

BAB III PELAKSANAAN KERJA PROFESI

3.1 Bidang Kerja

Dalam pelaksanaan kerja profesi di PT. Bank Syariah Indonesia yang berlokasi di Gedung Graha Mandiri Lt. 10, Jl. Imam Bonjol No. 61, Menteng, Jakarta Pusat. Praktikan ditempatkan pada Internal Audit Departemen *MIS & Data Analytics* di divisi *Data Analytics* yang menjalankan fungsi :

1. Membantu tugas *Departemen Head* dan *Group Head* dalam melakukan pengawasan dengan cara menjabarkan secara operasional baik perencanaan, pelaksanaan, maupun pemantauan hasil audit.
2. Melakukan monitoring tindak lanjut atas temuan dari hasil Audit kepada *Regional Branch Manager*.
3. Memastikan ketersediaan, kesesuaian dan kevalidan data dengan lancar dengan melakukan pembaharuan informasi secara berkala.
4. Meningkatkan pengetahuan organisasi *Internal Audit Group 2* dengan melakukan berbagi pengetahuan setiap bulan.

Dalam pelaksanaan kerja profesi, Praktikan menjabat sebagai *Data Analytics Auditor*. Praktikan melapor langsung kepada *Team Leader Data Analytics* yaitu bapak Tonny Harry Prasetyo dan Bapak Hidayat Tri Wahyudi sebagai *Department Head MIS & Data Analytics* sekaligus pembimbing kerja selama masa kerja profesi.

Adapun deskripsi pekerjaan atau *role* yang harus dilakukan praktikan pada divisi *Data Analytics* diantaranya :

1. Mengumpulkan, mengidentifikasi dan menyusun *risk event* yang akan menjadi objek analisa data dan *continuous monitoring*.
2. Menyusun dan mengusulkan penambahan dan atau *penggantian risk event* yang selaras dengan kondisi atau isu-isu terkini.

3. Melakukan analisa data atas hasil *continuous monitoring* dan hasil query mandiri kemudian menyusun dan menyampaikan laporan hasil analisa tersebut beserta rekomendasi perbaikannya.
4. Memonitor berfungsinya aplikasi *continuous monitoring* dengan baik dari sisi kehandalan aplikasi dan kualitas hasil query.
5. Melakukan pengajuan usulan pengembangan aplikasi *continuous monitoring* dan *data analytic* yang sedang digunakan.

Dalam menjalani profesi yang dijalankan oleh praktikan harus memiliki keahlian dan kompetensi kerja diantaranya :

1. Mempunyai keahlian teknis terkait dengan model data, proses pengembangan struktur database
2. Mempunyai keahlian teknis untuk melakukan pengolahan data dari berbagai sumber dan bentuk data
3. Memiliki kemampuan analisis yang kuat dengan kemampuan mengumpulkan, mengorganisasikan, menganalisa, dan menyebarkan sejumlah besar informasi dengan memperhatikan detail dan keakuratannya
4. Menguasai pertanyaan, menulis laporan dan menyajikan hasil temuan baik dari hasil pengolahan data amupun dari hasil tindak lanjut yang dilakukan oleh auditor

Seorang *Data Analytics* tidak hanya berhubungan dengan data dan infrastruktur saja melainkan harus bisa melakukan hubungan kerja yang profesional dengan lintas fungsional kerja lainnya, selain itu praktikan dituntut untuk dapat berkomunikasi dengan baik terhadap *stackholder* guna mempresentasikan hasil dari pengolahan data yang dilakukan.

Tantangan yang dihadapi pun sangat besar, karena selain harus dapat memastikan analisis data sesuai dengan inisiasi atas *risk event* yang sudah ditentukan, seorang *Data Analytics* juga diharuskan untuk dapat secara disiplin meningkatkan pengetahuan di bidang teknologi serta meminimalisir terjadinya kesenjangan antara teknologi yang digunakan perusahaan dengan yang saat ini tengah berkembang di masyarakat.

Manajer proyek juga harus dapat mempertanggungjawabkan pelaporan *Key Performance Index* (KPI) dari divisi *MIS & Data Analytics* agar dalam kondisi yang baik.

3.2 Pelaksanaan Kerja

Analisis data yang dilakukan adalah melakukan Analisa data pembiayaan yang terindikasi terdapat penyimpangan sesuai dengan Peraturan Operasional. Pada saat praktikan melakukan kerja profesi berkesempatan menganalisa data pembiayaan *First Payment Default* atau Nasabah sudah gagal bayar pada angsuran pertama. Untuk melakukan analisis data tersebut, praktikan terlebih dahulu harus membuat informasi pergerakan posisi kolektibilitas dari bulan lalu terhadap bulan berjalan menggunakan *python* dan *SQLite*.



