**Скипты по созданию таблиц для БД.**

CREATE SCHEMA DS;

CREATE TABLE ds.ft\_balance\_f(

on\_date DATE NOT NULL,

account\_rk INT NOT NULL,

currency\_rk INT,

balance\_out DECIMAL(15, 2),

PRIMARY KEY (on\_date, account\_rk)

);

CREATE TABLE ds.ft\_posting\_f(

oper\_date DATE NOT NULL,

credit\_account\_rk INT NOT NULL,

debet\_account\_rk INT NOT NULL,

credit\_amount DECIMAL(15, 2),

debet\_amount DECIMAL(15, 2)

);

CREATE TABLE ds.md\_account\_d(

data\_actual\_date DATE NOT NULL,

data\_actual\_end\_date DATE NOT NULL,

account\_rk INT NOT NULL,

account\_number TEXT NOT NULL,

char\_type CHAR(1) NOT NULL,

currency\_rk INT NOT NULL,

currency\_code VARCHAR(3) NOT NULL,

PRIMARY KEY (data\_actual\_date, account\_rk)

);

CREATE TABLE ds.md\_currency\_d(

currency\_rk INT NOT NULL,

data\_actual\_date DATE NOT NULL,

data\_actual\_end\_date DATE,

currency\_code VARCHAR(3),

code\_iso\_char VARCHAR(3),

PRIMARY KEY (currency\_rk, data\_actual\_date)

);

CREATE TABLE ds.md\_exchange\_rate\_d(

data\_actual\_date DATE NOT NULL,

data\_actual\_end\_date DATE,

currency\_rk INT NOT NULL,

reduced\_cource DECIMAL(22, 6),

code\_iso\_num VARCHAR(3),

PRIMARY KEY (data\_actual\_date, currency\_rk)

);

CREATE TABLE ds.md\_ledger\_account\_s(

chapter CHAR(1),

chapter\_name TEXT,

section\_number INT,

section\_name TEXT,

subsection\_name TEXT,

ledger1\_account INT,

ledger1\_account\_name TEXT,

ledger\_account INT NOT NULL,

ledger\_account\_name TEXT,

characteristic CHAR(1),

is\_resident INT,

is\_reserve INT,

is\_reserved INT,

is\_loan INT,

is\_reserved\_assets INT,

is\_overdue INT,

is\_interest INT,

pair\_account TEXT,

start\_date DATE NOT NULL,

end\_date DATE,

is\_rub\_only INT,

min\_term VARCHAR(1),

min\_term\_measure VARCHAR(1),

max\_term VARCHAR(1),

max\_term\_measure VARCHAR(1),

ledger\_acc\_full\_name\_translit VARCHAR(1),

is\_revaluation VARCHAR(1),

is\_correct VARCHAR(1),

PRIMARY KEY (ledger\_account, start\_date)

);

CREATE SCHEMA LOGS;

CREATE TABLE logs.etl\_logs(

log\_id SERIAL PRIMARY KEY NOT NULL,

process\_name TEXT,

start\_time TIMESTAMP NOT NULL,

end\_time TIMESTAMP,

rows\_processed INT DEFAULT 0,

process\_status VARCHAR(20) NOT NULL

CHECK(process\_status IN ('SUCCESS', 'FAILED', 'IN\_PROGRESS')),

error\_message TEXT,

duration INTERVAL GENERATED ALWAYS AS (end\_time - start\_time) STORED

CONSTRAINT valid\_end\_time CHECK(

(process\_status IN ('SUCCESS', 'FAILED') AND end\_time IS NOT NULL) OR

(process\_status = 'IN\_PROGRESS' AND end\_time IS NULL)

)

);

Ссылка на видео на яндекс диске: https://disk.yandex.ru/client/disk?idApp=client&dialog=slider&idDialog=%2Fdisk%2F2025-07-01%2018-18-56.mkv