

AKHIR

LAPORAN



**jAKARTA, aGUSTUS 2021**

**tIM DEVELOPER**

**CV. STRATEGI SELARAS PERFORMA**

**PENINGKATAN MUTU KUALITAS KARYA REKAM**

**(cleansing data)**

**TAHUN ANGGARAN 2021**

# **LAPORAN AKHIR**

**PENINGKATAN MUTU KUALITAS KARYA REKAM**

**(cleansing data)**

**TAHUN ANGGARAN 2021**

Jakarta, Agustus 2021

Disusun Oleh:

Tim Developer

CV. Strategi Selaras Performa



# **KATA PENGANTAR**

Segala puji bagi Tuhan YME atas limpahan rahmat dan hidayah-Nya sehingga penyusunan “Laporan Akhir Peningkatan Mutu Kualitas Karya Rekam (Cleansing Data)” dapat terselesaikan dengan baik. Semoga laporan akhir ini bermanfaat bagi kita semua.

Jakarta, Agustus 2021

Penyusun

# **DAFTAR ISI**

# **DAFTAR TABEL**

# **DAFTAR GAMBAR**

# **BAB I**

**IKHTISAR PEKERJAAN**

## **PENDAHULUAN**

**Cleansing data**atau yang disebut juga dengan data scrubbing merupakan suatu proses analisa mengenai kualitas dari data dengan mengubah. Bisa juga pengelola mengoreksi ataupun menghapus data tersebut. Data yang dibersihkan tersebut adalah data yang salah, rusak, tidak akurat, tidak lengkap dan salah format. Data yang tepat sangat berpengaruh terhadap pengambilan keputusan. Cleansing data ini untuk mengoreksi data secara sistematis lewat algoritma, sehingga bisa lebih menghemat biaya dan waktu. Pada prosesnya, *cleansing data* ini dilakukan tidak hanya dengan cara otomatis namun juga dengan proses manual, sehingga hasilnya akan lebih optimal.

Perpustakaan Nasional telah menghimpun KCKR menggunakan Aplikasi Penghimpun Konten Web , Aplikasi e-Deposit, dan INLIS Enterprise modul Deposit. Pada 3 aplikasi, saat ini memiliki data sejumlah 1.512.550 judul/item yang berasal dari Aplikasi Penghimpun Konten Web sejumlah 1.181.672 item, e-Deposit sejumlah 46.478 item, dan INLIS Enterprise modul Deposit sejumlah 284.400 judul. Dari data sejumlah 1.512.550 ini belum dipilah sehingga data berupa masih data kotor. Data kotor berasal dari kesalahan dalam penginputan, data yang rudak, data yang duplikat, dan kurangnya standarisasi data. Kondisi ini memerlukan proses *cleansing* *data* terhadap data dari 3 aplikasi tersebut, agar data dapat terkelola dengan baik.

## **TUJUAN**

Tujuan dari kegiatan ini adalah:

* + - 1. Memilah data sampah dan data duplikasi dari INLIS, e-deposit dan Penghimpun Konten Web

1. Menghasilkan data wajib serah, dan data mengenai Serah Simpan Karya Cetak Karya Rekam yang valid dari INLIS, e-Deposit dan Penghimpun Konten Web
2. Meningkatkan akurasi dan validitas data terhadap pelaksanaan UU SSKCKR

## **SASARAN**

Sasaran kegiatan Peningkatan Mutu Kualitas Karya Rekam (cleansing data) adalah sebagai berikut:

1. Dilakukannya penentuan kriteria kualitas data
2. Dilakukannya proses *cleansing data* INLIS modul deposit, eDeposit dan Penghimpun Konten Web dalam 4 tahapan yaitu: audit cleansing data; penentuan spesifikasi alur kerja; eksekusi alur kerja cleansing data; dan pengendalian dan pasca eksekusi *cleansing data*
3. Didapatkannya data INLIS Deposit, e-Deposit, dan Penghimpun Konten Web yang bersih dan akurat

## **OUTCOMES**

Outcomes Peningkatan Mutu Kualitas Karya Rekam, adalah :

1. Meningkatnya kepercayaan wajib serah terhadap kualitas data dan informasi karya rekam nasional
2. Meningkatnya kepatuhan wajib serah terhadap UU SSKCKR
3. Meningkatnya penggunaan Sistem Serah Simpan Karya Cetak dan Karya Rekam sebagai acuan yang valid bagi wajib simpan Provinsi

## **LOKASI KEGIATAN**

Lokasi kegiatan Peningkatan Mutu Kualitas Karya Rekam (cleansing data) di Direktorat Deposit dan Pengembangan Koleksi Perpustakaan, Perpustakaan Nasional Republik Indonesia (PERPUSNAS RI) Jl. Salemba Raya No.28 A, Jakarta Pusat.

## **WAKTU PELAKSANAAN**

Waktu pelaksanaan kegiatan Peningkatan Mutu Kualitas Karya Rekam (cleansing data) dimulai dari tanggal 28 Juli 2021 sampai dengan 27 Agustus 2021.

## **RUANG LINGKUP PELAKSANAAN KEGIATAN**

* + - 1. Melakukan rapat awal untuk menyamakan persepsi antara pengguna aplikasi dengan tim penyedia jasa
      2. Melakukan *asessmen* dan tujuan data yang dicleansing
      3. Menentukan tools untuk cleansing data
      4. Melakukan uji coba cleansing data pada data testing
      5. Mengadakan rapat peninjauan (*review*) *progress* pekerjaan
      6. Membuat dokumentasi langkah-langkah data cleansing yang dilakukan
      7. Membuat dokumen laporan pekerjaan
      8. Menentukan kriteria kualitas data
* Menentukan validitas data
* Melakukan *decleansing*
* Menentukan akurasi
* Menentukan kelengkapan data
* Menentukan konsistensi data
* Menentukan keseragaman

## **SUMBER PENDANAAN**

Pelaksanaan kegiatan Peningkatan Mutu Kualitas Karya Rekam (cleansing data) dibiayai Anggaran Pendapatan dan Belanja Negara (APBN) Perpustakaan Nasional RI Tahun Anggaran 2021.

# **BAB II**

**PROGRESS PEKERJAAN**

## **HASIL AUDIT CLEANSING DATA**

Adapun hasil audit *cleansing data* sebagai berikut;

1. INLIS

1. a. Mengisi ID ISBN ke dalam tabel wajib serah (penerbit) dan isian lainnya

Deskripsi:

Tabel wajib serah (penerbit) diisi dengan ID ISBN, sehingga ID ISBN terkait pada tabel penerbit

Tujuan:

ID ISBN terkait pada tabel penerbit, alamat kosong pada tabel penerbit sesuai alamat pada db ISBN, dan data penting lainnya terisi

1. b. Mengelompokkan penerbit berdasarkan jenis pemerintah dan swasta

Deskripsi:

Penerbit dikelompokkan berdasarkan jenis pemerintah dan swasta

Tujuan:

User dapat memperoleh informasi jenis penerbit berdasarkan jenisnya; pemerintah atau swasta

* + - * 1. a. Membuat list penerbit yang akan dihapus dan dipertahankan

Deskripsi:

Daftar penerbit yang akan dihapus dan dipertahankan dibuat dalam tabel excel

Tujuan:

Dengan dibuatnya tabel yang berisi penerbit yang akan dihapus dan dipertahankan terlihat jenis masalah yang dialami sehingga menjadi pertimbangan untuk dihapus atau dipertahankan data penerbit tersebut

* + - * 1. b. Menghapus Penerbit.

Deskripsi:

Data koleksi penerbit yang tidak diperlukan dihapus

Tujuan:

Dengan dihapusnya data koleksi yang tidak diperlukan, maka data koleksi penerbit menjadi lebih valid dari data-data yang tidak diperlukan

1. Collections yang tidak memiliki nomor deposit

Deskripsi:

Data koleksi deposit yang tidak memiliki nomor deposit didata dan dlaporkan

Tujuan:

Dengan adanya data dan laporan koleksi deposit yang tidak memiliki nomor deposit, dapat dijadikan pertimbangan untuk melakukan tindakan selanjutnya

1. Collections yang mempunyai ISBN elektronik, namun ada di dalam penerimaan KC INLIS deposit

Deskripsi:

Data koleksi deposit yang mempunyai ISBN elektronik, namun ada di dalam penerimaan KC INLIS deposit didata dan dlaporkan

Tujuan:

Dengan adanya data dan laporan mempunyai ISBN elektronik, namun ada di dalam penerimaan KC INLIS deposit, dapat dijadikan pertimbangan untuk melakukan tindakan selanjutnya

1. Melengkapi detail collections seperti tag MARC catalog

Deskripsi:

Detail koleksi deposit dilengkapi sesuai dengan tag MARC catalog

Tujuan:

Format metadata koleksi deposit sesuai dengan tag MARC catalog

1. Mengelompokkan penerbit ke dalam group penerbit

Deskripsi:

Penerbit dikelompokkan ke dalam group penerbit

Tujuan:

Pengelompokkan penerbit ke dalam group penerbit memudahkan pendataan dan pengelolaan data penerbit

1. Pengelompokan penerbit berdasarkan provinsi

Deskripsi:

Penerbit dikelompokkan berdasarkan data provinsi alamat penerbit

Tujuan:

Pengelompokkan penerbit berdasarkan provinsi memudahkan pendataan dan pengelolaan data penerbit

1. E-DEPOSIT
   * + 1. Memindahkan kontributor ke dalam tabel collection\_contributors

Deskripsi:

Data kontributor dipindahkan ke dalam Tabel collection\_contributors

Tujuan:

Data kontributor tersimpan dalam Tabel collection\_contributors

* + - 1. Melengkapi collections

Deskripsi:

Melengkapi data collections

Tujuan:

Data collections yang tidak lengkap harus dilengkapi, sehingga data koleksi yang lengkap memudahkan pendataan dan pengelolaan data koleksi deposit

