מיני פרויקט בסיסי נתונים

קק"ל טיולים ומסלולים

אסתר מלכה נוסבכר 213672132
תפארת זוננברג 212886956

קרן קימת לישראל
K K L - J N F

תוכן עניינים:

שלב א

תיאור הצעת הארגון

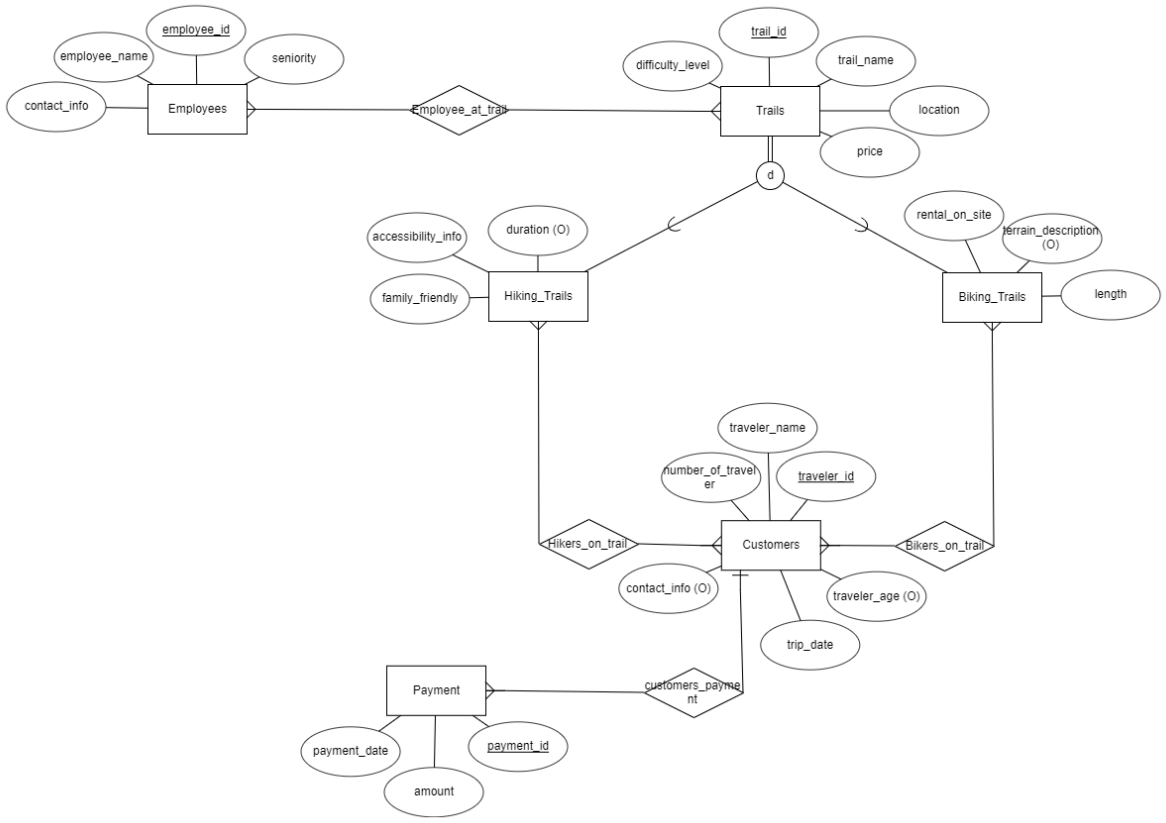
ארגון קק"ל (קרן קיימת לישראל) מתמקד בפיתוח ושימור מסלולי טיולים ושטחים פתוחים לציבור הרחב. כל אדם יכול לצאת למסלול טיול, ליהנות מהנוף ומהטבע, וללמוד על האזור.

- ניהול מסלולים: המערכת מנהלת מידע על מסלולי טיול שונים כולל פרטים כמו שם, דרגת קושי, מיקום ומחיר. המסלולים מחולקים לשני סוגים עיקריים: מסלולי הליכה ומסלולי רכיבה על אופניים, כאשר כל סוג מכיל פרטים ספציפיים כמו משך המסלול (במסלולי הליכה) או תיאור השטח (במסלולי רכיבה).
- ניהול מטיילים: המערכת שומרת מידע על מטיילים, כולל שם, גיל, פרטי קשר, מספר המטיילים ותאריך הטיול. כמו כן, המערכת מנהלת את הקשר בין מטיילים למסלולים בהם הם טיוליו.
- ניהול עובדים: המערכת כוללת מידע על עובדים המוצבים במסלולים שונים, כולל שם, ותק ופרטי קשר.
- ניהול תשלומים: המערכת מנהלת תשלומים של מטיילים עבור הטיולים, כולל סכום התשלום, תאריך התשלום וקישור בין המטייל לתשלום.