Gambar 3.1 Logo Python
Sumber : <https://www.python.org/>



Gambar 3.2 Logo SQLite
Sumber : <https://www.sqlite.org/index.html>

Saat ini tempat praktikan melakukan kerja profesi, infrastruktur database yang digunakan oleh perusahaan sebagai *core datawarehouse* adalah DB2 dari perusahaan ternama IBM dan SQL Server keluaran perusahaan besar yaitu Microsoft. Dengan kondisi saat ini yaitu masa

pandemi yang dimana praktikan dijadwalkan untuk siap bekerja dengan sistem WFH (*work from home*) dan WFO (*work from office*) dan terbatasnya akses user ke database yang disediakan perusahaan, untuk itu *Team Leader Data Analytics* yaitu Bapak Tonny Harry Prasetyo mempunyai saran untuk menggunakan alternatif lain untuk mengolah dan melakukan analisa data agar tetap produktif yaitu dengan menggunakan Bahasa *python* sebagai Bahasa yang digunakan untuk melakukan analisa data dan menggunakan *SQLite* sebagai databasenya.

Mengapa menggunakan Python?

Salah satu alasan menggunakan *python* sebagai bahasa untuk melakukan analisa data yaitu dikarenakan *Python* merupakan bahasa pemrograman yang berorientasi objek (*object oriented*), *general purpose* dapat digunakan untuk berbagai aktivitas kode lainnya. Dengan bantuan *Python*, *engineer* dapat menggunakan lebih sedikit barisan kode untuk menyelesaikan tugas. *Python* sangatlah cepat, dan banyak *library* yang membuat *Python* lebih disukai juga, seperti *Matplotlib*. Dan, banyak *library* yang digunakan untuk *scientific computing* juga.

Python merupakan bahasa programming yang fleksibel, menyebabkannya, dicintai oleh *data science*. Juga, orang yang ingin terjun ke dunia data science lebih memilih *Python* ketimbang bahasa-bahasa programming lainnya. Jadi, jika *programmer* ingin mencoba hal yang menarik dan unik, mereka bisa melakukannya dengan *Python*. *Programmer* bahkan dapat menulis *script* untuk aplikasi dan website, dengan cara kreatif yang mereka mau. *Python* juga merupakan bahasa yang paling gampang untuk dikuasai. Bahasanya cukup sederhana, dan sangat gampang dibaca juga. (HACKTIV8, 2019)

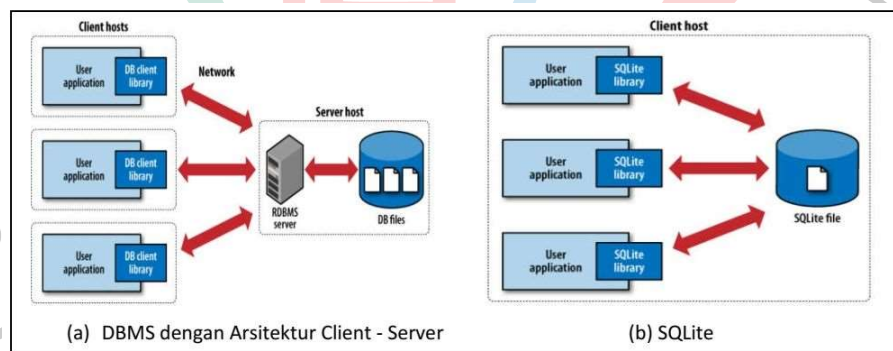
Mengapa menggunakan SQLite?

Penggunaan *SQLite* sebagai database dalam melakukan tugas analisa data bukan tanpa sebab, seperti disebutkan sebelumnya, pada masa pandemic praktikan diharuskan mengikuti komposisi kerja dengan menerapkan WFH dan WFO. Tentu jika praktikan sedang menerapkan

WFO akan mudah untuk melakukan akses dataset yang berada di *core data warehouse* perusahaan melalui jaringan intranet. Tetapi jika sedang menerapkan WFH, keterbatasan akses *core datawarehouse* perusahaan melalui jaringan internet akan sangat terbatas. Untuk itu praktikan menggunakan SQLite sebagai salah satu alternatif lain untuk tetap produktif selama WFH.

SQLite merupakan sebuah sistem manajemen basisdata relasional yang bersifat ACID-compliant dan memiliki ukuran pustaka kode yang relatif kecil, ditulis dalam bahasa C. SQLite merupakan proyek yang bersifat public domain yang dikerjakan oleh D. Richard Hipp. (Wikipedia, n.d.)

