

מיני פרוייקט בסיסי נתונים

מרכז חוגים – שלב א

אילה אפטרקוט 213136813 Tmunot1234567@gmail.com

נעמה שוייצר 325447654 Naamasch1@gmail.com

תשפ"ה

תוכן עניינים

2.....	מבוא
2	תיאור מילולי של המערכת
2	הטבלאות:
5	דיאגרמת ERD
5	תרשים DSD
6	יצירת הטבלאות
7	הפעלת פקודת desc
8	הכנסת נתונים
8	הכנסת נתונים ע"י data generator
8	הכנסת נתונים ע"י text importer
9	הכנסת נתונים ע"י שימוש בתכנות
10.....	גיבוי
11.....	שחזור

מבוא

תיאור מילולי של המערכת

דאטא בייס עבור מתנ"ס קהילתי המספק חוגים וסדנאות שונות.

הטבלאות:

Rooms[החדרים אליהם ישובצו החוגים]

City[רשימת ערים]

Age_Group [טווח הגילאים המיועד לקבוצה]

Workshops [הסדנאות]

Clients [נרשמים]

Teachers[מעבירי הסדנאות]

Groups_[קבוצות]

Scheduler[שיבוצים עבור מורה, קבוצה, וחדר.]

הישויות

- City(city_num, city_name)
- Rooms(room_ID, floor, isStudio, room_capacity)
isStudio [בדיקה האם החדר מתאים לשיעור ריקוד]
(למשל)
room_capacity [כמה תלמידים החדר יכול להכיל]
- Age_Group(age_ID, minAge, maxAge, descript)
כל קבוצת גיל הינה החל מהגיל המוגדר כ"גיל מינימלי" עד לגיל המוגדר כ"גיל מקסימלי".
"תיאור" יהיה בהתאם למילת מפתח כמופרט מתחת: (כלומר נתוני הטבלה ימולאו כדלהלן)
The age groups:
 1. Toddlers 3-5
 2. Kids 6-12
 3. Teens 13-17

4. Adult 18-60

5. Pensioners 60+ (maxAge=120)

- Workshops(workshop_ID, workshop_name, workshop_price, workshop_hours, workshop_capacity)
workshop_hours [מספר שעות בשבוע]
workshop_capacity [מקסימום משתתפים בחוג]

- Clients(client_ID, client_fname, client_lname, client_addr, client_TEL, client_payment, birthday)
client_addr [כתובת]
client_TEL [מספר טלפון לקוח]
client_payment [תשלום חודשי לקוח עבור סך החוגים שלוקח]
מפתח זר - קשר של בדיוק אחד מישות "לקוחות" אל ישות "עיר" המגביל שכל לקוח גר בעיר אחת.

- Teachers(teacher_ID, teacher_fname, teacher_lname, teacher_addr, teacher_TEL, teacher_salary)
teacher_addr [כתובת]
teacher_TEL [מספר טלפון מורה]
teacher_salary [משכורת חודשית מורה]
מפתח זר - קשר של בדיוק אחד מישות "מורים" אל ישות "עיר" המגביל שכל מורה גר בעיר אחת.

- Groups_(group_ID, amount, age_ID)
amount [מספר הרשומים בקבוצה]
מפתח זר - קשר של בדיוק אחד מישות "קבוצות" (Groups_) אל ישות "קבוצות גיל" (Age_Group) המגביל שלכל קבוצה יש בדיוק טווח גילאים אחד.

- Scheduler(scheduler_ID, sc_day, sc_time, room_ID, teacher_ID, workshop_ID)
sc_day [יום בשבוע]
sc_time [שעה ביום]
(sc_day, sc_time, room_ID) is unique [נועד לוודא שלא יהיו שתי שיבוצים לאותו החדר באותו הזמן]