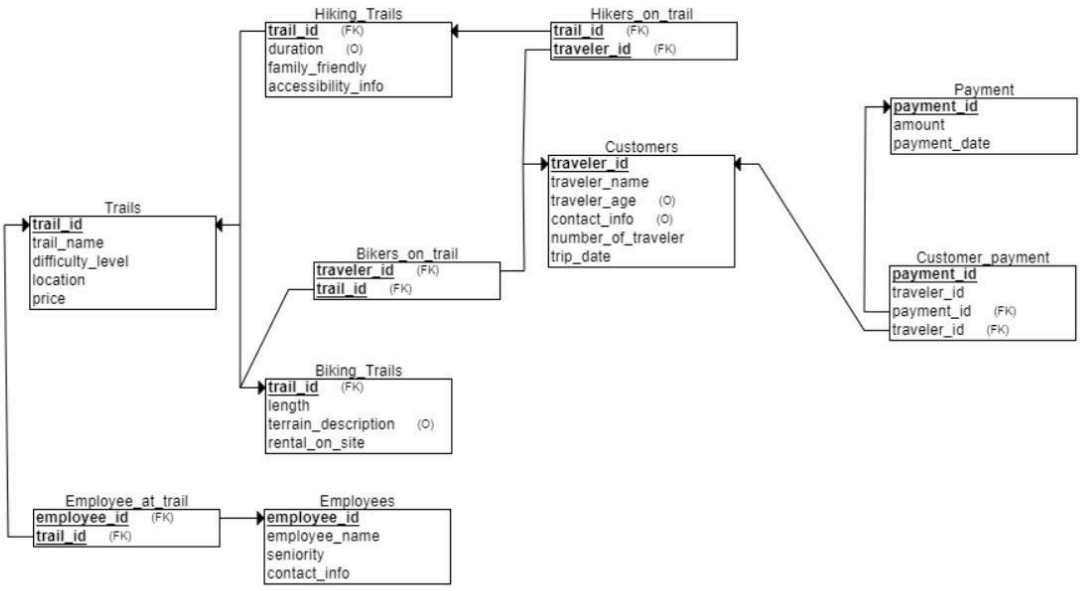
המערכת תוכננה כדי לייעל את ניהול הטיולים והמסלולים, לנהל את פרטי המטיילים והתשלומים

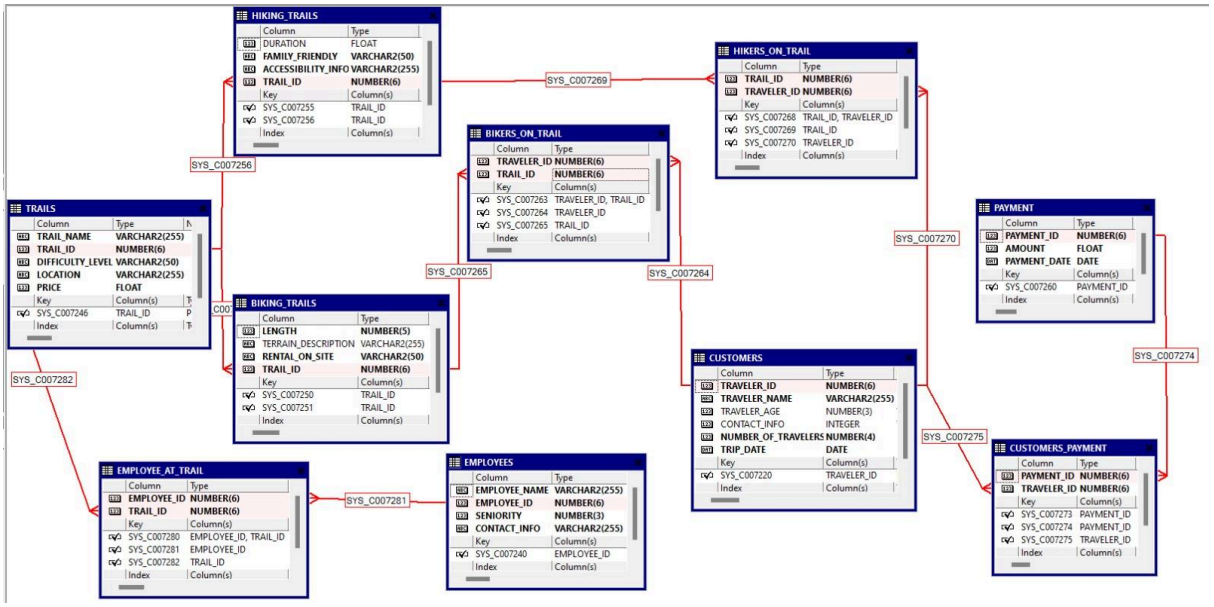
בצורה מסודרת ולספק מידע מעודכן ושירות טוב יותר למטיילים.

