PYTHON PROJECT (TZ)

BOZORQULOV SARDOR F-29

Reja:

1. Loyihaning maqsadi
2. Foydalanilgan texnologiyalar
3. Fayl yo’llarini aniqlash
4. ETL jarayoni bosqichlari
   1. Extract (Ma’lumotlarni yuklab olish)
   2. Transform (Ma’lumotlarni qayta ishlash)
   3. Load (Natijani aniqlash)
5. Arxitiktura sxemasi
6. Xatoliklarni qayta ishlash
7. Yakuniy natija

**1. Loyihaning maqsadi**

Bu loyiha ETL ni avtomatlashtirish uchun ishlab chiqilgan. Maqsad raw\_data faylidan xom ma’lumotlarni uqib lish, Json formatidagi malumotlarni tozalash va strukturalash, so’ngra dimension, fact, bridge jadvallarini yaratish va natijalarni xlsx qilib saqlash.

## 2. Foydalanilgan texnologiyalar

* Python 3.13
* Kutubxonalar:
  + Pandas - ma’lumotlarni o‘qish, transformatsiya va Excelga yozish
  + Json - JSON formatdagi kontentni parse qilish
  + Uuid - noyob identifikatorlar yaratish
  + OS – fayl yo‘llari bilan ishlash
  + Typing - type hint (qo‘shimcha tushuncha sifatida, kodni oson o‘qish uchun).
  + Warnings - keraksiz ogohlantirishlarni o‘chirib qo‘yish

## 3. Fayl yo‘llarini aniqlash

### df = pd.read\_csv(r"E:\MAAB\python\project\raw\_data.csv")

### current\_dir = os.path.dirname(os.path.abspath(\_\_file\_\_))

### input\_file = os.path.join(current\_dir, "raw\_data.csv")

### output\_file = os.path.join(r"E:\MAAB\python\project\final.xlsx")

* raw\_data.csv faylini o‘qib oladi.
* current\_dir → hozirgi papkani aniqlaydi.
* input\_file → kiruvchi fayl.
* output\_file → chiqadigan fayl (final.xlsx).

## 4. ETL jarayoni bosqichlari

* ***3.1 Extract (Ma’lumotlarni yuklab olish)***
  + extract\_data() metodi faylni .csv yoki .xlsx formatda ochadi.
  + raw\_content ustunidan JSON ma’lumotlar o‘qiladi.
  + JSON format xatolarga ega bo‘lsa, kod avtomatik tuzatishga harakat qiladi (ochiluvchi va yopiluvchi qavslarni aniqlash orqali).
  + Har bir satr parsed\_data ro‘yxatiga qo‘shiladi.

Natija: xom ma’lumotlar tozalanib, **Python obyektlari** sifatida qaytariladi.

* ***3.2 Transform (Ma’lumotlarni qayta ishlash)***
  + transform\_data() metodi parse qilingan ma’lumotlarni **dimension** va **fact** jadvallariga ajratadi.
  + **Dimension jadvallari**:
    - dim\_comm\_type – aloqa turi (meeting, email, chat, call va h.k.)
    - dim\_subject – suhbat yoki uchrashuv mavzusi
    - dim\_calendar – kalendar identifikatori
    - dim\_audio – audio fayl URL
    - dim\_video – video fayl URL
    - dim\_transcript – transkript fayl URL
    - dim\_user – barcha unikal foydalanuvchilar (email orqali)
  + Fact jadval:
    - dim\_user – barcha unikal foydalanuvchilar (email orqali)
  + Bridge jadval:
    - dim\_user – barcha unikal foydalanuvchilar (email orqali)
* ***3.3 Load (Natijani saqlash)***
  + load\_to\_excel() metodi barcha jadvallarni final.xlsx fayliga yozadi.
  + Har bir jadval alohida sheet sifatida saqlanadi:
    - **dim\_comm\_type**
    - **dim\_subject**
    - **dim\_calendar**
    - **dim\_audio**
    - **dim\_video**
    - **dim\_transcript**
    - **dim\_user**
    - **fact\_communication**
    - **bridge\_comm\_user**

## 4. Arxitektura sxemasi

## raw\_data.csv (xom fayl)

## ▼

## Extract (JSON parse)

## ▼

## Transform (Dimension + Fact + Bridge)

## ▼

## Load (final.xlsx)

## 5. Xatoliklarni qayta ishlash

* Fayl topilmasa → FileNotFoundError
* raw\_content ustuni bo‘lmasa → ValueError
* JSON noto‘g‘ri formatda bo‘lsa → avtomatik tuzatishga urinish
* ETL jarayonidagi boshqa xatolar → konsolga chiqariladi va traceback orqali ko‘rsatiladi
  + - Mening jadvalimda asosan scv formatni uqish va exelga yozishda kupincha xatoliklar yuz berdi.

## 6. Yakuniy natija

* final.xlsx faylida quyidagi jadvallar bo‘ladi:

1. **dim\_comm\_type**
2. **dim\_subject**
3. **dim\_calendar**
4. **dim\_audio**
5. **dim\_video**
6. **dim\_transcript**
7. **dim\_user**
8. **fact\_communication**
9. **bridge\_comm\_user**