Credit	Sicepat Express	2,000,000	0	New User	
9	1,110,009	2022-06-23	100,103	Credit	Lazada Express	2,000,000	0	[NULL]	
10	1,110,010	2022-07-01	100,102	Credit	Lazada Express	1,050,000	45,000	[NULL]	
11	1,110,011	2022-07-21	100,110	Wallet	Sicepat Express	550,000	15,000	[NULL]	
12	1,110,012	2022-07-30	100,110	Debit	JNE Express	490,000	35,000	Body Soap Promo	

THANK YOU!

~ DREAM