תרשים ERD:



תרשים DSD:




HIKING_TRAILS
Column Type
DURATION FLOAT
FAMILY_FRIENDLY VARCHAR2(50)
ACCESSIBILITY_INFO VARCHAR2(255)
TRAIL_ID NUMBER(6)
Key Column(s)
SYS_C007255 TRAIL_ID
SYS_C007256 TRAIL_ID
Index Column(s)
SYS_C007269
HIKERS_ON_TRAIL
Column Type
TRAIL_ID NUMBER(6)
TRAVELER_ID NUMBER(6)
Key Column(s)
SYS_C007268 TRAIL_ID, TRAVELER_ID
SYS_C007269 TRAIL_ID
SYS_C007270 TRAVELER_ID
Index Column(s)
SYS_C007256
BIKERS_ON_TRAIL
Column Type
TRAVELER_ID NUMBER(6)
TRAIL_ID NUMBER(6)
Key Column(s)
SYS_C007263 TRAVELER_ID, TRAIL_ID
SYS_C007264 TRAVELER_ID
SYS_C007265 TRAIL_ID
Index Column(s)
TRAILS
Column Type
TRAIL_NAME VARCHAR2(255)
TRAIL_ID NUMBER(6)
DIFFICULTY_LEVEL VARCHAR2(50)
LOCATION VARCHAR2(255)
PRICE FLOAT
Key Column(s)
SYS_C007246 TRAIL_ID
Index Column(s)
SYS_C007265
SYS_C007264
SYS_C007270
PAYMENT
Column Type
PAYMENT_ID NUMBER(6)
AMOUNT FLOAT
PAYMENT_DATE DATE
Key Column(s)
SYS_C007260 PAYMENT_ID
Index Column(s)
BIKING_TRAILS
Column Type
LENGTH NUMBER(5)
TERRAIN_DESCRIPTION VARCHAR2(255)
RENTAL_ON_SITE VARCHAR2(50)
TRAIL_ID NUMBER(6)
Key Column(s)
SYS_C007250 TRAIL_ID
SYS_C007251 TRAIL_ID
Index Column(s)
SYS_C007274
CUSTOMERS
Column Type
TRAVELER_ID NUMBER(6)
TRAVELER_NAME VARCHAR2(255)
TRAVELER_AGE NUMBER(3)
CONTACT_INFO INTEGER
NUMBER_OF_TRAVELERS NUMBER(4)
TRIP_DATE DATE
Key Column(s)
SYS_C007220 TRAVELER_ID
Index Column(s)
SYS_C007282
SYS_C007275
CUSTOMERS_PAYMENT
Column Type
PAYMENT_ID NUMBER(6)
TRAVELER_ID NUMBER(6)
Key Column(s)
SYS_C007273 PAYMENT_ID
SYS_C007274 PAYMENT_ID
SYS_C007275 TRAVELER_ID
Index Column(s)
EMPLOYEE_AT_TRAIL
Column Type
EMPLOYEE_ID NUMBER(6)
TRAIL_ID NUMBER(6)
Key Column(s)
SYS_C007280 EMPLOYEE_ID, TRAIL_ID
SYS_C007281 EMPLOYEE_ID
SYS_C007282 TRAIL_ID
Index Column(s)
SYS_C007281
EMPLOYEES
Column Type
EMPLOYEE_NAME VARCHAR2(255)
EMPLOYEE_ID NUMBER(6)
SENIORITY NUMBER(3)
CONTACT_INFO VARCHAR2(255)
Key Column(s)
SYS_C007240 EMPLOYEE_ID
Index Column(s)

החלטות עיצוב ונימוקים:

1. בחירת ישויות מרכזיות:
- מסלולים: מייצג את המסלולים השונים שקק"ל מציעה, והם חשובים מאוד לפעילות הארגון ודורשים ניהול מדויק.
- מסלולי הליכה ומסלולי רכיבה: הפיצול מאפשר ניהול משופר של סוגי המסלולים השונים, עם מידע ייחודי (למשל אורך, קשיים למסלולי רכיבה, ונגישות למסלולי הליכה).
- מטיילים: מייצג את המטיילים השונים המשתמשים במסלולים, כדי להתאים שירותים ולעקוב אחרי שימוש במסלולים.
- עובדים: חיוניים לתחזוקת המסלולים לתפעול, וניהול מידע עליהם חשוב.
- תשלומים: תיעוד התשלומים חיוני לניהול כלכלי ולמעקב אחרי השימוש במסלולים.