SQLite juga dapat dikatakan database yang sangat praktis, tidak memerlukan instalasi apapun pada perangkat komputer atau laptop yang digunakan. Dan dapat sangat mudah dari segi mobilitas nya.



Gambar 3.3 Perbedaan DBMS berarsitektur Client Server dengan SQLite

Sumber :

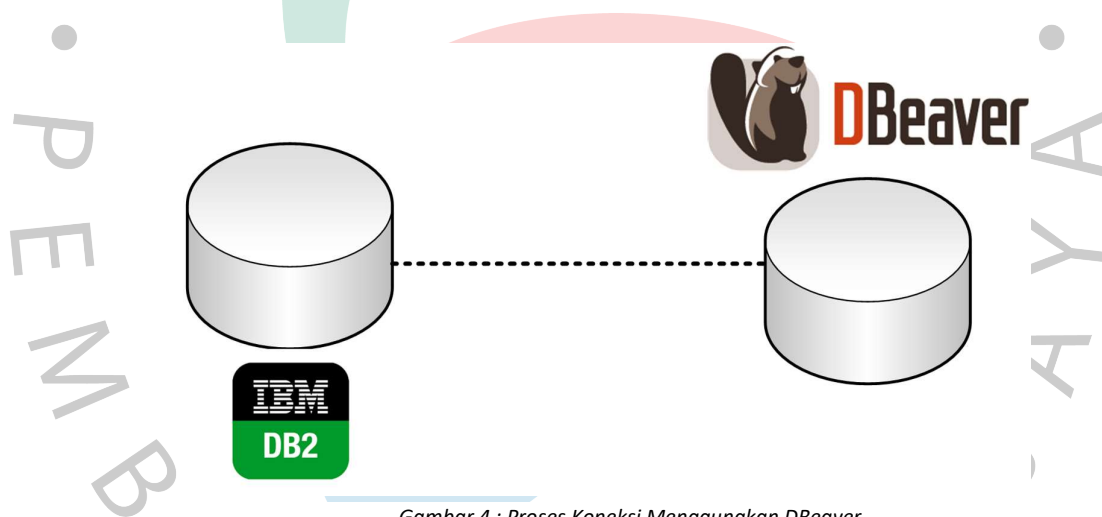
<https://www.kompasiana.com/edison1814/5f42712cd541df22f8627902/sqlite-perangkat-database-engine-yang-kecil-tapi-berguna>

Tahapan yang dilakukan sebagai berikut :

Salah satu tugas membuat Analisa data terkait pergerakan posisi kolektibilitas dari bulan lalu terhadap bulan berjalan atau bulan pelaporan.

Beberapa tahapan yang dilakukan adalah :

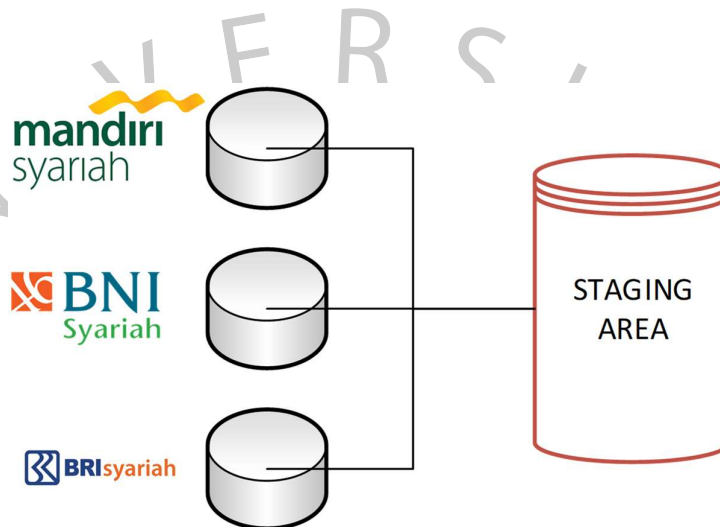
- a. Pengecekan ketersediaan data pada server DWH (*Data Warehouse*) untuk posisi akhir bulan.
- b. Jika data sudah tersedia pada *core datawarehouse* maka dilakukan pengumpulan data yang diperlukan dengan melakukan *joining* beberapa *schema* untuk memenuhi keperluan data yang dibutuhkan. Pada tahapan ini menggunakan *tools software DBeaver* untuk mengakses DB2 IBM database.