* + - 1. Mengelompokkan publisher pemerintah, swasta, dan perorangan

Deskripsi:

Publisher dikelompokkan berdasarkan jenis pemerintah, swasta, dan perorangan

Tujuan:

Pengelompokkan publisher berdasarkan jenis pemerintah, swasta, dan perorangan memudahkan pendataan dan pengelolaan data publisher

* + - 1. Membersihkan author yang sama

Deskripsi:

Author yang sama dibersihkan dari database

Tujuan:

Database author bersih dari data author yang sama sehingga tidak ada duplikasi author

* + - 1. Menghapus duplikasi code dengan sistem code ISBN yang sama pada collection

Deskripsi:

Collection yang memiliki kode ISBN yang sama maka dihapus dengan menyisakan kode ISBN paling terbaru

Tujuan:

Koleksi bersih dari duplikasi code ISBN

* + - 1. Menghapus duplikasi nama penerbit

Deskripsi:

Duplikasi nama penerbit dihapus

Tujuan:

Database nama penerbit bersih dari duplikasi nama penerbit

* + - 1. Menggabungkan koleksi dari penerbit dengan /DEP ke penerbit yang benar

Deskripsi:

Menggabungkan koleksi dari penerbit dengan /DEP ke penerbit yang benar

Tujuan:

Diperoleh data koleksi yang benar dari penerbit

* + - 1. Pengelompokan penerbit beserta koleksi berdasarkan Provinsi

Deskripsi:

Database penerbit beserta koleksi dikelompokkan berdasarkan Provinsi

Tujuan:

Pengelompokkan penerbit beserta koleksi berdasarkan Provinsi memudahkan pendataan dan pengelolaan data penerbit

* + - 1. Pengelompokkan penerbit kedalam grup penerbit

Deskripsi:

Penerbit dikelompokkan ke dalam group penerbit

Deskripsi:

Tujuan:

Pengelompokkan penerbit ke dalam group penerbit memudahkan pendataan dan pengelolaan data penerbit

* + - 1. Sinkronisasi koleksi yang ada di pengelolaan e-Deposit dengan data ISBN (received\_date)

Deskripsi:

Koleksi yang ada di pengelolaan e-Deposit dilakukan proses sinkronisasi dengan data ISBN (received\_date)

Tujuan:

Data koleksi yang ada di pengelolaan e-Deposit dilakukan sinkronisasi dengan data ISBN (received\_date)

1. PENGHIMPUN KONTEN WEB
   1. Menghapus salah satu archive jurnal yang sama namun berbeda link. Archive yang dihapus adalah archive dengan link yang lama

Deskripsi:

Salah satu archive jurnal yang sama namun berbeda link dihapus dengan kriteria archive yang dihapus adalah archive dengan link yang lama

Tujuan:

Database archive jurnal bersih dari archive yang sama namun berbeda link

* 1. Cleansing files yang rusak

Deskripsi:

Dilakukan pembersihan files konten yang rusak (0 byte)

Tujuan:

Files konten yang rusak tidak lagi tersimpan didalam aplikasi

* 1. Pengelompokan publisher, archive, dan artikel berdasarkan provinsi

Deskripsi:

Publisher, archive, dan artikel dikelompokkan berdasarkan provinsi

Tujuan:

Pengelompokkan publisher, archive, dan artikel berdasarkan provinsi memudahkan pendataan dan pengelolaan data publisher, archive, dan artikel

## **LANGKAH-LANGKAH CLEANSING DATA INLIS**

1.a. Mengisi ID ISBN ke dalam tabel wajib serah (penerbit) dan isian lainnya

* Langkah-langkah:
* Kaitkan ID ISBN pada tabel penerbit
* Isikan alamat yang masih kosong pada tabel penerbit sesuai alamat pada db ISBN
* Isikan data lainnya yang penting
* Mengisi PUBLISHER\_ISBN\_ID di MASTER\_PUBLISHER database INLIS dengan kd\_penerbit di mst\_penerbit database ISBN
* Mengisi kolom ADDRES1, ADDRESS1, ADDRESS2, ADDRESS3, POS\_CODE yang NULL di MASTER\_PUBLISHER database INLIS dengan alamat\_penerbit, alamat\_penerbit2, alamat\_penerbit3, alamat\_penerbit4 di mst\_penerbit database ISBN
* Mengisi PROVINCE di MASTER\_PUBLISHER database INLIS dengan concatenate (REGION-NAMAPROPINSI) dan di vlookup dari tabel PROPINSI database INLIS
* Mengisi PUBLISHER\_GROUP\_NAME di MASTER\_PUBLISHER database INLIS dengan di vlookup dari MASTER\_PUBLISHER\_GROUP database INLIS
* Screenshot;



* Tools yang digunakan:

Bahasa yang digunakan adalah python dengan tools anaconda untuk sub tools jupyter notebook

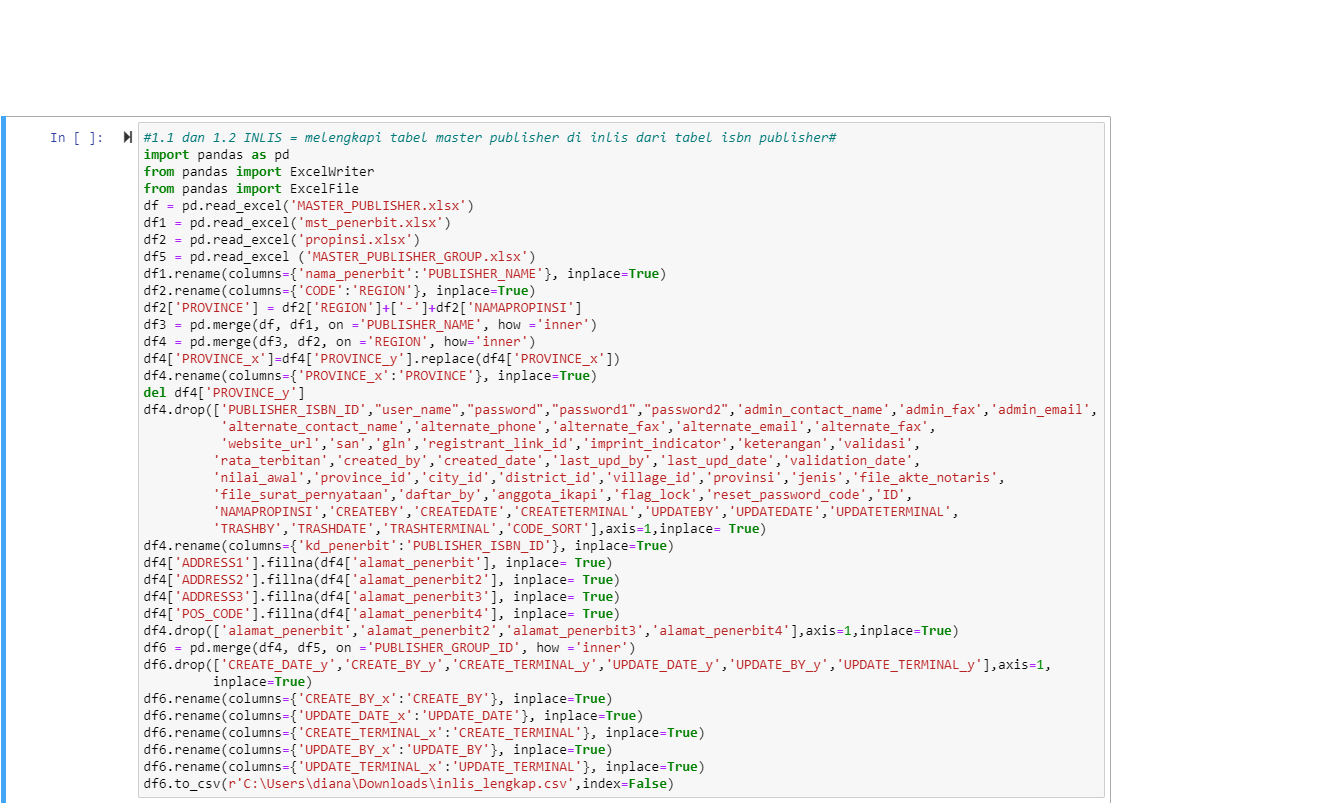
* Algoritma / Flowchart:
* Coding
* Perencanaan;