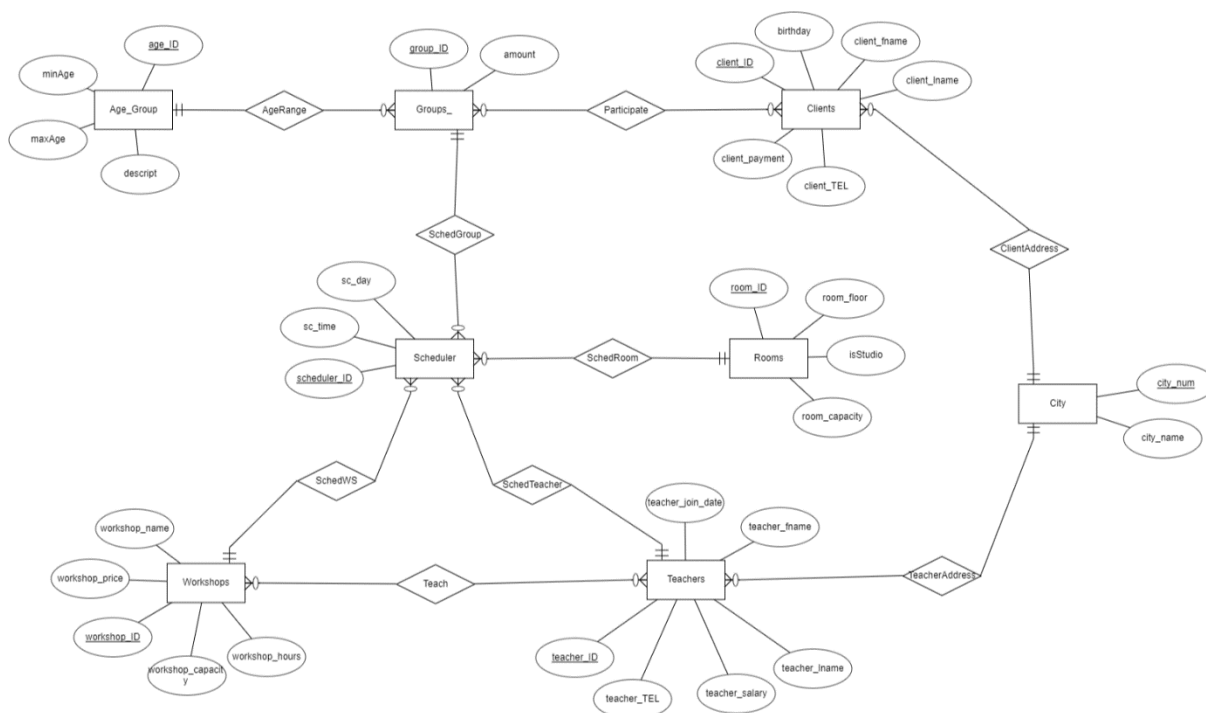
נועד לוודא שלא יהיו שתי שיבוצים של אותו מורה [(sc_day, sc_time, teacher_ID) is unique
[באותו הזמן]

מפתחות זרים - קשרים של בדיוק אחד, המגבילים את השיבוץ כך שלכל שיבוץ יהיה בדיוק חדר אחד, מורה אחד, וסדנא אחת.