Gambar 4 : Proses Koneksi Menggunakan DBeaver

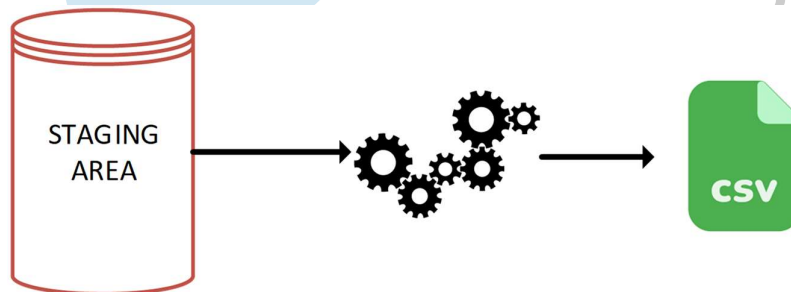
- c. Melakukan pengecekan validasi *sampling* hasil query pengumpulan data awal terhadap informasi di web *dashboard portal*.
- d. Kondisi saat ini posisi data masih ada di beberapa *core datawarehouse*. Untuk data legacy dengan melakukan pengumpulan data seperti :
 - 1) Legacy Bank Rakyat Indonesia Syariah (BRIS) berasal dari Saldolist BRIS.

- 2) Legacy Bank Negara Indonesia Syariah (BNIS) berasal dari Saldolist BNIS.
- 3) Legacy Data Bank Syariah Mandiri (BSM) berasal dari saldolist BSM



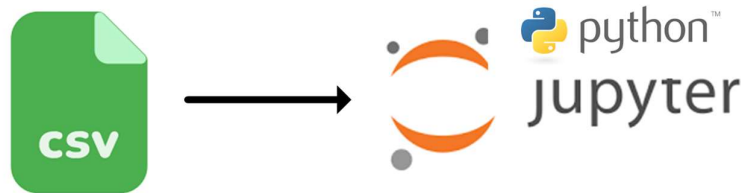
Gambar 5 : Proses Staging Data Dari Legacy

- e. Proses selanjutnya adalah setelah data sudah sesuai dengan *dashboard* , melakukan ekspor data hasil query dalam bentuk csv file yang nantinya akan diolah menggunakan *Python* dan *SQLite*.



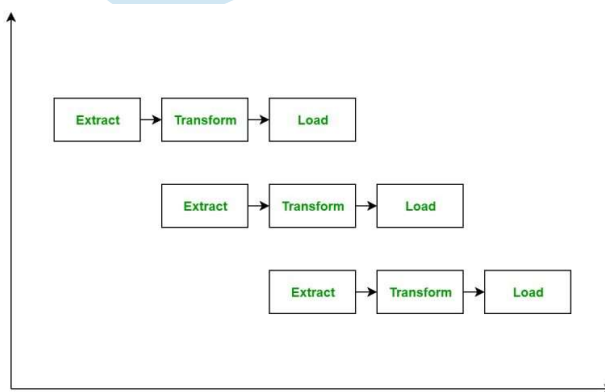
Gambar 6 : Proses ekspor data dalam bentuk csv file

- f. Proses berikutnya adalah proses ETL (*extract, transform, load*) yaitu proses integrasi data. Import csv file yang sudah terbentuk dari data staging untuk di olah dengan *python* menggunakan IDE Jupyter Notebook.



Gambar 7 : Proses import csv file menggunakan Jupyter Notebook

Pada tahapan ini untuk mempermudah proses agar dibuat satu folder khusus sesuai dengan data yang akan diolah nantinya. Penggunaan csv file dikarenakan kita dapat memindahkan suatu data dari satu sumber ke sumber lainnya dengan cukup mudah dan tidak diperlukannya proses manual input. Csv file adalah data berbasis teks yang berisikan data-data yang pada umumnya dipisahkan antara data kolom satu dengan yang lain dengan tanda baca koma (,) atau bisa dengan tanda baca lainnya yang bersifat unik. Hal ini juga dapat disebut sebagai proses ETL (*Extract, Transform, Load*)



Gambar 8 : Proses ETL Pipeline
Sumber : Geeksforgeeks.org

Proses import data menggunakan python cukup membantu dalam hal membaca apakah isi dari data tersebut merupakan kumpulan teks yang bertipe *string*, atau berupa tanggal, bahkan merupakan tipe data *interger* atau *float*, secara otomatis akan terbaca sesuai dengan tipe datanya. CSV atau comma separated value adalah salah satu tipe file yang digunakan secara luas di dunia programming. Tidak hanya itu CSV pun sering digunakan dalam pengolahan informasi yang dihasilkan spreadsheet untuk diproses lebih lanjut melalui mesin analitik. CSV pun dianggap sebagai file yang agnostik karena dapat digunakan oleh berbagai database untuk proses backup data.

g. Melakukan proses *cleansing* data menggunakan *Jupyter Notebook* seperti :

1) Penyesuaian tipe data.