2. בחירת שדות ומאפיינים:
- מזהים ייחודיים (trail_id, traveler_id, employee_id, payment_id): מבטיחים ייחודיות ומעקב מדויק עבור כל ישות.
- רמת קושי ותיאור השטח (difficulty_level, terrain_description): מסייעים למטיילים לבחור את המסלול המתאים להם בהתאם לרמת הקושי ולתנאי השטח.
- מתאים למשפחות ומידע על נגישות (family_friendly, accessibility_info): מאפשרים למטיילים עם צרכים מיוחדים ולמשפחות לבחור מסלולים המתאימים להם.
- סכום תשלום ותאריך (amount, payment_date): התיעוד מאפשר ניהול כלכלי יעיל ומעקב אחרי העסקאות.

3. יחסים בין הישויות:
- מטיילים במסלול רכיבה והליכה: מייצגים את הקשרים בין המטיילים למסלולים, למעקב אחרי המשתתפים ומניעת כפילויות.
- עובד במסלול: מאפשר שיוך של עובדים לתחזוקת וניהול מסלולים מסוימים, מבטיח ניהול ובקרה יעילים בשטח.

4. ניהול תשלומים:
- התיעוד המפורט של התשלומים, כולל מזהה המטייל והתשלום, מאפשר מעקב אחרי הכנסות הארגון ווידוא תשלומים עבור שימוש במסלולים.

נימוקים:
- דיוק וייחודיות: מזהים ייחודיים לכל ישות ויחס מבטיחים אחסון מדויק וייחודי, מונעים כפילויות ושגיאות.
- גמישות והתאמה: הפיצול של מסלולים לסוגי פעילות מאפשר התאמה משופרת לצרכי המטיילים ומשפר את החיפוש וההתאמה למסל.

פעולות creatTable:

```
CREATE TABLE Employees
(
  employee_name VARCHAR(255) NOT NULL,
  employee_id NUMERIC(6) NOT NULL,
  seniority NUMERIC(3) NOT NULL,
  contact_info VARCHAR(255) NOT NULL,
  PRIMARY KEY (employee_id)
);

CREATE TABLE Trails
(
  trail_name VARCHAR(255) NOT NULL,
  trail_id NUMERIC(6) NOT NULL,
  difficulty_level VARCHAR(50) NOT NULL,
  location VARCHAR(255) NOT NULL,
  price FLOAT NOT NULL,
  PRIMARY KEY (trail_id)
);

CREATE TABLE Biking_Trails
(
  length NUMERIC(5) NOT NULL,
  terrain_description VARCHAR(255),
  rental_on_site VARCHAR(50) NOT NULL,
  trail_id NUMERIC(6) NOT NULL,
  PRIMARY KEY (trail_id),
  FOREIGN KEY (trail_id) REFERENCES Trails(trail_id)
);
```

```
CREATE TABLE Hiking_Trails
(
  duration FLOAT,
  family_friendly VARCHAR(50) NOT NULL,
  accessibility_info VARCHAR(255) NOT NULL,
  trail_id NUMERIC(6) NOT NULL,
  PRIMARY KEY (trail_id),
  FOREIGN KEY (trail_id) REFERENCES Trails(trail_id)
);

CREATE TABLE Employee_at_trail
(
  employee_id NUMERIC(6) NOT NULL,
  trail_id NUMERIC(6) NOT NULL,
  PRIMARY KEY (employee_id, trail_id),
  FOREIGN KEY (employee_id) REFERENCES Employees(employee_id),
  FOREIGN KEY (trail_id) REFERENCES Trails(trail_id)
);

CREATE TABLE Customers_payment
(
  payment_id NUMERIC(6) NOT NULL,
  traveler_id NUMERIC(6) NOT NULL,
  PRIMARY KEY (payment_id),
  FOREIGN KEY (payment_id) REFERENCES Payment(payment_id),
  FOREIGN KEY (traveler_id) REFERENCES Customers(traveler_id)
);
```

```sql
CREATE TABLE Customers
(
  traveler_id NUMERIC(6) NOT NULL,
  traveler_name VARCHAR(255) NOT NULL,
  traveler_age NUMERIC(3),
  contact_info INT,
  number_of_travelers NUMERIC(4) NOT NULL,
  trip_date DATE NOT NULL,
  PRIMARY KEY (traveler_id)
);

CREATE TABLE Bikers_on_trail
(
  traveler_id NUMERIC(6) NOT NULL,
  trail_id NUMERIC(6) NOT NULL,
  PRIMARY KEY (traveler_id, trail_id),
  FOREIGN KEY (traveler_id) REFERENCES Customers(traveler_id),
  FOREIGN KEY (trail_id) REFERENCES Biking_Trails(trail_id)
);

CREATE TABLE Hikers_on_trail
(
  trail_id NUMERIC(6) NOT NULL,
  traveler_id NUMERIC(6) NOT NULL,
  PRIMARY KEY (trail_id, traveler_id),
  FOREIGN KEY (trail_id) REFERENCES Hiking_Trails(trail_id),
  FOREIGN KEY (traveler_id) REFERENCES Customers(traveler_id)
);
```