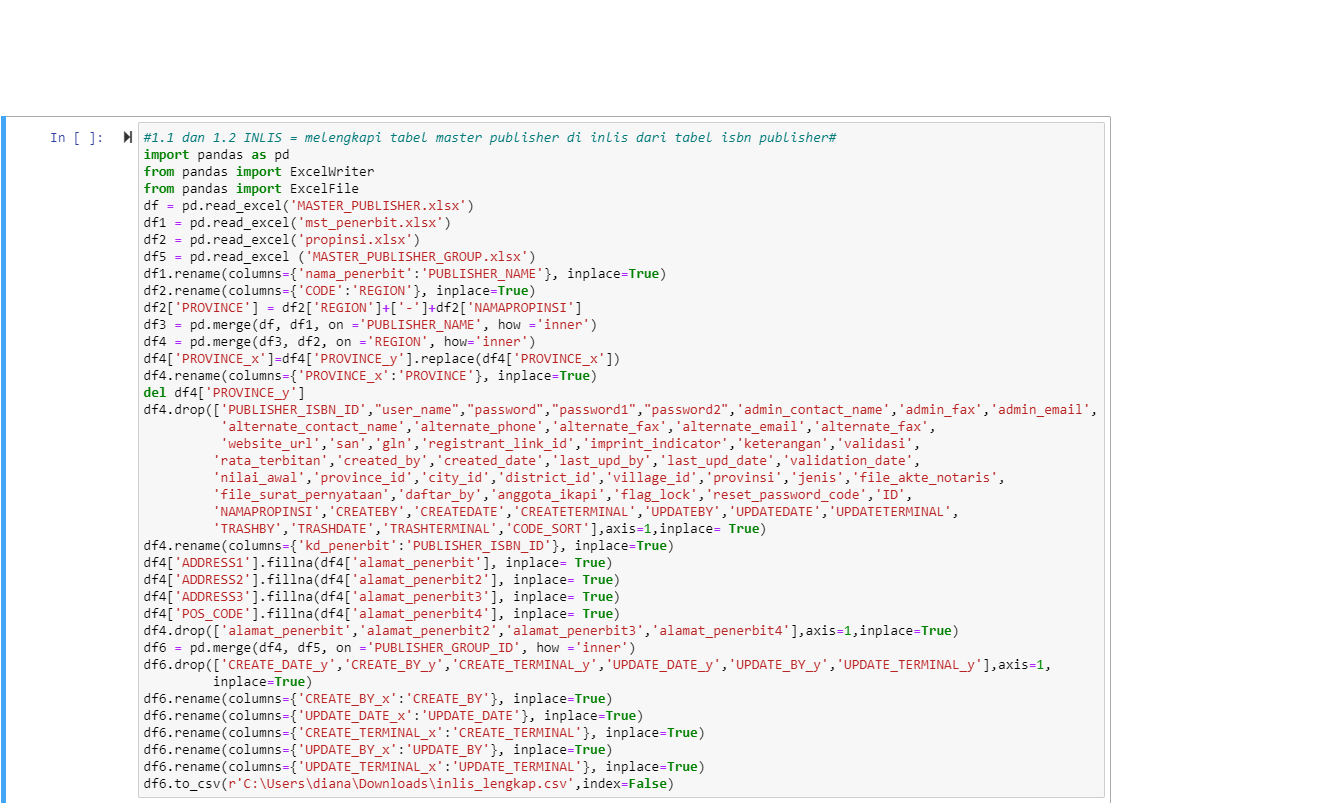
* Proses awal;



* Proses;



* Proses akhir;



* Proses
* Proses awal;
* Install Pandas = pip install pandas
* Import pandas alias pd
* Menggunakan excelwriter dan excelfile untuk olah data excel
* Inisiasi file apa saja yang ingin di ditambahkan ke MASTER\_PUBLISHER
* Proses;
* Rename kolom nama penerbit di mst\_penerbit dengan kolom PUBLISHER\_NAME agar sama di MASTER\_PUBLISHER.
* Rename kolom code di PROPINSI dengan kolom REGION agar sama di MASTER\_PUBLISHER.
* Concatenate kolom REGION dan kolom NAMAPROPINSI sehingga menghasilkan kolom baru bernama PROVINCE di tabel PROPINSI.
* Masukkan semua informasi yang ada di mst\_penerbit ISBN ke dalam MASTER\_PUBLISHER INLIS dengan patokan PUBLISHER\_NAME sebut file df4.
* Masukkan semua informasi yang ada di PROPINSI ke dalam df4.
* Membersihkan kolom kolom yang tidak digunakan dengan drop.
* Rename kd\_penerbit di df4 menjadi PUBLISHER\_ISBN\_ID.
* Mengisi kolom yang NULL di ADDRESS1, ADDRESS2, ADDRESS3,POS\_CODE dengan alamat\_penerbit, alamat\_penerbit2, alamat\_penerbit3, alamat\_penerbit4.
* Hapus kolom yang tidak digunakan.
* Memasukkan semua informasi yang ada di table MASTER\_PUBLISHER\_GROUP ke dalam MASTER\_PUBLISHER yang baru (df4).
* Membersihkan data yang tidak perlu.
* Proses akhir;
* Export tabel yang baru df6 ke csv atau ke xls dengan index=false supaya index nya tidak ikut terexport
* Hasil pekerjaan

Terlampir

1.b. Mengelompokkan penerbit berdasarkan jenis pemerintah dan swasta

* Langkah-langkah:
* Membuat rule yang menandakan pemerintah atau swasta. Buat dalam skrip sql atau pemrograman lainnya. Lampirkan skrip di sini.
* Lampirkan hasil sebelum di lakukan cleansing dan setelah cleansing.
* Laporkan pada user.
* Tools yang digunakan:
* Algoritma / Flowchart:
* Coding
* Screenshot proses
* Hasil pekerjaan

2.a. Membuat list penerbit yang akan dihapus dan dipertahankan

* Langkah-langkah:
* Buat daftar penerbit yang akan dihapus dan dipertahankan berdasarkan nama atau alamat yang mirip,
* Buat dalam excel
* Lampirkan hasil pekerjaan pada task ini
* Tools yang digunakan:
* Algoritma / Flowchart:
* Coding
* Screenshot proses
* Hasil pekerjaan

2. b. Menghapus Penerbit.

* Langkah-langkah:
* Ganti publisher\_id pada tabel collections menjadi publisher\_id yang baru
* Hapus penerbit yang tidak diperlukan
* Tuliskan script dan screenshoot hasil pekerjaan

* Tools yang digunakan:
* Algoritma / Flowchart:
* Coding
* Screenshot proses
* Hasil pekerjaan
  + 1. Collections yang tidak memiliki nomor deposit
* Langkah-langkah:
* Lakukan query ke database untuk collections yang tidak memiliki nomor deposit namun memiliki category\_id = 4
* Simpan dalam excel
* Lampirkan hasilnya di sini
* Tools yang digunakan:
* Algoritma / Flowchart:
* Coding
* Screenshot proses
* Hasil pekerjaan
  + 1. Collections yang mempunyai ISBN elektronik, namun ada di dalam

penerimaan KC INLIS deposit

* Langkah-langkah:
* Pada database ISBN di tabel mst\_isbn tidak ada kolom yang menandakan jenis ISBN elektronik maupun ISBN cetak cetak , sehingga perlu dibuat view pada sqlserver local
* Buat script untuk membandingkan ISBN pada collections dan pada View yang telah dibuat. Boleh menggunakan apa pun. Lampirkan script di sini
* Simpan daftar collections dalam excel, lampirkan di sini
* Report ke user, untuk kelanjutan data tersebut
* Tools yang digunakan:
* Algoritma / Flowchart:
* Coding

View pada sqlserver local, berikut skrip nya :

SELECT

a.kd\_penerbit,

a.kd\_penerbit\_dtl,

a.title,

a.kepeng creator,

a.tahun\_terbit year,

a.edisi edition,

a.seri,

CASE WHEN a.title like '%sumber elektronis%'

THEN 'elek'

ELSE 'cetak'

END jenis,

b.keterangan,

a.peruntukan,

a.tahun\_terbit,

(a.kd\_penerbit + a.kd\_penerbit\_dtl) bar\_val,

ISNULL(a.kepeng, '') kepeng,

a.edisi,

a.jml\_hlm,

a.ketebalan,

a.call\_number,

a.catatan,

a.media,

a.jejakan1,

a.jejakan2,

a.jejakan3,

a.jejakan4,

a.jejakan5,

a.subjek,

a.subjek2,

a.subjek3,

a.subjek4,

a.subjek5,

b.created\_date,

CASE

WHEN b.received\_date IS NULL

THEN '0'

ELSE CONVERT(VARCHAR(10), b.received\_date,104)