Dikarenakan sumber data yang digunakan adalah berekstensi *.csv* yang pada dasarnya file ini adalah file berbasis teks, sehingga jika ingin dilakukan pengolahan data dibutuhkan proses perubahan tipe data. Pada umumnya *library* pada *python* akan secara otomatis membaca tipe data apa yang sesuai dengan data yang ada, apakah berbentuk *string*, *datetime*, *interger*, ataupun *float*.

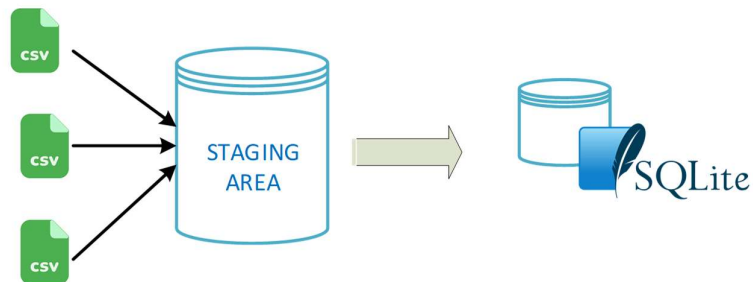
2) Proses seleksi kolom data yang sesuai dengan kebutuhan.

Hal ini cukup membantu mempersingkat proses pengolahan data nantinya dikarenakan hanya kolom-kolom yang sesuai dengan kebutuhan yang akan diimport dan dijadikan *dataframe* pada *Jupyter Notebook* nantinya.

3) Menghilangkan *whitespace* pada semua *value* dan *header*.

Proses ini diperlukan untuk memitigasi jika pada saat proses ekspor data dari *core datawarehouse* ada beberapa terdapat kelebihan spasi pada bagian depan atau belakang dari *value* ataupun dari *header name*. Python adalah Bahasa yang sangat

sensitive terhadap penamaan, penulisan baik itu huruf besar kecil maupun character dan spasi termasuk juga.



Gambar 9 : Proses ETL (Extract, Transform, Load)

h. Menggunakan *library* yang dimiliki *python* untuk memudahkan proses pekerjaan seperti :

1) Numpy

Numpy menjadi salah satu *library* yang paling banyak digunakan dalam data Science. Numpy yang merupakan singkatan dari Numerical Python menjadi alat analisis dan juga alat dalam pembuatan model. *Library* ini merupakan bagian dari SciPy yaitu ekosistem berbasis Python yang lebih besar dari tools open source. Selain digunakan untuk menyelesaikan persamaan linier dan perhitungan matematis lainnya, Numpy juga banyak digunakan untuk menjadi wadah multi-dimensi yang serbaguna bagi berbagai jenis data *generic*.

Fitur :

- a) Menyediakan fungsi yang cepat untuk melakukan proses data numeric
- b) Lebih efisien dikarenakan berorientasi array
- c) Proses yang singkat dan cepat

2) Pandas

Library Python lain yang sering digunakan dalam Data Analytics adalah Pandas. Numpy dan Pandas menjadi *library* yang lebih sering digunakan secara bersamaan. Sehingga tidak heran jika Pandas juga merupakan bagian dari SciPy serta tersedia di

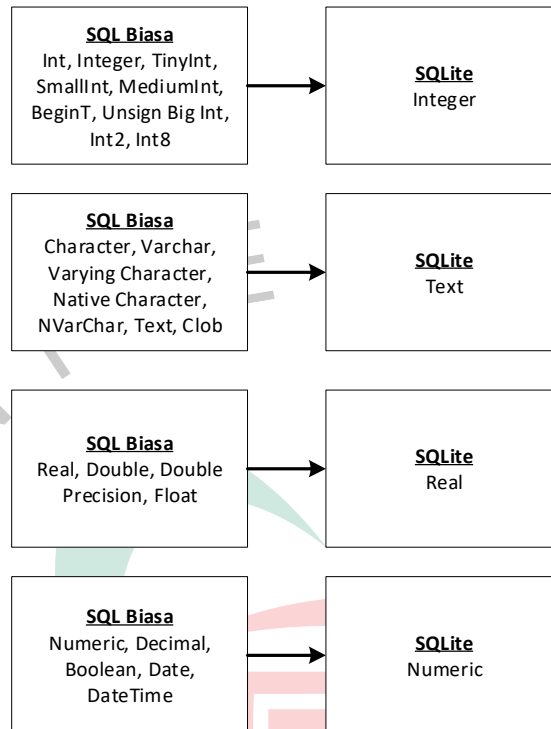
bawah lisensi *software open source* BSD. Pandas menjadi sangat ahli dalam mengatasi data yang tidak lengkap, tidak teratur, dan tidak terstruktur.