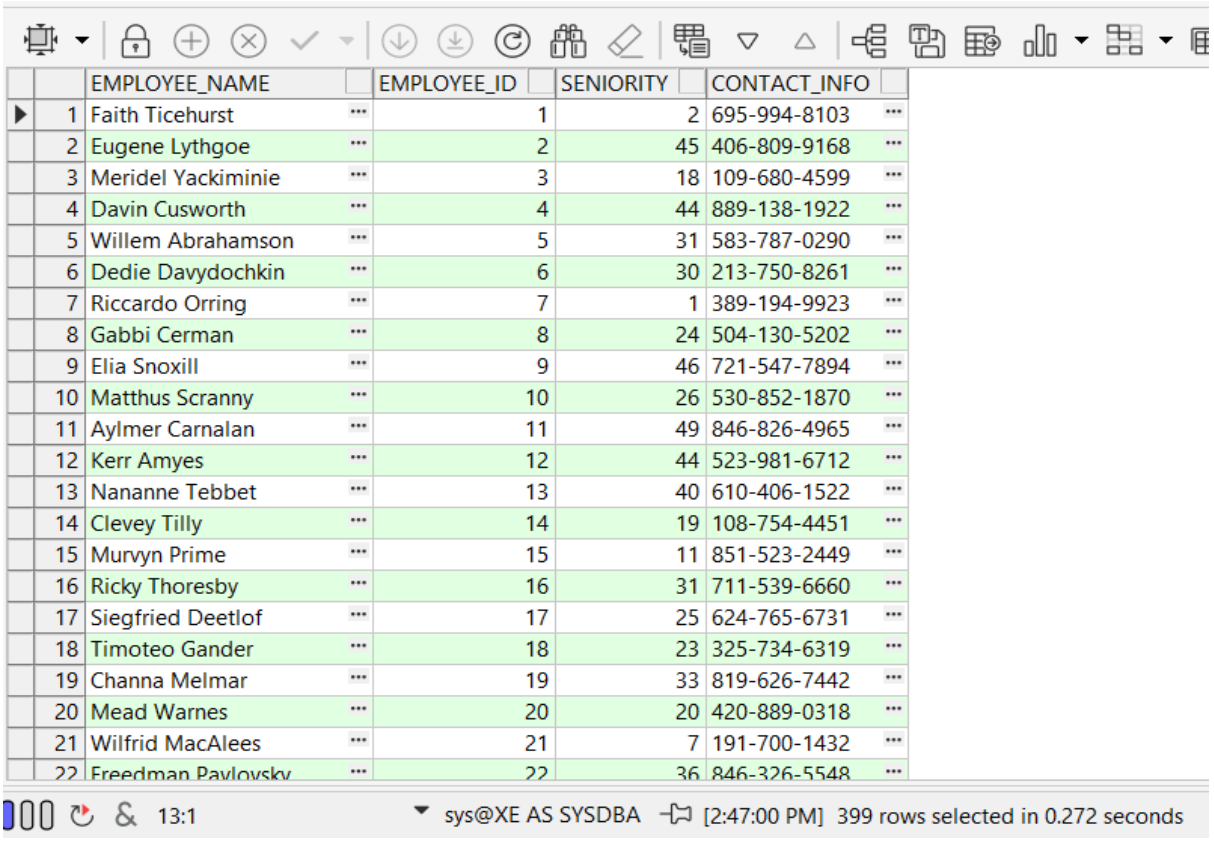
פעולות Desc:

```
Connected to Oracle Database 21c Express Edition Release 21.0.0.0.0
Connected as sys@XE AS SYSDBA

SQL>
SQL> DESC TRAILS;
Name             Type          Nullable Default Comments
---------------- ------------- -------- ------- --------
TRAIL_NAME       VARCHAR2(255)
TRAIL_ID         NUMBER(6)
DIFFICULTY_LEVEL VARCHAR2(50)
LOCATION         VARCHAR2(255)
PRICE            FLOAT
SQL> DESC BIKING_TRAILS;
Name                Type          Nullable Default Comments
------------------- ------------- -------- ------- --------
LENGTH              NUMBER(5)
TERRAIN_DESCRIPTION VARCHAR2(255) Y
RENTAL_ON_SITE      VARCHAR2(50)
TRAIL_ID            NUMBER(6)
SQL> DESC CUSTOMERS;
Name                Type          Nullable Default Comments
------------------- ------------- -------- ------- --------
TRAVELER_ID         NUMBER(6)
TRAVELER_NAME       VARCHAR2(255)
TRAVELER_AGE        NUMBER(3)     Y
CONTACT_INFO        INTEGER       Y
NUMBER_OF_TRAVELERS NUMBER(4)
TRIP_DATE           DATE
SQL> DESC BIKERS_ON_TRAIL;
```

```
Name         Type      Nullable Default Comments
------------ --------- -------- ------- --------
TRAVELER_ID  NUMBER(6)
TRAIL_ID     NUMBER(6)
SQL> DESC PAYMENT;
Name         Type      Nullable Default Comments
------------ --------- -------- ------- --------
PAYMENT_ID   NUMBER(6)
AMOUNT       FLOAT
PAYMENT_DATE DATE
SQL> DESC CUSTOMERS_PAYMENT;
Name         Type      Nullable Default Comments
------------ --------- -------- ------- --------
PAYMENT_ID   NUMBER(6)
TRAVELER_ID  NUMBER(6)
SQL> DESC EMPLOYEES;
Name          Type          Nullable Default Comments
------------- ------------- -------- ------- --------
EMPLOYEE_NAME VARCHAR2(255)
EMPLOYEE_ID   NUMBER(6)
SENIORITY     NUMBER(3)
CONTACT_INFO  VARCHAR2(255)
SQL> DESC EMPLOYEE_AT_TRAIL;
Name         Type      Nullable Default Comments
------------ --------- -------- ------- --------
EMPLOYEE_ID  NUMBER(6)
TRAIL_ID     NUMBER(6)
SQL> DESC HIKING_TRAILS;
Name               Type          Nullable Default Comments
------------------ ------------- -------- ------- --------
DURATION           FLOAT         Y
FAMILY_FRIENDLY    VARCHAR2(50)
ACCESSIBILITY_INFO VARCHAR2(255)
TRAIL_ID           NUMBER(6)
SQL> DESC HIKERS_ON_TRAIL;
Name         Type      Nullable Default Comments
------------ --------- -------- ------- --------
TRAIL_ID     NUMBER(6)
TRAVELER_ID  NUMBER(6)
```