END received\_date,

b.prefix\_element + '-' + b.publisher\_element + '-' + b.item\_element + '-' +

b.check\_digit AS isbnno

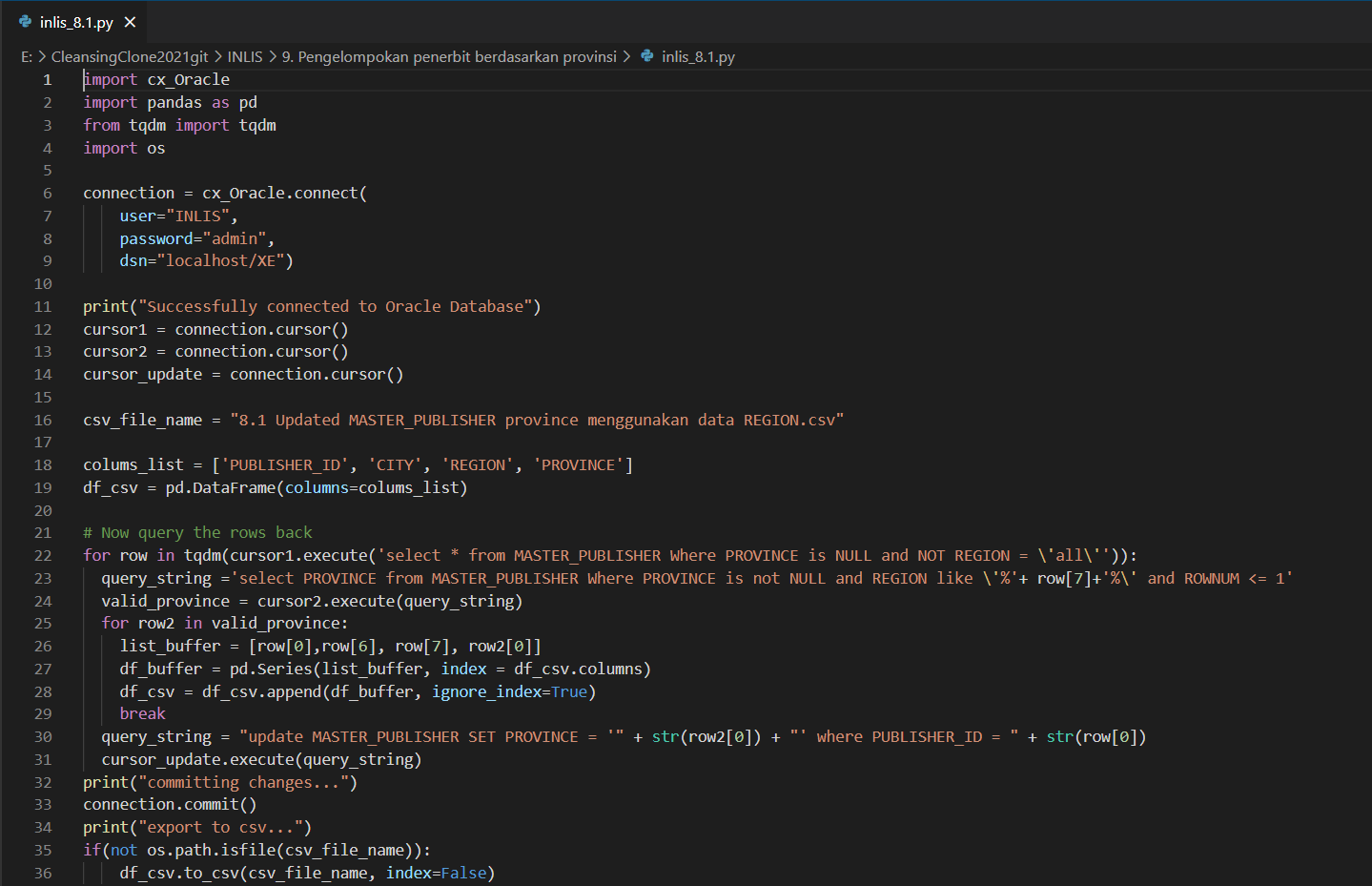
FROM

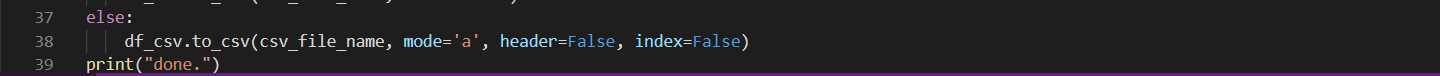
mst\_isbn b

LEFT JOIN mst\_penerbit\_dtl a ON (a.kd\_penerbit\_dtl = b.kd\_penerbit\_dtl)

* Screenshot proses
* Hasil pekerjaan

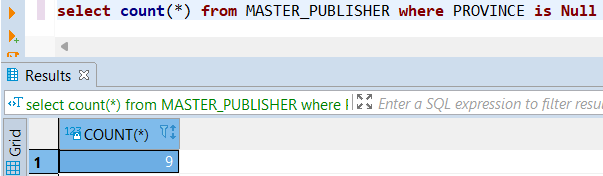
* + 1. Melengkapi detail collections seperti tag MARC catalog
* Langkah-langkah:
* Tools yang digunakan:
* Algoritma / Flowchart:
* Coding
* Screenshot proses
* Hasil pekerjaan
  + 1. Mengelompokkan penerbit ke dalam group penerbit
* Langkah-langkah:
* Perguruan tinggi, kementerian/lembaga bisa dibuat grup berdasarkan nama
* contoh : BPS, Universitas Indonesia
* Tools yang digunakan:
* Algoritma / Flowchart:
* Coding
* Screenshot proses
* Hasil pekerjaan
  + 1. Pengelompokan penerbit berdasarkan Provinsi
* Langkah-langkah:
* Tools yang digunakan:
* Bahasa Pemrograman Python
* Python Virtual Environment
* Python Libraries:
  + cx\_Oracle
  + Pandas
* Algoritma / Flowchart:
* Coding





* Screenshot proses

Sesudah;

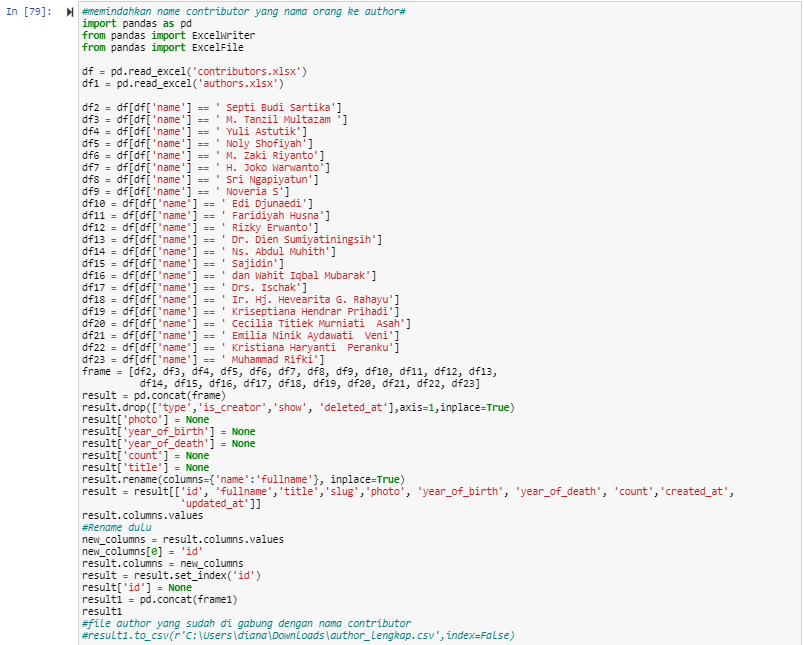


* Hasil pekerjaan

Terlampir

## **LANGKAH-LANGKAH CLEANSING DATA EDEPOSIT**

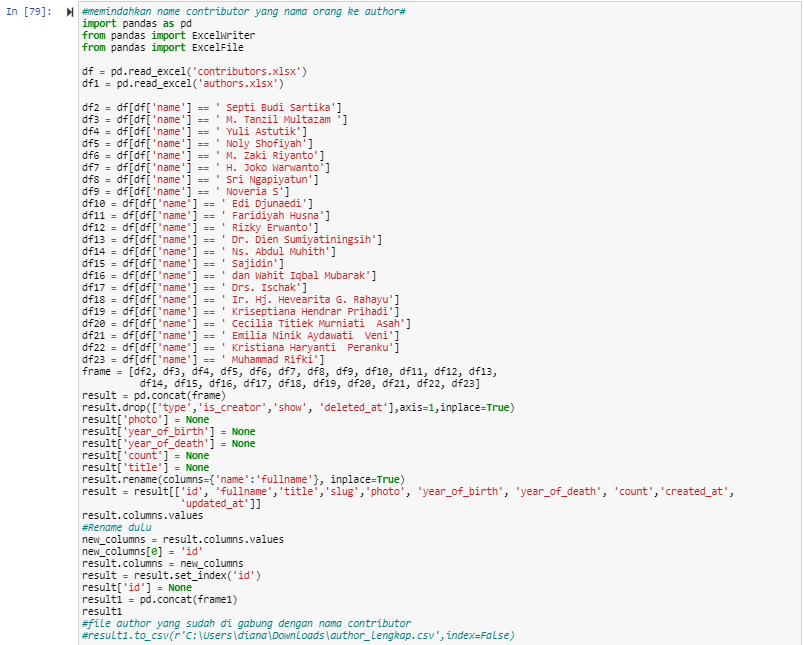
* + - 1. Memindahkan kontributor ke dalam tabel collection\_contributors
* Langkah-langkah:
* Filter data pada tabel contributors yang merupakan nama orang
* Pindahkan data tersebut pada tabel collection\_contributors
* Jika author belum ada, tambahkan pada tabel authors
* Lampirkan data sebelum dan sesudah
* Lampirkan langkah-langkah, query dan skrip yang digunakan di sini
* Membaca file kolom name di table contributor edeposit yang nama orang
* Dibuat dalam satu dataframe yaitu ‘frame’ untuk diconcat dataframe baru yaitu ‘result’
* Hapus kolom yang tidak perlu di dataframe result
* Menambahkan head yang sesuai dengan dataframe author
* Mengatur urutan kolom agar sesuai dataframe author
* Mengubah kolom index menjadi id di dataframe result
* Gabungkan dataframe author dengan dataframe result
* Lalu export
* Screenshot;



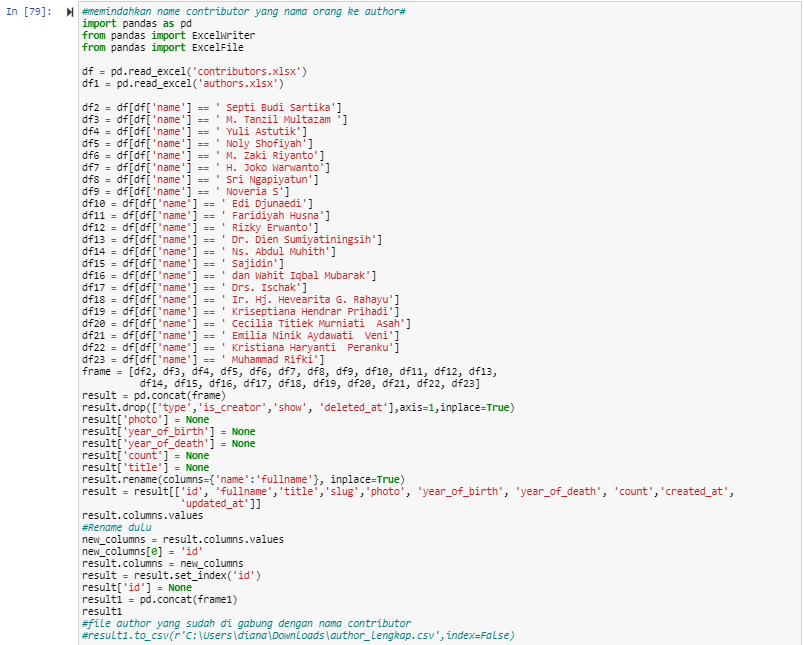
* Tools yang digunakan:

Bahasa yang digunakan adalah python dengan tools anaconda untuk sub tools jupyter notebook

