1 Tworzenie tabel

```
CREATE TABLE REGIONS2 (
   region_id NUMBER PRIMARY KEY,
   region_name VARCHAR2(25)
);


CREATE TABLE COUNTRIES2 (
   country_id CHAR(2) PRIMARY KEY,
   country_name VARCHAR2(40),
   region_id NUMBER,
   FOREIGN KEY (region_id) REFERENCES REGIONS2(region_id)
);

CREATE TABLE LOCATIONS2 (
   location_id NUMBER PRIMARY KEY,
   street_address VARCHAR2(40),
   postal_code VARCHAR2(12),
   city VARCHAR2(30),
   state_province VARCHAR2(25),
   country_id CHAR(2),
   FOREIGN KEY (country_id) REFERENCES COUNTRIES2(country_id)
);


CREATE TABLE DEPARTMENTS2 (
   department_id NUMBER PRIMARY KEY,
```

```
  department_name VARCHAR2(30),
  manager_id NUMBER,
  location_id NUMBER,
  FOREIGN KEY (location_id) REFERENCES LOCATIONS2(location_id)
  -- FOREIGN KEY (manager_id) dodamy później po utworzeniu EMPLOYEES2
);


CREATE TABLE JOBS2 (
  job_id VARCHAR2(10) PRIMARY KEY,
  job_title VARCHAR2(35),
  min_salary NUMBER,
  max_salary NUMBER,
  CONSTRAINT chk_salary2 CHECK (min_salary < max_salary AND max_salary >= 2000)
);


CREATE TABLE EMPLOYEES2 (
  employee_id NUMBER PRIMARY KEY,
  first_name VARCHAR2(20),
  last_name VARCHAR2(25),
  email VARCHAR2(100),
  phone_number VARCHAR2(20),
  hire_date DATE,
  job_id VARCHAR2(10),
  salary NUMBER,
  commission_pct NUMBER(2,2),
  manager_id NUMBER,
  department_id NUMBER,
  FOREIGN KEY (job_id) REFERENCES JOBS2(job_id),
  FOREIGN KEY (manager_id) REFERENCES EMPLOYEES2(employee_id),
```

```
  FOREIGN KEY (department_id) REFERENCES DEPARTMENTS2(department_id)
);


CREATE TABLE JOB_HISTORY2 (
  employee_id NUMBER,
  start_date DATE,
  end_date DATE,
  job_id VARCHAR2(10),
  department_id NUMBER,
  PRIMARY KEY (employee_id, start_date),
  FOREIGN KEY (employee_id) REFERENCES EMPLOYEES2(employee_id),
  FOREIGN KEY (job_id) REFERENCES JOBS2(job_id),
  FOREIGN KEY (department_id) REFERENCES DEPARTMENTS2(department_id)
);
```

2 Usuwanie starego projektu

```
DROP TABLE C CASCADE CONSTRAINTS;
DROP TABLE COUNTRIES CASCADE CONSTRAINTS;
DROP TABLE COUNTRIES2 CASCADE CONSTRAINTS;
DROP TABLE DEPARTMENTS CASCADE CONSTRAINTS;
DROP TABLE DEPARTMENTS2 CASCADE CONSTRAINTS;
DROP TABLE EMPLOYEES CASCADE CONSTRAINTS;
DROP TABLE EMPLOYEES2 CASCADE CONSTRAINTS;
DROP TABLE JOB_HISTORY CASCADE CONSTRAINTS;
DROP TABLE JOB_HISTORY2 CASCADE CONSTRAINTS;
DROP TABLE JOBS CASCADE CONSTRAINTS;
DROP TABLE JOBS2 CASCADE CONSTRAINTS;
```

```
DROP TABLE LOCATIONS CASCADE CONSTRAINTS
```

```
DROP TABLE LOCATIONS2 CASCADE CONSTRAINTS;
DROP TABLE M CASCADE CONSTRAINTS;
DROP TABLE REGIONS CASCADE CONSTRAINTS;
DROP TABLE REGIONS2 CASCADE CONSTRAINTS;
```