מהנתונים בטבלות לאחר הרצת פקודות selectAll

טבלת Employees



	EMPLOYEE_NAME	EMPLOYEE_ID	SENIORITY	CONTACT_INFO
1	Faith Ticehurst	1	2	695-994-8103
2	Eugene Lythgoe	2	45	406-809-9168
3	Meridel Yackiminie	3	18	109-680-4599
4	Davin Cusworth	4	44	889-138-1922
5	Willem Abrahamson	5	31	583-787-0290
6	Dedie Davydochkin	6	30	213-750-8261
7	Riccardo Orring	7	1	389-194-9923
8	Gabbi Cerman	8	24	504-130-5202
9	Elia Snoxill	9	46	721-547-7894
10	Matthus Scranny	10	26	530-852-1870
11	Aylmer Carnalan	11	49	846-826-4965
12	Kerr Amyes	12	44	523-981-6712
13	Nananne Tebbet	13	40	610-406-1522
14	Clevey Tilly	14	19	108-754-4451
15	Murvyn Prime	15	11	851-523-2449
16	Ricky Thoresby	16	31	711-539-6660
17	Siegfried Deetlof	17	25	624-765-6731
18	Timoteo Gander	18	23	325-734-6319
19	Channa Melmar	19	33	819-626-7442
20	Mead Warnes	20	20	420-889-0318
21	Wilfrid MacAlees	21	7	191-700-1432
22	Freedman Pavlovsky	22	36	846-326-5548