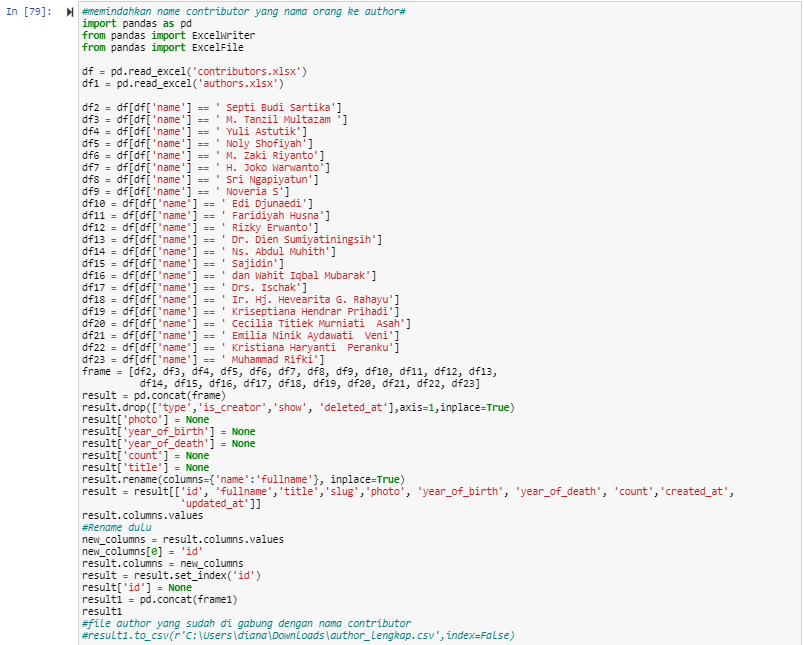
* Algoritma / Flowchart:
* Coding
* Perencanaan;



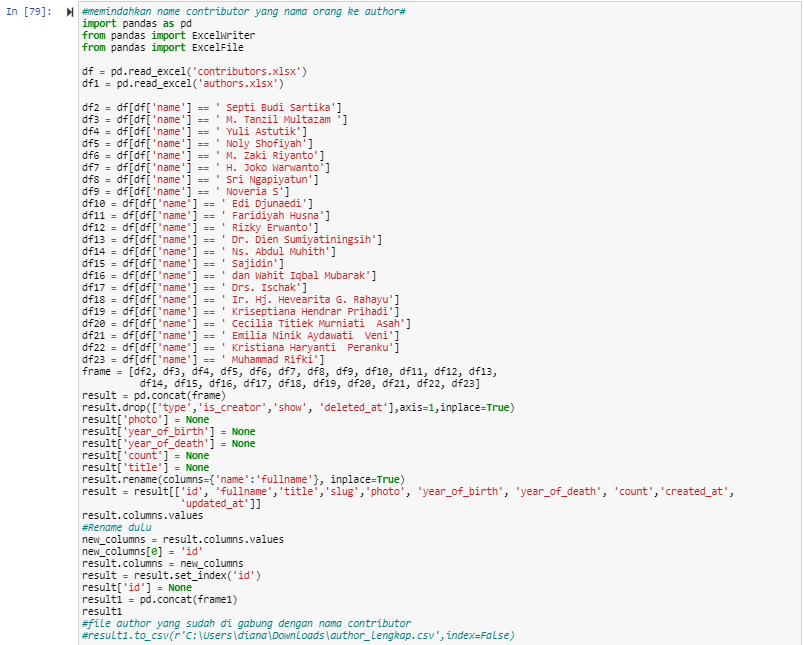
* Awal Proses eDeposit memindah author;



* Proses eDeposit memindah author



* Akhir Proses eDeposit memindah author



* Proses
* Awal proses
* Install pandas : pip install pandas
* Import pandas as pd
* Import excelwriter dan excelfile untuk eksekusi file xls
* Definisikan file yang ingin di olah dengan suatu nama
* Proses
* Membaca file kolom name di table contributor edeposit yang nama orang
* Dibuat dalam satu dataframe yaitu ‘frame’ untuk diconcat dataframe baru yaitu ‘result’
* Hapus kolom yang tidak perlu di dataframe result
* Menambahkan head yang sesuai dengan dataframe author
* Mengatur urutan kolom agar sesuai dataframe author
* Mengubah kolom index menjadi id di dataframe result
* Gabungkan dataframe author dengan dataframe result
* Lalu export
* Akhir proses
* Export file yang sudah di olah ke csv
* Hasil pekerjaan

Terlampir

* + - 1. Melengkapi collections
* Langkah-langkah:
* Lengkapi kolom access = 2
* Lengkapi preview dan perbaiki data preview
* Lengkapi city\_id dengan menggunakan data dari publishers
* Lampirkan skrip, dan bukti pekerjaan di sini
* Membaca file kolom name di publishers yang ada dengan kategori tertentu untuk di pisahkan berdasarkan 3 untuk pemerintah, 1 untuk swasta
* Jika kolom golongan null maka di isi 2 sebagai perorangan
* Lalu export
* Screenshot;



* Tools yang digunakan:

Bahasa yang digunakan adalah python dengan tools anaconda untuk sub tools jupyter notebook

* Algoritma / Flowchart:
* Coding
* Perencanaan;



* Awal;



* Proses;



* Akhir;



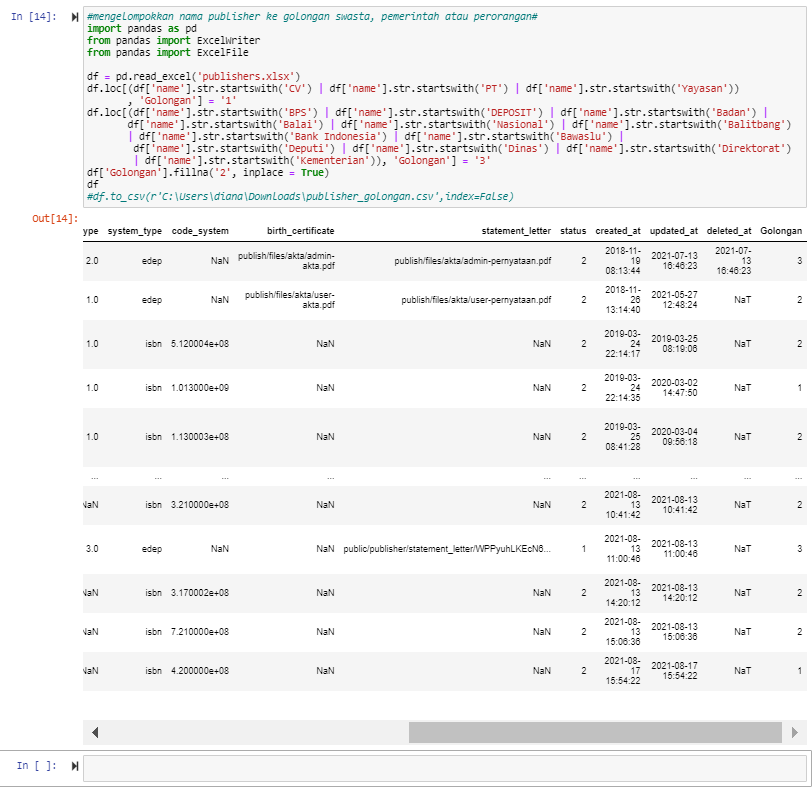
* Proses
* Awal proses
* Install pandas : pip install pandas
* Import pandas alias pd
* Import excelwriter dan excelfile untuk mengeksekusi xls
* Inisialisasi nama df untuk membaca excel
* Proses
* Membaca file xls publisher dengan mencari dikolom name untuk tulisan CV, PT, dll untuk di kategorikan sesuai pengelompokan
* Kolom golongan yang null di isi 2
* Akhir proses
* Membaca file xls publisher dengan mencari dikolom name untuk tulisan CV, PT, dll untuk di kategorikan sesuai pengelompokan
* Kolom golongan yang null di isi 2
* Hasil pekerjaan

Terlampir

* + - 1. Mengelompokkan publisher pemerintah, swasta, dan perorangan
* Langkah-langkah:
* Buat rule yang mengelompokan penerbit pemerintah, swasta, atau perorangan
* Lampirkan skrip / rule atau langkah-langkah yang diperlukan di sini
* Lampirkan hasil seblum dan sesudah
* Tools yang digunakan:

Bahasa yang digunakan adalah python dengan tools anaconda untuk sub tools jupyter notebook

* Algoritma / Flowchart:
* Coding



* Screenshot proses



* Hasil pekerjaan

Terlampir

* + - 1. Membersihkan author yang sama
* Langkah-langkah:
* Tools yang digunakan:
* Bahasa Pemrograman Python
* Python Virtual Environment
* Python Libraries:
  + Peewee
  + Pandas
  + Playhouse
* Algoritma / Flowchart:
* Coding
* Screenshot proses
* Hasil pekerjaan
  + - 1. Menghapus duplikasi code dengan sistem code ISBN yang sama pada collection
* Langkah-langkah:
* Collection status 1,2,3
* Dihapus file yang terbaru (status dipertahankan status 2)
* Tools yang digunakan:
* Bahasa Pemrograman Python
* Python Virtual Environment
* Python Libraries:
  + Peewee
  + Pandas
  + Playhouse
* Algoritma / Flowchart:
* Coding
* Screenshot proses
* Hasil pekerjaan