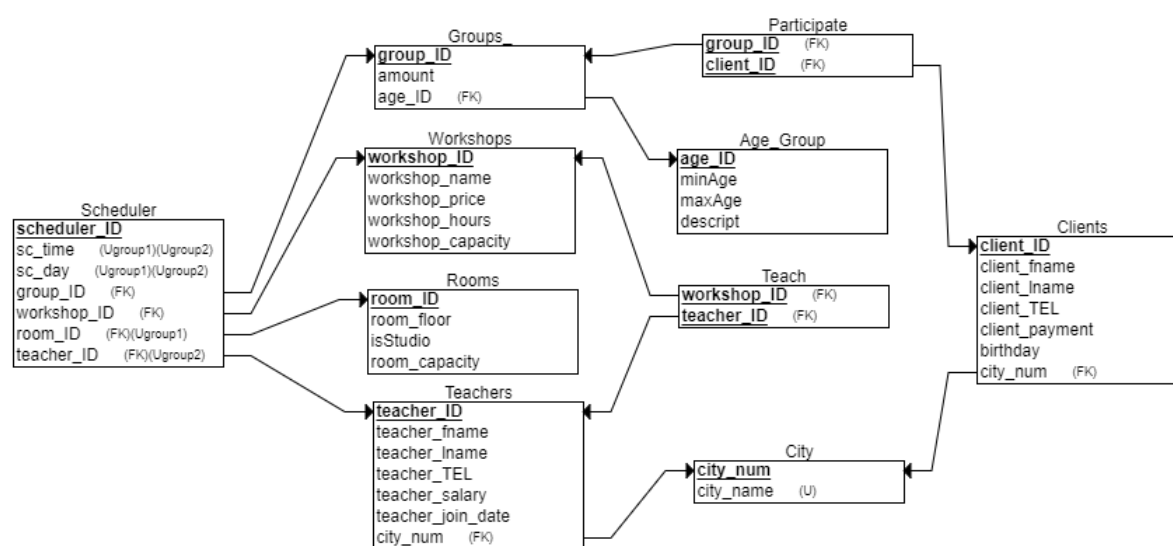
דקשרים

- Teach(teacher_ID, workshop_ID)
מורה יכול ללמד כמה מספר, וקורס יכול להיות מועבר על ידי מספר מורים.
- Participate(client_ID, group_ID)
לקוח יכול ללשתתף בכמה קבוצות, וקבוצה יכולה להכיל מספר משתתפים.

דיאגרמת ERD



תרשים DSD



יצירת הטבלאות

(סדר היצירה הוא כסדר השמופיעות בעמודות. השמאלית ואז הימנית)

```
CREATE TABLE Age_Group
(
    age_ID NUMERIC(5),
    minAge NUMERIC(5),
    maxAge NUMERIC(5),
    descript VARCHAR(15),
    PRIMARY KEY (age_ID)
);

CREATE TABLE City
(
    city_num NUMERIC(5),
    city_name VARCHAR(15),
    PRIMARY KEY (city_num),
    UNIQUE(city_name)
);

CREATE TABLE Clients
(
    client_ID NUMERIC(5),
    client_fname VARCHAR(15),
    client_lname VARCHAR(15),
    client_addr NUMERIC(5),
    client_TEL NUMERIC(5),
    client_payment NUMERIC(5),
    birthday DATE,
    PRIMARY KEY (client_ID),
    FOREIGN KEY (client_addr) REFERENCES City(city_num)
);

CREATE TABLE Workshops
(
    workshop_ID NUMERIC(5),
    workshop_name VARCHAR(15),
    workshop_price NUMERIC(5),
    workshop_hours NUMERIC(5),
    workshop_capacity NUMERIC(5),
    PRIMARY KEY (workshop_ID)
);

CREATE TABLE Teachers
(
    teacher_ID NUMERIC(5),
    teacher_fname VARCHAR(15),
    teacher_lname VARCHAR(15),
    teacher_addr NUMERIC(5),
    teacher_TEL NUMERIC(5),
    teacher_salary NUMERIC(5),
    teacher_join_date DATE,
    PRIMARY KEY (teacher_ID),
    FOREIGN KEY (teacher_addr) REFERENCES City(city_num)
);

CREATE TABLE Groups_
(
    group_ID NUMERIC(5),
    amount NUMERIC(5),
    age_ID NUMERIC(5),
    PRIMARY KEY (group_ID),
    FOREIGN KEY (age_ID) REFERENCES Age_Group(age_ID)
);
```

```
CREATE TABLE Rooms
(
    room_ID NUMERIC(5),
    room_floor NUMERIC(5),
    isStudio CHAR(1) CHECK (isStudio IN ('T', 'F')),
    room_capacity NUMERIC(5),
    PRIMARY KEY (room_ID)
);

CREATE TABLE Scheduler
(
    sc_time NUMERIC(5) CHECK (sc_time BETWEEN 6 AND 24),
    sc_day NUMERIC(5) CHECK (sc_day BETWEEN 1 AND 7),
    scheduler_ID NUMERIC(5),
    group_ID NUMERIC(5),
    workshop_ID NUMERIC(5),
    room_ID NUMERIC(5),
    teacher_ID NUMERIC(5),
    PRIMARY KEY (scheduler_ID),
    FOREIGN KEY (group_ID) REFERENCES Groups_(group_ID),
    FOREIGN KEY (workshop_ID) REFERENCES Workshops(workshop_ID),
    FOREIGN KEY (room_ID) REFERENCES Rooms(room_ID),
    FOREIGN KEY (teacher_ID) REFERENCES Teachers(teacher_ID),
    UNIQUE(sc_day, sc_time, room_ID),
    UNIQUE(sc_day, sc_time, teacher_ID)
);

CREATE TABLE Teach
(
    workshop_ID NUMERIC(5),
    teacher_ID NUMERIC(5),
    PRIMARY KEY (workshop_ID, teacher_ID),
    FOREIGN KEY (workshop_ID) REFERENCES Workshops(workshop_ID),
    FOREIGN KEY (teacher_ID) REFERENCES Teachers(teacher_ID)
);

CREATE TABLE Participate
(
    group_ID NUMERIC(5),
    client_ID NUMERIC(5),
    PRIMARY KEY (group_ID, client_ID),
    FOREIGN KEY (group_ID) REFERENCES Groups_(group_ID),
    FOREIGN KEY (client_ID) REFERENCES Clients(client_ID)
);
```