3 Tworzenie nowego projektu

```
-- Tworzenie tabeli kategorii produktów
CREATE TABLE product_categories (
  category_id NUMBER PRIMARY KEY,
  category_name VARCHAR2(100) NOT NULL,
  description VARCHAR2(500),
  created_at TIMESTAMP DEFAULT SYSTIMESTAMP,
  updated_at TIMESTAMP DEFAULT SYSTIMESTAMP
);

-- Tworzenie tabeli dostawców
CREATE TABLE suppliers (
  supplier_id NUMBER PRIMARY KEY,
  company_name VARCHAR2(100) NOT NULL,
  contact_name VARCHAR2(100),
  phone VARCHAR2(20),
```

```
    email VARCHAR2(100),
    address VARCHAR2(200),
    city VARCHAR2(50),
    country VARCHAR2(50),
    created_at TIMESTAMP DEFAULT SYSTIMESTAMP,
    updated_at TIMESTAMP DEFAULT SYSTIMESTAMP
);

-- Tworzenie tabeli produktów
CREATE TABLE products (
    product_id NUMBER PRIMARY KEY,
    product_name VARCHAR2(100) NOT NULL,
    category_id NUMBER,
    supplier_id NUMBER,
    unit_price NUMBER(10,2) NOT NULL,
    units_in_stock NUMBER DEFAULT 0,
    units_on_order NUMBER DEFAULT 0,
    reorder_level NUMBER DEFAULT 0,
    discontinued NUMBER(1) DEFAULT 0,
    created_at TIMESTAMP DEFAULT SYSTIMESTAMP,
    updated_at TIMESTAMP DEFAULT SYSTIMESTAMP,
    CONSTRAINT fk_product_category FOREIGN KEY (category_id) REFERENCES
product_categories(category_id),
    CONSTRAINT fk_product_supplier FOREIGN KEY (supplier_id) REFERENCES
suppliers(supplier_id)
);

-- Tworzenie tabeli klientów
CREATE TABLE customers (
    customer_id NUMBER PRIMARY KEY,
    company_name VARCHAR2(100) NOT NULL,
    contact_name VARCHAR2(100),
```

```
    phone VARCHAR2(20),
    email VARCHAR2(100),
    address VARCHAR2(200),
    city VARCHAR2(50),
    country VARCHAR2(50),
    created_at TIMESTAMP DEFAULT SYSTIMESTAMP,
    updated_at TIMESTAMP DEFAULT SYSTIMESTAMP
);

-- Tworzenie tabeli pracowników
CREATE TABLE employees (
    employee_id NUMBER PRIMARY KEY,
    first_name VARCHAR2(50) NOT NULL,
    last_name VARCHAR2(50) NOT NULL,
    email VARCHAR2(100) UNIQUE,
    phone VARCHAR2(20),
    hire_date DATE NOT NULL,
    job_title VARCHAR2(100),
    salary NUMBER(10,2),
    manager_id NUMBER,
    created_at TIMESTAMP DEFAULT SYSTIMESTAMP,
    updated_at TIMESTAMP DEFAULT SYSTIMESTAMP,
    CONSTRAINT fk_employee_manager FOREIGN KEY (manager_id) REFERENCES
employees(employee_id)
);

-- Tworzenie tabeli zamówień
CREATE TABLE orders (
    order_id NUMBER PRIMARY KEY,
    customer_id NUMBER,
    employee_id NUMBER,
```

```
  order_date TIMESTAMP DEFAULT SYSTIMESTAMP,
  required_date TIMESTAMP,
  shipped_date TIMESTAMP,
  ship_address VARCHAR2(200),
  ship_city VARCHAR2(50),
  ship_country VARCHAR2(50),
  status VARCHAR2(20) DEFAULT 'NEW',
  created_at TIMESTAMP DEFAULT SYSTIMESTAMP,
  updated_at TIMESTAMP DEFAULT SYSTIMESTAMP,
  CONSTRAINT fk_order_customer FOREIGN KEY (customer_id) REFERENCES customers(customer_id),
  CONSTRAINT fk_order_employee FOREIGN KEY (employee_id) REFERENCES employees(employee_id)
);

-- Tworzenie tabeli szczegółów zamówień
CREATE TABLE order_details (
  order_id NUMBER,
  product_id NUMBER,
  unit_price NUMBER(10,2) NOT NULL,
  quantity NUMBER NOT NULL,
  discount NUMBER(3,2) DEFAULT 0,
  created_at TIMESTAMP DEFAULT SYSTIMESTAMP,
  updated_at TIMESTAMP DEFAULT SYSTIMESTAMP,
  CONSTRAINT pk_order_details PRIMARY KEY (order_id, product_id),
  CONSTRAINT fk_order_detail_order FOREIGN KEY (order_id) REFERENCES orders(order_id),
  CONSTRAINT fk_order_detail_product FOREIGN KEY (product_id) REFERENCES products(product_id)
);

-- Tworzenie tabeli magazynów
CREATE TABLE warehouses (
```

```
  warehouse_id NUMBER PRIMARY KEY,
  warehouse_name VARCHAR2(100) NOT NULL,
  address VARCHAR2(200),
  city VARCHAR2(50),
  country VARCHAR2(50),
  created_at TIMESTAMP DEFAULT SYSTIMESTAMP,
  updated_at TIMESTAMP DEFAULT SYSTIMESTAMP
);

-- Tworzenie tabeli stanów magazynowych
CREATE TABLE inventory (
  warehouse_id NUMBER,
  product_id NUMBER,
  quantity NUMBER DEFAULT 0,
  created_at TIMESTAMP DEFAULT SYSTIMESTAMP,
  updated_at TIMESTAMP DEFAULT SYSTIMESTAMP,
  CONSTRAINT pk_inventory PRIMARY KEY (warehouse_id, product_id),
  CONSTRAINT fk_inventory_warehouse FOREIGN KEY (warehouse_id) REFERENCES warehouses(warehouse_id),
  CONSTRAINT fk_inventory_product FOREIGN KEY (product_id) REFERENCES products(product_id)
);
```