Terlampir

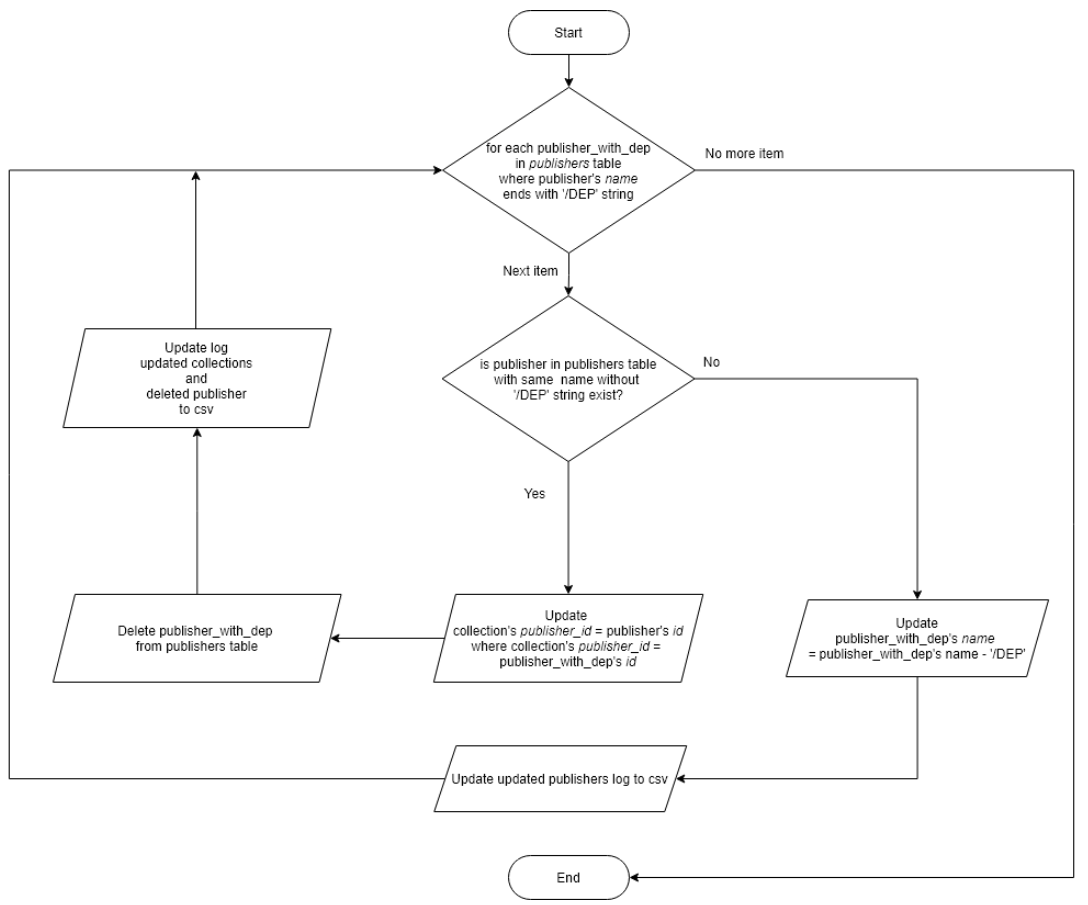
* + - 1. Menghapus duplikasi nama penerbit
* Langkah-langkah:
* Data update pada database hanya ada pada data 6.3.
* Csv 6.1 dan 6.2 hanya sebagai data support untuk task 9.
* Tools yang digunakan:
* Bahasa Pemrograman Python
* Python Virtual Environment
* Python Libraries:
  + Peewee
  + Pandas
  + Playhouse
* Algoritma / Flowchart:
* Coding
* Screenshot proses
* Hasil pekerjaan

Terlampir

* + - 1. Menggabungkan koleksi dari penerbit dengan /DEP ke penerbit yang benar

Tools:

* Langkah-langkah:
* Mengelompokkan setiap penerbit yang diakhiri dengan '/DEP' string
* Memilih penerbit di tabel penerbit dengan nama yang sama tanpa keterangan '/ DEP' string.
* Proses hapus publisher\_with\_dep dari tabel penerbit
* Proses perbaruan log update collections dan hapus penerbit ke csv
* Tools yang digunakan:
* Bahasa Pemrograman Python
* Python Virtual Environment
* Python Libraries:
  + Peewee
  + Pandas
  + Playhouse
* Algoritma / Flowchart:

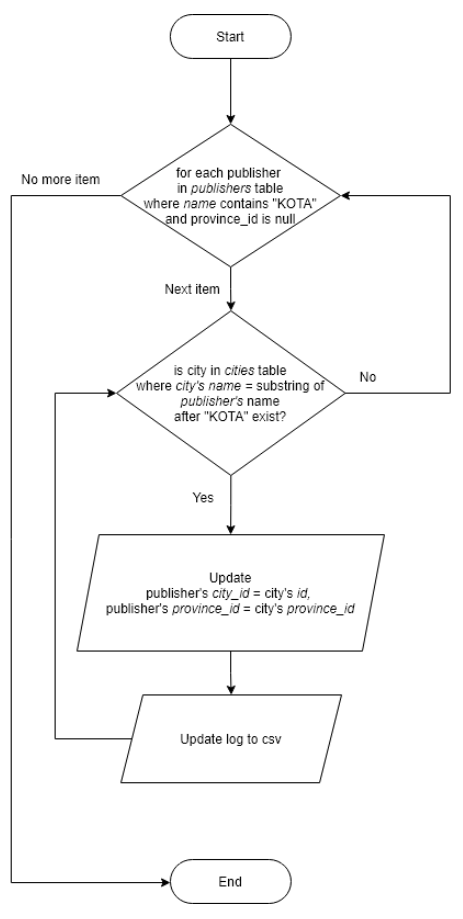


Penjelasan;

* + 1. Mulai
    2. Mengelompokkan setiap penerbit dengan dep di tabel penerbit dimana nama penerbit diakhiri dengan '/DEP' string
    3. Kemudian memilih penerbit di tabel penerbit dengan nama yang sama tanpa keterangan '/ DEP' string. Untuk penerbit dengan nama yang sama tanpa keterangan ‘/DEP’ string lanjut ke proses perbarui publisher\_id koleksi = id penerbit di mana publisher\_id koleksi = id penerbit\_with\_dep, jika tidak sama maka masuk proses perbarui nama publisher\_with\_dep = nama publisher\_with\_dep - /DEP'
    4. Untuk penerbit dengan nama yang sama tanpa keterangan ‘/DEP’ string lanjut ke proses hapus publisher\_with\_dep dari tabel penerbit, kemudian lanjut ke proses perbarui log update collections dan hapus penerbit ke csv. Untuk penerbit yang tidak sama lanjut ke proses perbarui updated publishers log ke csv.
    5. Hasil proses pembaruan log update collections berupa penerbit dengan dep di tabel penerbit di mana nama penerbit diakhiri dengan '/DEP' string yang telah diperbaharui
    6. Selesai
* Coding
* Screenshot proses
* Hasil pekerjaan

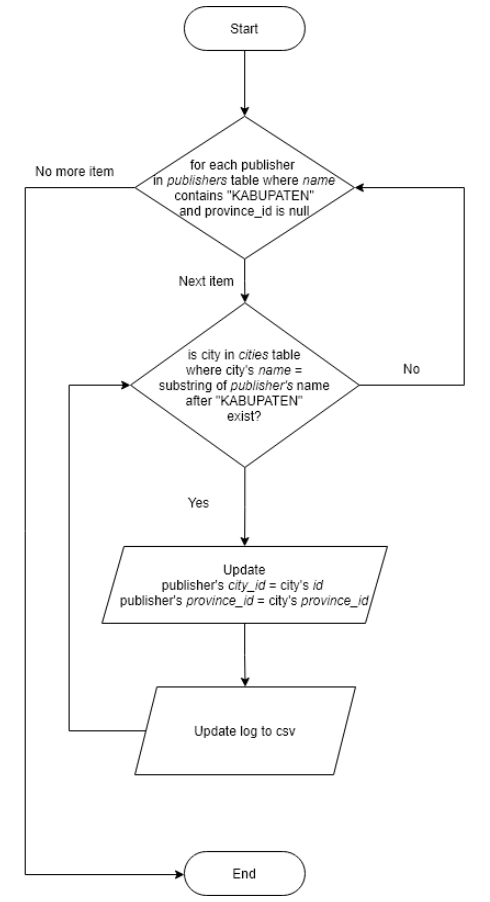
Terlampir

* + - 1. Pengelompokan penerbit beserta koleksi berdasarkan Provinsi
* Langkah-langkah:
* Updated publishers province\_id menggunakan keyword Kota
* Updated publishers province\_id menggunakan keyword Kabupaten
* Updated publishers province\_id menggunakan keyword Provinsi
* Updated publishers province\_id menggunakan keyword Akhiran Alamat
* Tools yang digunakan:
* Bahasa Pemrograman Python
* Python Virtual Environment
* Python Libraries:
  + Peewee
  + Pandas
  + Playhouse
* Algoritma / Flowchart:
  + 1. Updated publishers province\_id menggunakan keyword Kota;



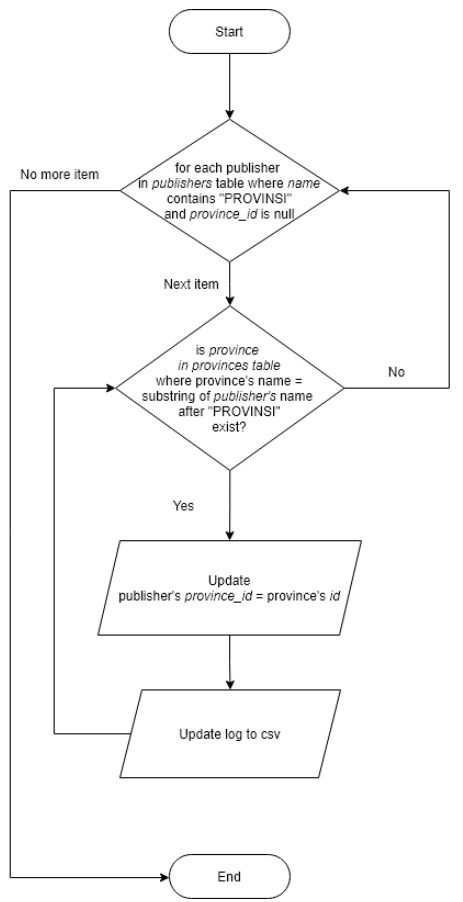
Penjelasan;