Library ini juga dilengkapi dengan tools yang digunakan untuk membentuk, menggabungkan, menganalisis, serta memvisualisasikan dataset. Pada dasarnya ada tiga jenis struktur data di library Pandas ini, yaitu Series (satu dimensi dan merupakan array homogen), DataFrame (dua dimensi dengan kolom yang bersifat heterogen), serta Panel (tiga dimensi, array size mutable). Untuk menggunakan library ini, kita perlu mengimport nya terlebih dahulu. Biasanya library ini disingkat dengan pd.

3) Sqlite3

Sebuah *library python* database engine yang tidak terikat dengan server yang bersifat standalone artinya berdiri sendiri tanpa ada pengaturan seperti database kebanyakan tapi memiliki fungsi yang sama dengan database pada umumnya, perintah dasar seperti sql biasa masih dapat dilakukan di sqlite.

Penggunaan SQLite pada pekerjaan yang sedang dikerjakan oleh praktikan sangat membantu dan cukup praktis dan mudah dalam pengoperasiannya. SQLite tidak memerlukan server khusus. Dapat dipindah ke dalam folder mana saja sesuai dengan kebutuhan praktikan.



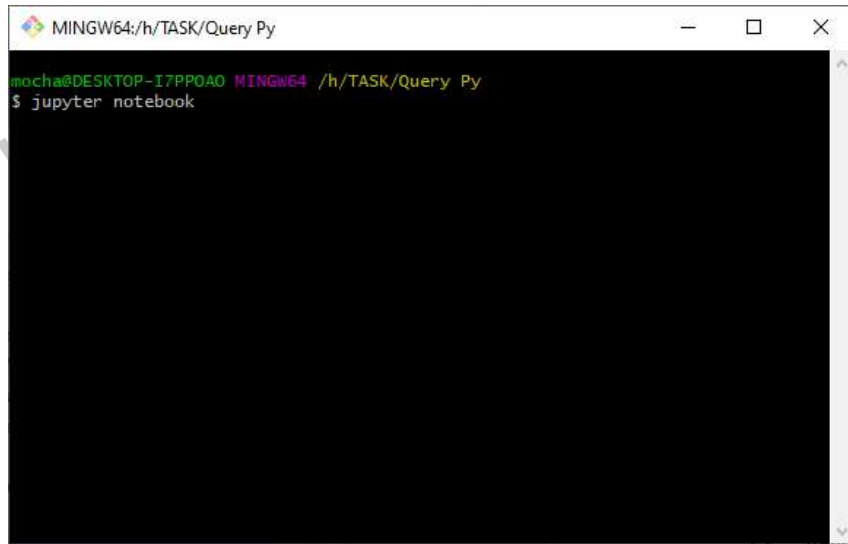
Gambar 10: Tipe data sqlite ada empat yang mewakili tipe database pada umumnya

4) Matplotlib

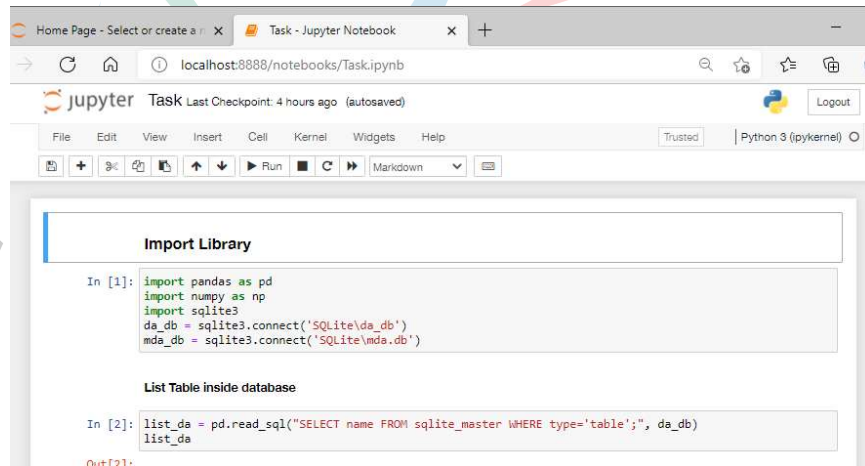
Library Python ini juga merupakan bagian dari paket inti *SciPy* dan berada di bawah lisensi BSD. Dengan *library* ini, kita dapat membuat *chart*, grafik, histogram, dll dengan sangat mudah dan tanpa memerlukan banyak kode. Hal ini karena library Matplotlib memang didesain untuk menghasilkan visualisasi yang sederhana dan kuat.