4 Błąd podczas wysyłania danych do Oracle

```
OMEN@HPOMEN MINGW64 /d/semestr10MAGISTERKA/ZSBD/Materiały/py
$ python data_loader.py
Traceback (most recent call last):
  File "D:\semestr10MAGISTERKA\ZSBD\Materiały\py\data_loader.py", line 218, in <module>
```

```
  loader = DataLoader()
         ^^^^^^^^^^^^
File "D:\semestr10MAGISTERKA\ZSBD\Materiały\py\data_loader.py", line 30, in __init__
  self.connection = oracledb.connect(
                    ^^^^^^^^^^^^^^^^^
File "C:\Users\OMEN\AppData\Roaming\Python\Python312\site-packages\oracledb\connection.py", line 1020, in connect
  return conn_class(dsn=dsn, pool=pool, params=params, **kwargs)
         ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
File "C:\Users\OMEN\AppData\Roaming\Python\Python312\site-packages\oracledb\connection.py", line 130, in __init__
  impl.connect(params_impl)
File "src/oracledb/impl/thin/connection.pyx", line 338, in oracledb.thin_impl.ThinConnImpl.connect
File "src/oracledb/impl/thin/connection.pyx", line 328, in oracledb.thin_impl.ThinConnImpl.connect
File "src/oracledb/impl/thin/connection.pyx", line 215, in oracledb.thin_impl.ThinConnImpl._connect_with_params
File "src/oracledb/impl/thin/connection.pyx", line 186, in oracledb.thin_impl.ThinConnImpl._connect_with_description
File "src/oracledb/impl/thin/connection.pyx", line 127, in oracledb.thin_impl.ThinConnImpl._connect_with_address
File "src/oracledb/impl/thin/protocol.pyx", line 266, in oracledb.thin_impl.Protocol._connect_phase_two
File "src/oracledb/impl/thin/protocol.pyx", line 414, in oracledb.thin_impl.Protocol._process_message
oracledb.exceptions.DatabaseError: ORA-01017: niepoprawna nazwa użytkownika/hasło; odmowa zalogowania
Help: https://docs.oracle.com/error-help/db/ora-01017/

OMEN@HPOMEN MINGW64 /d/semestr10MAGISTERKA/ZSBD/Materiały/py
$
```

Dodałem je ręcznie :

INSERT INTO employees VALUES (101, 'John', 'Doe', 'john.doe@example.com', TO_DATE('2022-01-15', 'YYYY-MM-DD'), 'IT_PROG', 6000);

INSERT INTO employees VALUES (102, 'Jane', 'Smith', 'jane.smith@example.com', TO_DATE('2021-06-10', 'YYYY-MM-DD'), 'HR_REP', 4500);