הפעלת פקודת desc

```
SQL> desc Age_Group
Name      Type      Nullable Default Comments
-----
AGE_ID    NUMBER(5)
MINAGE    NUMBER(5)  Y
MAXAGE    NUMBER(5)  Y
DESCRIPT  VARCHAR2(15) Y
```

```
SQL> desc City
Name      Type      Nullable Default Comments
-----
CITY_NUM  NUMBER(5)
CITY_NAME VARCHAR2(15) Y
```

```
SQL> desc Clients
Name      Type      Nullable Default Comments
-----
CLIENT_ID NUMBER(5)
CLIENT_FNAME VARCHAR2(15) Y
CLIENT_LNAME VARCHAR2(15) Y
CLIENT_ADDR NUMBER(5)  Y
CLIENT_TEL  NUMBER(5)  Y
CLIENT_PAYMENT NUMBER(5) Y
BIRTHDAY    DATE      Y
```

```
SQL> desc Workshops
Name      Type      Nullable Default Comments
-----
WORKSHOP_ID NUMBER(5)
WORKSHOP_NAME VARCHAR2(15) Y
WORKSHOP_PRICE NUMBER(5)  Y
WORKSHOP_HOURS NUMBER(5)  Y
WORKSHOP_CAPACITY NUMBER(5) Y
```

```
SQL> desc Teach
Name      Type      Nullable Default Comments
-----
WORKSHOP_ID NUMBER(5)
TEACHER_ID  NUMBER(5)
```

```
SQL> desc Participate
Name      Type      Nullable Default Comments
-----
GROUP_ID  NUMBER(5)
CLIENT_ID NUMBER(5)
```

```
SQL> desc Teachers
Name      Type      Nullable Default Comments
-----
TEACHER_ID NUMBER(5)
TEACHER_FNAME VARCHAR2(15) Y
TEACHER_LNAME VARCHAR2(15) Y
TEACHER_ADDR NUMBER(5)  Y
TEACHER_TEL  NUMBER(5)  Y
TEACHER_SALARY NUMBER(5) Y
TEACHER_JOIN_DATE DATE      Y
```

```
SQL> desc Groups_
Name      Type      Nullable Default Comments
-----
GROUP_ID  NUMBER(5)
AMOUNT    NUMBER(5) Y
AGE_ID    NUMBER(5) Y
```

```
SQL> desc Rooms
Name      Type      Nullable Default Comments
-----
ROOM_ID   NUMBER(5)
ROOM_FLOOR NUMBER(5) Y
ISSTUDIO  CHAR(1)  Y
ROOM_CAPACITY NUMBER(5) Y
```

```
SQL> desc Scheduler
Name      Type      Nullable Default Comments
-----
SC_TIME   NUMBER(5) Y
SC_DAY    NUMBER(5) Y
SCHEDULER_ID NUMBER(5)
GROUP_ID  NUMBER(5) Y
WORKSHOP_ID NUMBER(5) Y
ROOM_ID   NUMBER(5) Y
TEACHER_ID NUMBER(5) Y
```


הכנסת נתונים

הכנסת נתונים ע"י data generator

The screenshot shows the Data Generator tool interface for the 'ROOMS' table. The 'Owner' is set to 'AYALA' and the 'Table' is 'ROOMS'. The 'Number of records' is set to 500. The table structure is as follows:

Name	Type	Size	Data
ROOM_ID	NUMBER	5	Sequence(11,1)
ROOM_FLOOR	NUMBER	5	Random(0,4)
ISSTUDIO	CHAR	1	List('F','T')
ROOM_CAPACITY	NUMBER	5	Random(4,40)
*			

השתמשנו בשיטה זו להכניס נתונים לטבלאות:

City, Clients, Teachers, Groups_, Rooms

הסבר: ניסינו להכניס 500 רשומות לכל אחד מהנ"ל מתוך תקווה שבסופו של דבר לפחות 400 רשומות יעמדו בהגדרות שהצבנו.