- i. Melihat hasil dari hasil syntax python yang sudah dibuat pada *Jupyter Notebook*. Penggunaan *Jupyter Notebook* cukup atraktif, kita bisa melihat hasil dari syntax yang kita buat. Selain hasil, kita juga dapat membuat narasi-narasi untuk menjelaskan lebih detail baik informasi terkait syntax, maupun hasil dari syntax tersebut dalam bentuk paragraph ataupun narasi lainnya.

Untuk menggunakan *Jupyter Notebook* kita cukup menggunakan *Command Prompt* ataupun menggunakan *Git Bash*. Berikut adalah contoh memanggil *Jupyter Notebook* menggunakan *Git Bash* :

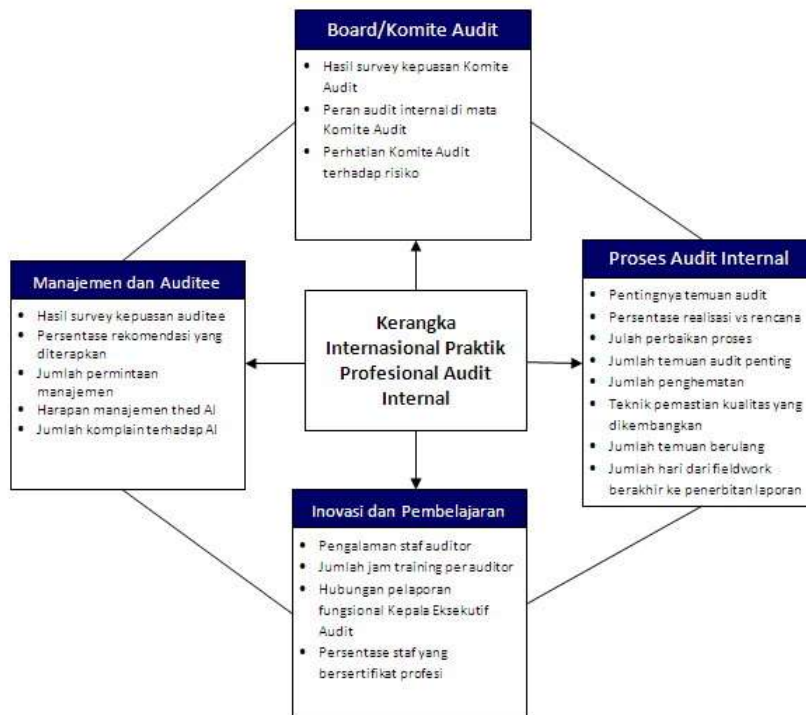


Gambar 11: Membuka Jupyter Notebook melalui Git Bash



Gambar 13: Contoh tampilan Jupyter Notebook

- j. Hasil dari semua syntax jika sudah sesuai makan dapat langsung di *export* ke dalam bentuk csv file atau Excel file untuk dilakukan pengolahan selanjutnya dijadikan laporan.
- k. Sebelum laporan di publish, memastikan terlebih dahulu melakukan proses validasi hasil dilakukan oleh *Team Leader*. Jika hasil validasi sudah sesuai makan dapat dilakukan pembuatan laporan akhir.
- l. Setelah laporan akhir dipublish, selanjutnya auditor data analitik terus memantau perkembangan dari laporan yang kita berikan kepada *auditee* guna untuk mendapatkan *feedback* dan dilakukan tindak lanjut hasil audit.



Gambar 14 : Alur Kerja Audit Internal

Hasil *Key Risk Indicator* dengan tema *First Payment Default* atau Nasabah pembiayaan segmentasi konsumsi dengan kondisi pada rentan waktu 0 sampai 6 bulan pertama dari pencairan pembiayaan terdapat penunggakan pembayaran angsuran. Hasil dari analisa atas penyebab terjadinya penunggakan pembayaran angsuran bisa disebabkan oleh banyak faktor seperti :