* + 1. Mulai
    2. Proses pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata “KOTA” dan province\_id is null
    3. Kemudian dilanjutkan proses pemilihan data tabel kota dimana nama kota = substring nama penerbit setelah “KOTA” ada
    4. Jika ditemukan tabel kota dimana nama kota = substring nama penerbit setelah “KOTA” ada, maka masuk dalam proses pembaharuan data untuk publisher’s city\_id = city’s id dan publisher’s province\_id = city’s province\_id. Jika tidak ditemukan tabel kota dimana nama kota = substring nama penerbit setelah “KOTA” ada, maka kembali ke proses sebelumnya yaitu pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata “KOTA” dan province\_id is null
    5. Setelah proses pembaharuan data untuk publisher’s city\_id = city’s id dan publisher’s province\_id = city’s province\_id dilakukan, maka proses selanjunya perbarui log menjadi format CSV. Setelah file diperbarui menjadi format CSV, maka kembali ke proses pemilihan data tabel kota dimana nama kota = substring nama penerbit setelah “KOTA” ada. Jika tidak ditemukan tabel kota dimana nama kota = substring nama penerbit setelah “KOTA” ada, maka kembali ke proses sebelumnya yaitu pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata “KOTA” dan province\_id is null
    6. Selesai
    7. Updated publishers province\_id menggunakan keyword Kabupaten;



Penjelasan;

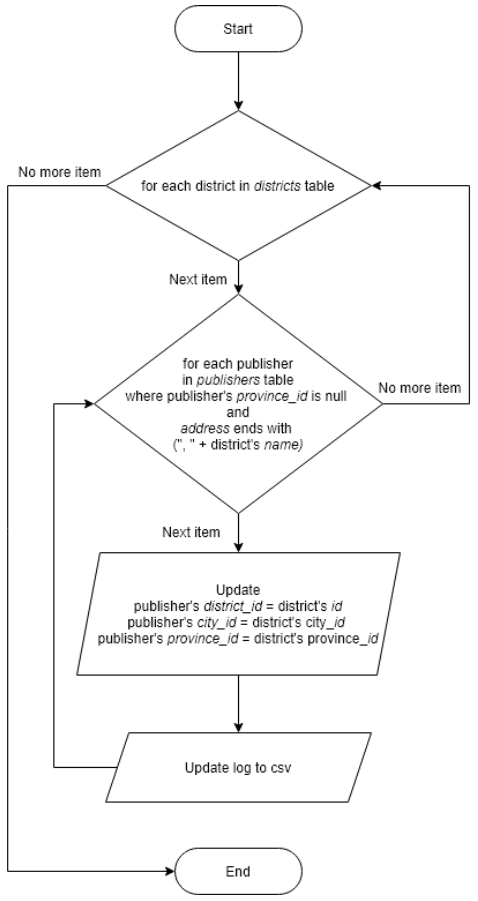
* + 1. Mulai
    2. Proses pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata “KABUPATEN” dan province\_id is null
    3. Kemudian dilanjutkan proses pemilihan data tabel kota dimana nama kota = substring nama penerbit setelah “KABUPATEN” ada
    4. Jika ditemukan tabel kota dimana nama kota = substring nama penerbit setelah “KABUPATEN” ada, maka masuk dalam proses pembaharuan data untuk publisher’s city\_id = city’s id dan publisher’s province\_id = city’s province\_id. Jika tidak ditemukan tabel kota dimana nama kota = substring nama penerbit setelah “KABUPATEN” ada, maka kembali ke proses sebelumnya yaitu pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata “KABUPATEN” dan province\_id is null
    5. Setelah proses pembaharuan data untuk publisher’s city\_id = city’s id dan publisher’s province\_id = city’s province\_id dilakukan, maka proses selanjunya perbarui log menjadi format CSV. Setelah file diperbarui menjadi format CSV, maka kembali ke proses pemilihan data tabel kota dimana nama kota = substring nama penerbit setelah “KABUPATEN” ada. Jika tidak ditemukan tabel kota dimana nama kota = substring nama penerbit setelah “KABUPATEN” ada, maka kembali ke proses sebelumnya yaitu pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata “KABUPATEN” dan province\_id is null
    6. Selesai
    7. Updated publishers province\_id menggunakan keyword Provinsi;



Penjelasan;

* + 1. Mulai
    2. Proses pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata “PROVINSI” dan province\_id is null
    3. Kemudian dilanjutkan proses pemilihan data tabel kota dimana nama kota = substring nama penerbit setelah “PROVINSI” ada
    4. Jika ditemukan tabel kota dimana nama kota = substring nama penerbit setelah “PROVINSI” ada, maka masuk dalam proses pembaharuan data untuk publisher’s provinces\_id = province’s\_id. Jika tidak ditemukan tabel kota dimana nama kota = substring nama penerbit setelah “PROVINSI” ada, maka kembali ke proses sebelumnya yaitu pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata “PROVINSI” dan province\_id is null
    5. Setelah proses pembaharuan data untuk publisher’s provinces\_id = province’s\_id dilakukan, maka proses selanjunya perbarui log menjadi format CSV. Setelah file diperbarui menjadi format CSV, maka kembali ke proses pemilihan data tabel kota dimana nama kota = substring nama penerbit setelah “PROVINSI” ada. Jika tidak ditemukan tabel kota dimana nama kota = substring nama penerbit setelah “PROVINSI” ada, maka kembali ke proses sebelumnya yaitu pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata “PROVINSI” dan province\_id is null
    6. Selesai
    7. Updated publishers province\_id menggunakan keyword Akhiran Alamat;

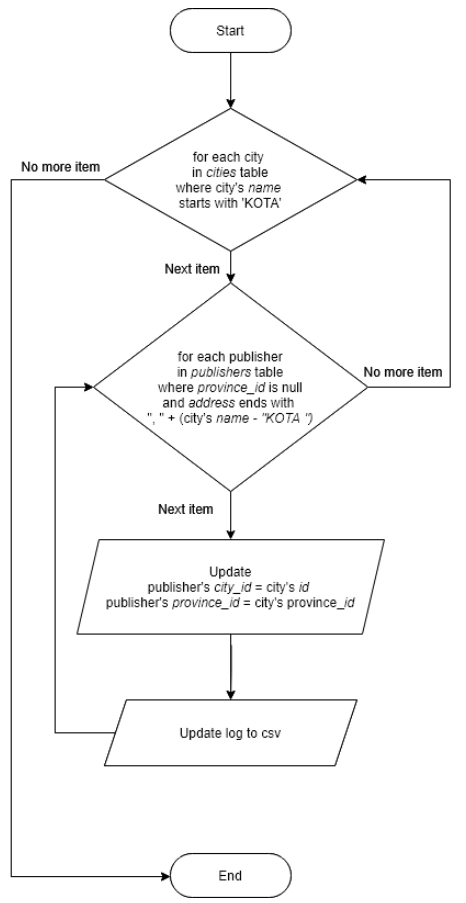
4.1. Berdasarkan Kelurahan dalam tabel Kelurahan



Penjelasan;

* + 1. Mulai
    2. Proses pemilihan data setiap Kelurahan dalam tabel Kelurahan
    3. Kemudian dilanjutkan proses pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan (“,”+ nama kelurahan)
    4. Jika penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan (“,”+ nama kelurahan), maka masuk dalam proses pembaharuan data untuk publisher’s district\_id = district’s id, publisher city\_id = district’s city\_id dan publisher’s province\_id = district’s province\_id. Jika tidak ditemukan penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan (“,”+ nama kelurahan), maka kembali ke proses sebelumnya yaitu Proses pemilihan data setiap Kelurahan dalam tabel Kelurahan
    5. Setelah proses pembaharuan data untuk publisher’s district\_id = district’s id, publisher city\_id = district’s city\_id dan publisher’s province\_id = district’s province\_id dilakukan, maka proses selanjunya perbarui log menjadi format CSV. Setelah file diperbarui menjadi format CSV, maka kembali ke proses pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan (“,”+ nama kelurahan). Jika tidak ditemukan penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan (“,”+ nama kelurahan), maka kembali ke proses sebelumnya yaitu Proses pemilihan data setiap Kelurahan dalam tabel Kelurahan
    6. Selesai

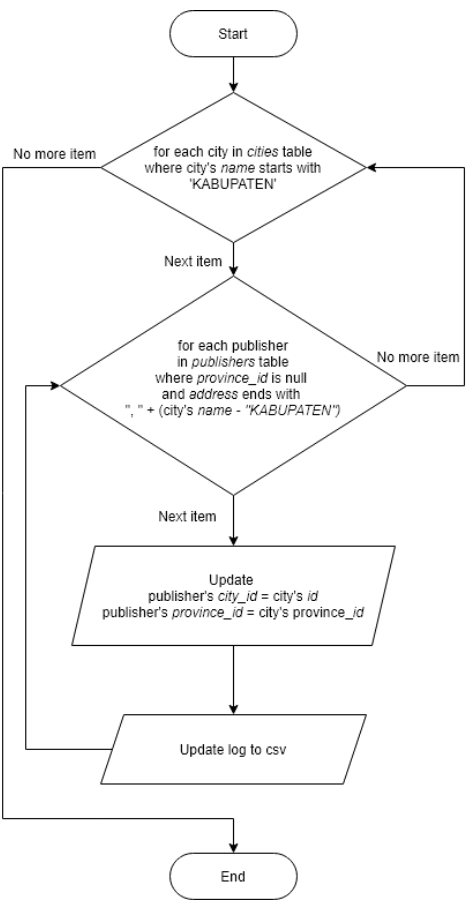
4.2. Berdasarkan Kota dalam tabel Kota dengan awalan ‘Kota’



Penjelasan;