13:1 sys@XE AS SYSDBA [2:47:00 PM] 399 rows selected in 0.272 seconds

טבלת

Trails

	TRAIL_NAME	TRAIL_ID	DIFFICULTY_LEVEL	LOCATION	PRICE
1	Mount Meron Trail	20001	Easy	Upper Galilee	25
2	Masada Summit Trail	20002	Moderate	Dead Sea	30
3	Ein Gedi Nature Reserve	20003	Moderate	Dead Sea	20
4	Banias Waterfall Trail	20004	Moderate	Golan Heights	35
5	Ramon Crater Rim Trail	20005	Difficult	Negev Desert	50
6	Nahal Snir Nature Trail	20006	Easy	Upper Galilee	15
7	Wadi Qelt Trail	20007	Moderate	Judean Desert	40
8	Ein Avdat Canyon Trail	20008	Moderate	Negev Desert	30
9	Mount Tabor Trail	20009	Difficult	Lower Galilee	45
10	Mount Arbel Trail	20010	Difficult	Lower Galilee	55
11	Tel Dan Nature Reserve	20011	Easy	Upper Galilee	20
12	En Gedi Waterfalls Trail	20012	Moderate	Dead Sea	35
13	Sataf Nature Trail	20013	Easy	Jerusalem Hills	10
14	Yehudiya Forest Nature R	20014	Moderate	Golan Heights	25
15	Ein Hod Artists' Trail	20015	Easy	Carmel Mountains	20
16	Nahal Kziv Trail	20016	Moderate	Upper Galilee	30
17	Mount Carmel Ridge Path	20017	Difficult	Carmel Mountains	60
18	Ein Gedi Ein Gedi Oasis	20018	Moderate	Dead Sea	35
19	Banias River Trail	20019	Easy	Golan Heights	15
20	Har HaAri (Lion Mountai	20020	Difficult	Lower Galilee	50
21	Nahal Amud Trail	20021	Moderate	Upper Galilee	40
22	Wadi Zin Trail	20022	Moderate	Negev Desert	25

1 of 956 sys@XE AS SYSDBA [2:49:28 PM] 956 rows selected in 0.859 seconds

טבלת Hikers_on_taril

	TRAIL_ID	TRAVELER_ID
1	20005	362489
2	20006	347614
3	20007	337485
4	20007	358368
5	20008	382744
6	20010	341258
7	20011	304909
8	20011	343493
9	20012	320582
10	20012	356309
11	20013	304368
12	20013	351731
13	20015	327072
14	20015	371040
15	20015	382090
16	20016	320716
17	20017	336289
18	20020	352966
19	20021	313501
20	20021	317503
21	20021	381015
22	20022	372179

1 of 457 0:01 sys@XE AS SYSDBA [2:54:24 PM] 457 rows selected in 1.149 seconds

טבלת Bikers_on_taril

	TRAVELER_ID	TRAIL_ID
1	300840	20063
2	301134	20337
3	302370	20308
4	302463	20269
5	302840	20004
6	303084	20185
7	303108	20629
8	304083	20020
9	304532	20274
10	304628	20193
11	305177	20111
12	306033	20290
13	306247	21153
14	307301	20274
15	307988	20237
16	308099	20306
17	308240	20021
18	308572	20190
19	309051	20032
20	309080	21076
21	309142	20228
22	309318	21143