הכנסת נתונים ע"י text importer

The screenshot shows the Text Importer tool interface for the 'TEACHERS' table. The 'Owner' is 'AYALA' and the 'Table' is 'TEACHERS'. The 'Commit every...' is set to 0. The 'Fields' section lists the following fields:

- Field1 -> TEACHER_ID (NUMBER)
- Field2 -> TEACHER_FNAME (VARCHAR2)
- Field3 -> TEACHER_LNAME (VARCHAR2)
- Field4 -> TEACHER_ADDR (NUMBER)
- Field5 -> TEACHER_TEL (NUMBER)
- Field6 -> TEACHER_SALARY (NUMBER)
- Field7 -> TEACHER_JOIN_DATE (DATE)

The 'Field' dropdown is set to 'TEACHER_JOIN_DATE'. The 'Fieldtype' is 'Date'. The 'SQL function' is 'to_date(%, MM/DD/YYYY)'. The 'Result Preview' section shows a table with 7 columns and 6 rows of data.

1	2	3	4	5	6	7
600	Kalvin	Kleine	1	55121	1350	1/4/1962
601	Rickie	Stenbridge	2	12827	1504	9/23/1974
602	Delora	De Paoli	3	38602	1794	2/23/1983
603	Makina	Bouts	4	35225	1759	5/12/1981
604	Trace	Hopkins	5	39920	1222	5/4/1954
605	Makila	Remden	6	75971	917	12/26/1972
606	Ainsley	Merryden	7	30226	1885	4/4/1968

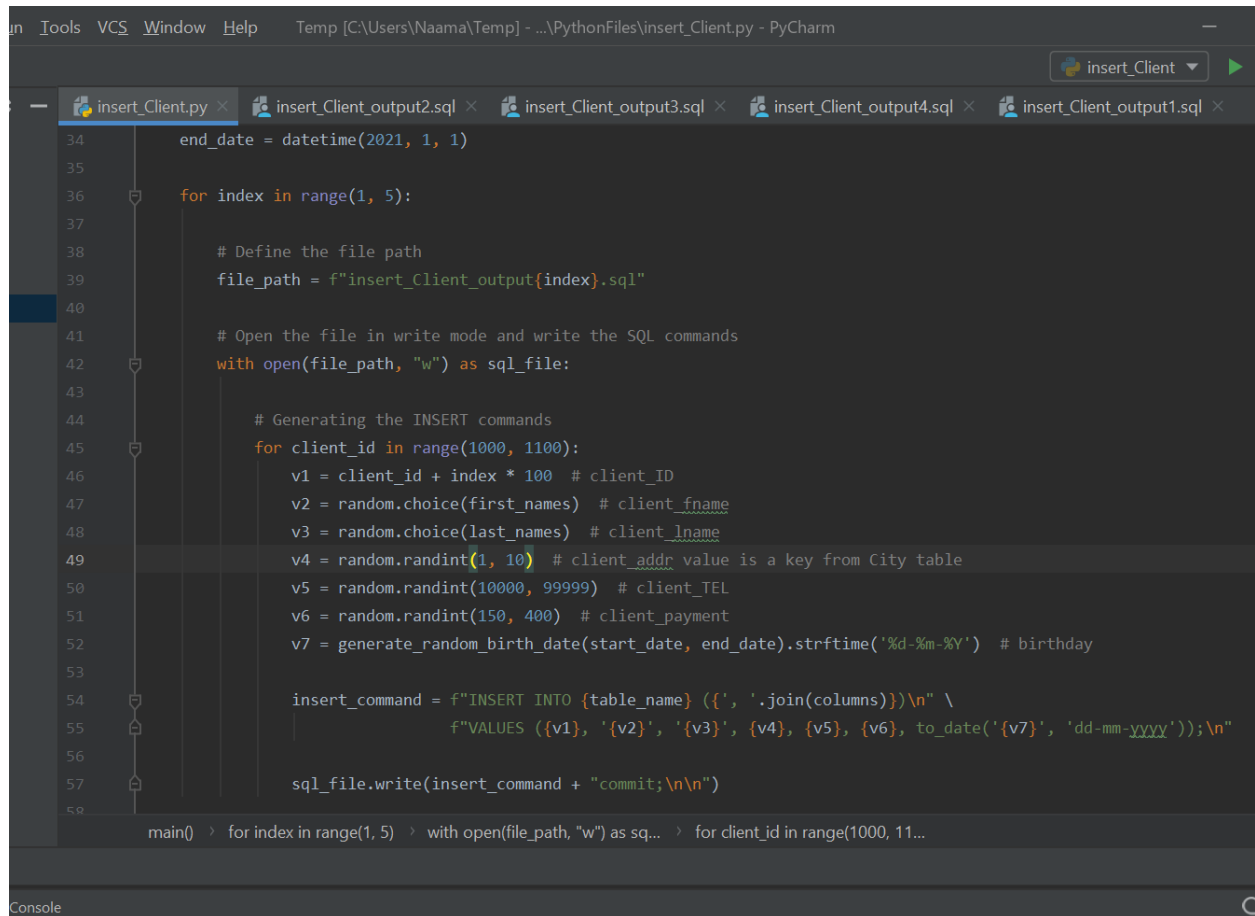
השתמשנו בשיטה זו להכניס נתונים לטבלאות:

Teachers, Workshops, Teach

הזנו ידנית נתונים ל-3 קבצי טקסט עבור כל אחת מהטבלאות הנ"ל.

הכנסת נתונים ע"י שימוש בתכנות

כתבנו סקריפט בפייתון שמייצר 400 פקודות INSERT עבור טבלת CLIENTS וכותב אותם לתוך 4 קבצי SQL שונים של 100 פקודות כל אחד. (החלוקה בשל קריסת PLSQL בהרצת 400 הפקודות בבת אחת.)



```
34 end_date = datetime(2021, 1, 1)
35
36 for index in range(1, 5):
37
38     # Define the file path
39     file_path = f"insert_Client_output{index}.sql"
40
41     # Open the file in write mode and write the SQL commands
42     with open(file_path, "w") as sql_file:
43
44         # Generating the INSERT commands
45         for client_id in range(1000, 1100):
46             v1 = client_id + index * 100 # client_ID
47             v2 = random.choice(first_names) # client_fname
48             v3 = random.choice(last_names) # client_lname
49             v4 = random.randint(1, 10) # client_addr value is a key from City table
50             v5 = random.randint(10000, 99999) # client_TEL
51             v6 = random.randint(150, 400) # client_payment
52             v7 = generate_random_birth_date(start_date, end_date).strftime('%d-%m-%Y') # birthday
53
54             insert_command = f"INSERT INTO {table_name} ({', '.join(columns)})\n" \
55                               f"VALUES ({v1}, '{v2}', '{v3}', {v4}, {v5}, {v6}, to_date('{v7}', 'dd-mm-yyyy'));\n"
56
57             sql_file.write(insert_command + "commit;\n\n")
58
59 main() > for index in range(1, 5) > with open(file_path, "w") as sql... > for client_id in range(1000, 11...
```

הסבר:

מלבד ערך הת"ז (שהוא מספר רץ) הקוד בוחר ערכים רנדומלים מתוך טווחים מוגדרים:

- שם פרטי/משפחה – נבחרים מתוך מאגרי שמות מוגדרים בקוד (20 שמות פרטיים ו-20 שמות משפחה).
- כתובת – ערך בין 1 ל-10 (ת"ז של ערים שהוכנסו ידנית למאגר נתונים)
- מספר טלפון – מספר בין 5 ספרות.
- סכום תשלום – ערך בין 150 ל-400.
- תאריך לידה – תאריך המוגרל בין טווח תאריכים הגיוני שהוגדר מראש.

	Name	Type	Compiled
	AGE_GROUP	TABLE	23/05/2024 23:25:22
	CITY	TABLE	23/05/2024 23:25:22
	CLIENTS	TABLE	23/05/2024 23:25:22
	GROUPS_	TABLE	23/05/2024 23:25:22
	PARTICIPATE	TABLE	23/05/2024 23:25:22
	ROOMS	TABLE	23/05/2024 23:25:22
	SCHEDULER	TABLE	23/05/2024 23:25:22
	TEACH	TABLE	23/05/2024 23:25:22
	TEACHERS	TABLE	23/05/2024 23:25:22
	WORKSHOPS	TABLE	23/05/2024 23:25:22

Oracle Export SQL Inserts PL/SQL Developer Log

☐ Drop tables ☒ Create tables ☐ Truncate tables ☒ Delete records ☒ Disable triggers

☐ Zip

☒ Disable foreign key constraints ☒ Include storage ☒ Include privileges

Commit every 100 records (0 = never)

Where clause

Output file C:\Users\User\study\Basnat\DBProject_8613_7654\Stage1\baucki Export

ayala@xe Exporting... Done