* + 1. Mulai
    2. Proses pemilihan data setiap kota dalam tabel kota yang mengandung kata “KOTA”
    3. Kemudian dilanjutkan proses pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan (“,”+ nama kota – “KOTA”)
    4. Jika penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan (“,”+ nama kota – “KOTA”), maka masuk dalam proses pembaharuan data untuk publisher’s city\_id = city’s id, dan publisher’s province\_id = city’s province\_id. Jika tidak ditemukan penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan (“,”+ nama kota – “KOTA”), maka kembali ke proses sebelumnya yaitu Proses pemilihan data setiap kota dalam tabel kota yang mengandung kata “KOTA”
    5. Setelah proses pembaharuan data untuk publisher’s city\_id = city’s id, dan publisher’s province\_id = city’s province\_id dilakukan, maka proses selanjunya perbarui log menjadi format CSV. Setelah file diperbarui menjadi format CSV, maka kembali ke proses pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan (“,”+ nama kota – “KOTA”)). Jika tidak ditemukan penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan (“,”+ nama kota – “KOTA”)), maka kembali ke proses sebelumnya yaitu Proses pemilihan data setiap kota dalam tabel kota yang mengandung kata “KOTA”
    6. Selesai

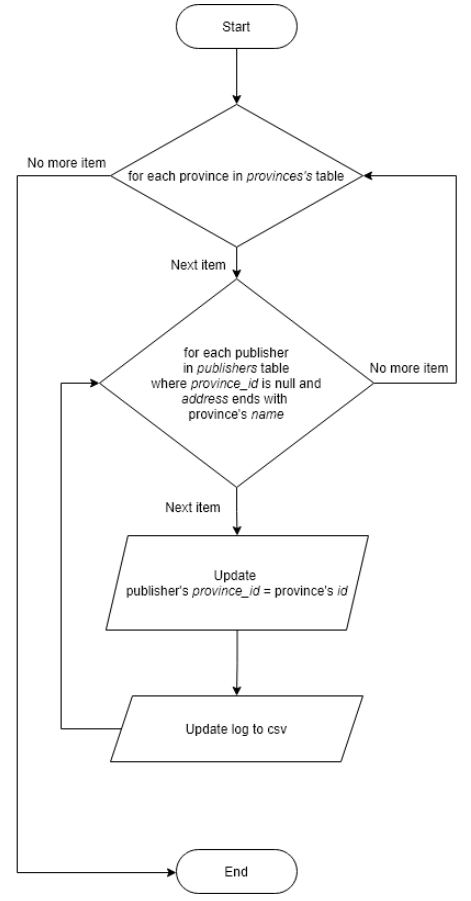
4.3. Berdasarkan Kota dalam tabel Kota dengan awalan ‘Kabupaten’



Penjelasan;

* + 1. Mulai
    2. Proses pemilihan data setiap kota dalam tabel kota yang mengandung kata “KABUPATEN”
    3. Kemudian dilanjutkan proses pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan (“,”+ nama kota – “KABUPATEN”)
    4. Jika penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan (“,”+ nama kota – “KABUPATEN”), maka masuk dalam proses pembaharuan data untuk publisher’s city\_id = city’s id, dan publisher’s province\_id = city’s province\_id. Jika tidak ditemukan penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan (“,”+ nama kota – “KABUPATEN”), maka kembali ke proses sebelumnya yaitu Proses pemilihan data setiap kota dalam tabel kota yang mengandung kata “KABUPATEN”
    5. Setelah proses pembaharuan data untuk publisher’s city\_id = city’s id, dan publisher’s province\_id = city’s province\_id dilakukan, maka proses selanjunya perbarui log menjadi format CSV. Setelah file diperbarui menjadi format CSV, maka kembali ke proses pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan (“,”+ nama kota – “KABUPATEN”)). Jika tidak ditemukan penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan (“,”+ nama kota – “KABUPATEN”)), maka kembali ke proses sebelumnya yaitu Proses pemilihan data setiap kota dalam tabel kota yang mengandung kata “KABUPATEN”
    6. Selesai

4.4. Berdasarkan provinsi dalam tabel provinsi



Penjelasan;

* + 1. Mulai
    2. Proses pemilihan data setiap provinsi dalam tabel provinsi yang mengandung kata provinsi
    3. Kemudian dilanjutkan proses pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan nama provinsi
    4. Jika penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan nama provinsi, maka masuk dalam proses pembaharuan data untuk publisher’s province\_id = province’s\_id. Jika tidak ditemukan penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan nama provinsi, maka kembali ke proses sebelumnya yaitu Proses pemilihan data setiap provinsi dalam tabel provinsi yang mengandung kata provinsi
    5. Setelah proses pembaharuan data untuk publisher’s province\_id = province’s\_id dilakukan, maka proses selanjunya perbarui log menjadi format CSV. Setelah file diperbarui menjadi format CSV, maka kembali ke proses pemilihan data setiap penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan nama provinsi. Jika tidak ditemukan penerbit dalam tabel penerbit yang mengandung kata province\_id is null dan alamat yang diakhiri dengan nama provinsi, maka kembali ke proses sebelumnya yaitu Proses pemilihan data setiap provinsi dalam tabel provinsi yang mengandung kata provinsi
    6. Selesai
* Coding
* Screenshot proses
* Hasil pekerjaan

Terlampir

* + - 1. Pengelompokkan penerbit kedalam grup penerbit
* Langkah-langkah:
* Tools yang digunakan:
* Algoritma / Flowchart:
* Coding
* Screenshot proses
* Hasil pekerjaan
  + - 1. Sinkronisasi koleksi yang ada di pengelolaan e-Deposit dengan data ISBN (received\_date)
* Langkah-langkah:
* Tools yang digunakan:
* Algoritma / Flowchart:
* Coding
  + 1. v\_isbn\_jenis (database SQL server/ ISBN)

Create view v\_isbn\_jenis as SELECT

a.kd\_penerbit,

a.kd\_penerbit\_dtl,

a.title,

a.kepeng creator,

a.tahun\_terbit year,

a.edisi edition,

a.seri,

CASE WHEN a.title like '%sumber elektronis%'

THEN 'elek'

ELSE 'cetak'

END jenis,

b.keterangan,

a.peruntukan,

a.tahun\_terbit,

(a.kd\_penerbit + a.kd\_penerbit\_dtl) bar\_val,

ISNULL(a.kepeng, '') kepeng,

a.edisi,

a.jml\_hlm,

a.ketebalan,

a.call\_number,

a.catatan,

a.media,

a.jejakan1,

a.jejakan2,

a.jejakan3,

a.jejakan4,

a.jejakan5,

a.subjek,

a.subjek2,

a.subjek3,

a.subjek4,

a.subjek5,

b.created\_date,

CONVERT(VARCHAR(10), b.received\_date,104) received\_date,

b.prefix\_element + '-' + b.publisher\_element + '-' + b.item\_element + '-' +

b.check\_digit AS isbnno

FROM

mst\_isbn b

LEFT JOIN mst\_penerbit\_dtl a ON (a.kd\_penerbit\_dtl = b.kd\_penerbit\_dtl)

* + 1. v\_isbn\_jenis\_elek (database SQL server/ ISBN)

Create view v\_isbn\_jenis\_elek as

select \* from v\_isbn\_jenis where jenis='elek' and received\_date is null

* + 1. v\_collection\_received (database MySql/ Edeposit)

create view v\_collection\_received as

SELECT \*,date\_format(received\_at,'%c/%e/%Y') as received\_date FROM edeposit\_v3.collections c where c.received\_at is not null

* Screenshot proses
* Hasil pekerjaan

Terlampir

## **LANGKAH-LANGKAH CLEANSING DATA PENGHIMPUN KONTEN WEB**

1. Menghapus salah satu archive jurnal yang sama namun berbeda link. Archive yang dihapus adalah archive dengan link yang lama

* Langkah-langkah:
* Ada jurnal ganti link, contoh [ui.ac.id](http://ui.ac.id) menjadi [journal.ui.ac.id](http://journal.ui.ac.id), records tidak terupdate dengan link baru
* Hapus archive
* Hapus archive\_settings
* Hapus records
* Tools yang digunakan:
* Algoritma / Flowchart:
* Coding
* Screenshot proses
* Hasil pekerjaan

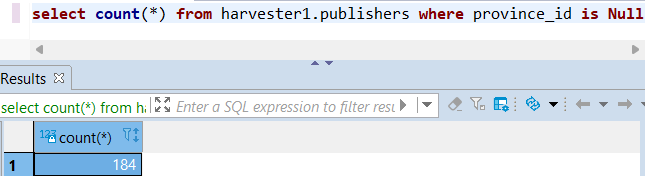
1. Cleansing files yang rusak

* Langkah-langkah:
* Baca file size jika 0 Kb dianggap file rusak
* hapus dari database
* hapus dari storage
* Tools yang digunakan:
* Algoritma / Flowchart:
* Coding
* Screenshot proses
* Hasil pekerjaan

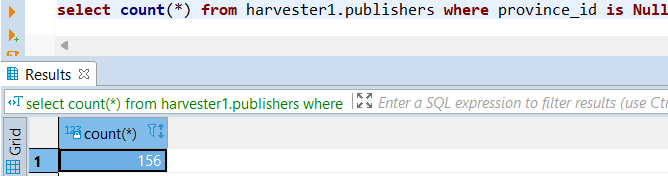
1. Pengelompokan publisher, archive , dan artikel berdasarkan provinsi

* Langkah-langkah:
* Tools yang digunakan:
* Bahasa Pemrograman Python
* Python Virtual Environment
* Python Libraries:
  + Peewee
  + Pandas
  + Playhouse
* Algoritma / Flowchart:
* Coding
* Screenshot proses

Sebelum;



Sesudah;



* Hasil pekerjaan

Terlampir

## **REKOMENDASI DAN SARAN**

# **PENUTUP**

# **LAMPIRAN**