- a. Indikasi dana atau saldo pada rekening afiliasi dimana tempat rekening penampungan pembayaran angsuran pembiayaan kurang dari minimal nominal satu kali angsuran komposisi pokok dan margin, ataupun minimal senilai dengan nominal kewajiban pokoknya saja. Hal ini menyebabkan tidak terbacanya minimal proses oleh system yang mengakibatkan gagal debet oleh sistem.
- b. Indikasi berkurangnya sumber penghasilan utama nasabah yang mengakibatkannya tidak dapat terpenuhi kewajiban dalam membayar angsuran. Poin ini harus dibuktikan dan dikaji lebih mendalam oleh rekan *auditee* yang berada di cabang terkait.
- c. Indikasi *fraud* yang dilakukan oleh oknum yang tidak bertanggung jawab. Pola yang umum dilakukan oleh oknum tersebut adalah membuat profil nasabah palsu atau penggunaan profil nasabah lama yang sudah tidak ada lagi keawajiban di bank selanjutnya digunakan sebagai dokumen pengajuan pembiayaan.
- d. Indikasi *fraud* atau kelalaian dalam proses verifikasi dokumen, proses analisis pembiayaan, proses persetujuan pengajuan. Keterbatasan tim verifikasi lapangan membuat celah kejahatan yang dapat dilakukan oleh oknum petugas bank terkait.
- e. Kurangnya pengawasan yang dilakukan oleh tim *collection* dalam hal monitoring dan pengendalian pergerakan portfolio dari *bucket to bucket* kolektibilitas yang mengakibatkan turunnya kualitas pembiayaan nasabah pembiayaan baru.

Kriteria pengambilan data adalah sebagai berikut :

- a. Pengambilan periode data adalah posisi data kualitas pembiayaan akhir bulan pelaporan dengan histori kualitas pembiayaan nasabah 6 bulan dari data bulan pelaporan.
- b. Kriteria nasabah adalah :
 - 1) Nasabah aktif pembiayaan (*on balance*)
 - 2) Nasabah bukan merupakan penerima program Restrukturisasi baik program bank maupun dari program pemerintah
 - 3) Nasabah dengan umur pencairan 0 (nol) sampai dengan 6 (enam) bulan pada saat akhir bulan pelaporan

Poin yang ditekankan pada hal ini berdampak pada kualitas portfolio pembiayaan sangat besar dikarenakan akan membawa pengaruh yang cukup signifikan terhadap Cadangan Kerugian Penurunan Nilai (CKPN), dimana bank harus mengeluarkan atau mencadangkan dana atas penurunan kualitas pembiayaan yang terjadi akibat hasil temuan-temuan dari proses analisa data.

3.3 Kendala Yang Dihadapi

Kendala yang dihadapi praktikan adalah belum tersedia data mart pada unit Audit Internal. Hal ini dikarenakan perusahaan tempat praktikan bekerja adalah perusahaan hasil penggabungan 3 bank syariah milik pemerintah. Kondisi saat praktikan melakukan kerja praktik, untuk sumber data belum sepenuhnya terintegrasi menjadi satu database, melainkan masih berada di server perusahaan *legacy*. Hanya terdapat 1 database sementara atau tampungan yang digunakan untuk proses production bukan sebagai data mart atau non production.

Saat ini praktikan untuk mengakses data langsung mengakses ke datawarehouse production yang mengakibatkan terbatasnya waktu untuk pengaksesan data. Proses pengambilan data harus menunggu proses ETL harian selesai dilakukan oleh sistem sehingga proses pemrosesan data dilakukan pada sore hari.

Saat ini tempat praktikan melakukan kerja profesi, infrastruktur database yang digunakan oleh perusahaan sebagai *core datawarehouse* adalah DB2 dari perusahaan ternama IBM dan SQL Server keluaran perusahaan besar yaitu Microsoft. Dengan kondisi saat ini yaitu masa pandemi yang dimana praktikan dijadwalkan untuk siap bekerja dengan sistem WFH (*work from home*) dan WFO (*work from office*) dan terbatasnya akses user ke database yang disediakan perusahaan.

3.4 Cara Mengatasi Kendala

Penggunaan SQLite sebagai database dalam melakukan tugas analisa data bukan tanpa sebab, seperti disebutkan sebelumnya, pada masa pandemic praktikan diharuskan mengikuti komposisi kerja dengan menerapkan WFH dan WFO. Tentu jika praktikan sedang menerapkan WFO akan mudah untuk melakukan akses dataset yang berada di *core data warehouse* perusahaan melalui jaringan intranet. Tetapi jika sedang

menerapkan WFH, keterbatasan akses *core datawarehouse* perusahaan melalui jaringan internet akan sangat terbatas. Untuk itu praktikan menggunakan SQLite sebagai salah satu alternatif lain untuk tetap produktif selama WFH.