INSERT INTO employees VALUES (103, 'Robert', 'Brown', 'robert.brown@example.com', TO_DATE('2020-09-23', 'YYYY-MM-DD'), 'FI_MGR', 7000);

INSERT INTO employees VALUES (104, 'Linda', 'Johnson', 'linda.johnson@example.com', TO_DATE('2023-03-01', 'YYYY-MM-DD'), 'SA_REP', 5500);

INSERT INTO employees VALUES (105, 'Michael', 'White', 'michael.white@example.com', TO_DATE('2022-05-12', 'YYYY-MM-DD'), 'IT_PROG', 6200);

INSERT INTO employees VALUES (106, 'Emily', 'Clark', 'emily.clark@example.com', TO_DATE('2021-08-19', 'YYYY-MM-DD'), 'HR_REP', 4600);

INSERT INTO employees VALUES (107, 'David', 'Lee', 'david.lee@example.com', TO_DATE('2020-11-30', 'YYYY-MM-DD'), 'FI_MGR', 7100);

INSERT INTO employees VALUES (108, 'Sarah', 'Walker', 'sarah.walker@example.com', TO_DATE('2023-02-14', 'YYYY-MM-DD'), 'SA_REP', 5600);

INSERT INTO employees VALUES (109, 'Chris', 'Hall', 'chris.hall@example.com', TO_DATE('2022-03-22', 'YYYY-MM-DD'), 'IT_PROG', 6300);

INSERT INTO employees VALUES (110, 'Amanda', 'Young', 'amanda.young@example.com', TO_DATE('2021-07-25', 'YYYY-MM-DD'), 'HR_REP', 4700);

INSERT INTO employees VALUES (111, 'Matthew', 'King', 'matthew.king@example.com', TO_DATE('2020-10-18', 'YYYY-MM-DD'), 'FI_MGR', 7200);

INSERT INTO employees VALUES (112, 'Jessica', 'Scott', 'jessica.scott@example.com', TO_DATE('2023-01-09', 'YYYY-MM-DD'), 'SA_REP', 5700);

INSERT INTO employees VALUES (113, 'Daniel', 'Green', 'daniel.green@example.com', TO_DATE('2022-04-05', 'YYYY-MM-DD'), 'IT_PROG', 6400);

INSERT INTO employees VALUES (114, 'Laura', 'Adams', 'laura.adams@example.com', TO_DATE('2021-09-12', 'YYYY-MM-DD'), 'HR_REP', 4800);

INSERT INTO employees VALUES (115, 'James', 'Baker', 'james.baker@example.com', TO_DATE('2020-12-21', 'YYYY-MM-DD'), 'FI_MGR', 7300);

INSERT INTO employees VALUES (116, 'Olivia', 'Evans', 'olivia.evans@example.com', TO_DATE('2023-03-15', 'YYYY-MM-DD'), 'SA_REP', 5800);

```
INSERT INTO employees VALUES (117, 'Joshua', 'Turner', 'joshua.turner@example.com',
TO_DATE('2022-06-17', 'YYYY-MM-DD'), 'IT_PROG', 6500);

INSERT INTO employees VALUES (118, 'Megan', 'Parker', 'megan.parker@example.com',
TO_DATE('2021-10-28', 'YYYY-MM-DD'), 'HR_REP', 4900);

INSERT INTO employees VALUES (119, 'Andrew', 'Collins', 'andrew.collins@example.com',
TO_DATE('2020-08-07', 'YYYY-MM-DD'), 'FI_MGR', 7400);

INSERT INTO employees VALUES (120, 'Stephanie', 'Mitchell',
'stephanie.mitchell@example.com', TO_DATE('2023-04-03', 'YYYY-MM-DD'), 'SA_REP', 5900);
```

```
DESC employees;
```

```
no rows selected

Name        Null?    Type
----------- -------- -------------
EMPLOYEE_ID NOT NULL NUMBER
FIRST_NAME  NOT NULL VARCHAR2(50)
LAST_NAME   NOT NULL VARCHAR2(50)
EMAIL                VARCHAR2(100)
PHONE                VARCHAR2(20)
HIRE_DATE   NOT NULL DATE
JOB_TITLE            VARCHAR2(100)
SALARY               NUMBER(10,2)
```

MANAGER_ID NUMBER

CREATED_AT TIMESTAMP(6)

UPDATED_AT TIMESTAMP(6)