1 of 440 0:01 sys@XE AS SYSDBA [2:54:50 PM] 440 rows selected in 1.409 seconds

טבלת Customers

	TRAVELER_ID	TRAVELER_NAME	TRAVELER_AGE	CONTACT_INFO	NUMBER_OF_TRAVELERS	TRIP_DATE
1059	397327	Josh Travolta	94	-4877	35	10/17/2023
1285	347234	Jerry Stanton	4	-4875	59	9/10/2022
5289	363846	Angie Guinness	74	-4874	58	8/12/2020
4665	378571	Shannon Scaggs	5	-4873	100	1/17/2023
2134	357905	Hazel Midler	2	-4872	38	9/22/2023
1473	376818	Lois Nolte	98	-4872	67	2/28/2023
3904	321675	Loren O'Sullivan	8	-4871	76	6/13/2020
5794	316371	Jesus Wayans	60	-4868	6	8/15/2021
2390	313465	Campbell O'Donnell	25	-4867	54	2/8/2020
6449	355082	Tea Leto	57	-4866	46	5/23/2021
2524	309232	Bebe Bosco	93	-4866	84	11/16/2021
740	398830	Ivan Vaughn	100	-4864	7	11/15/2023
5357	333076	Brent Numan	15	-4862	21	4/1/2020
1518	337140	Charlize Hobson	77	-4861	43	11/30/2023
3889	398363	Greg Brown	49	-4860	86	8/6/2022
6376	314137	Eric Caviezel	57	-4858	2	7/17/2022
5269	394021	Cole Frampton	43	-4858	22	1/6/2021
5264	300112	Scott Davidson	59	-4858	68	7/10/2022
6044	301600	Emily Galecki	54	-4856	89	9/17/2022
543	322515	Brooke Sartain	10	-4855	11	5/5/2020
9	391175	Delbert Belles	66	-4854	20	7/12/2023
4072	343515	Rickie Ryder	28	-4853	42	8/19/2020

3603 of 6467 0:06 sys@XE AS SYSDBA traveler_age, number(3), optional

טבלת Payment

	PAYMENT_ID	AMOUNT	PAYMENT_DATE
1	567962	2	8/26/2020
2	572495	3	12/21/2021
3	507570	2	12/24/2022
4	518548	4	5/4/2023
5	554578	4	10/7/2023
6	567886	4	12/3/2023
7	554641	4	2/10/2023
8	558268	2	2/9/2024
9	543001	5	12/14/2022
10	576551	2	1/5/2020
11	598201	2	4/21/2021
12	525490	3	11/13/2022
13	510939	4	12/5/2023
14	550008	2	10/12/2022
15	584875	4	5/19/2022
16	561573	2	1/22/2020
17	586286	4	2/4/2020
18	511517	3	6/7/2022
19	585361	1	4/18/2022
20	599002	2	4/20/2024
21	589834	3	7/30/2022
22	538123	1	10/9/2022

1 of 6006 0:09 sys@XE AS SYSDBA [2:57:41 PM] 6006 rows selected in 9.769 seconds

טבלת Customer_payment

	PAYMENT_ID	TRAVELER_ID
1	554578	395846
2	507420	315234
3	595977	341520
4	599002	328007
5	505149	352609
6	533678	340097
7	593568	320113
8	569830	350418
9	580286	348023
10	550741	336751
11	599141	307301
12	544150	379142
13	591761	331611
14	598260	307530
15	593984	393770
16	511158	331597
17	597113	352760
18	585051	399024
19	550706	392136
20	588422	362331
21	535493	323078
22	576551	370072

1 of 544 0:10 sys@XE AS SYSDBA [2:58:01 PM] 544 rows selected in 10.070 seconds

טבלת Hiking_trails

	DURATION	FAMILY_FRIENDLY	ACCESSIBILITY_INFO	TRAIL_ID
1	9	no	not wheelchair accessible	20325
2	2	no	wheelchair accessible	20322
3	3	no	wheelchair accessible	20174
4	1	yes	wheelchair accessible	20261
5	5	yes	not wheelchair accessible	20112
6	13	yes	not wheelchair accessible	20262
7	15	no	not wheelchair accessible	20210
8	2	yes	not wheelchair accessible	20205
9	10	no	wheelchair accessible	20608
10	6	no	wheelchair accessible	20122
11	14	yes	not wheelchair accessible	20637
12	12	yes	wheelchair accessible	20632
13	7	no	wheelchair accessible	20329
14	5	no	not wheelchair accessible	20237
15	9	no	wheelchair accessible	20088
16	8	no	wheelchair accessible	20415
17	8	no	not wheelchair accessible	20056
18	1	no	not wheelchair accessible	20079
19	12	yes	wheelchair accessible	20275
20	7	yes	wheelchair accessible	20296
21	9	yes	not wheelchair accessible	20215
22	14	yes	wheelchair accessible	20401

1 of 519 0:10 sys@XE AS SYSDBA [2:58:20 PM] 519 rows selected in 10.416 seconds

טבלת Biking_trails

	LENGTH	TERRAIN_DESCRIPTION	RENTAL_ON_SITE	TRAIL_ID
1	8	hilly	no	20164
2	3	hilly	no	20188
3	1	hilly	no	20293
4	2	rocky	yes	20037
5	7	smooth	yes	20143
6	8	hilly	no	20336
7	4	flat	yes	20217
8	2	hilly	yes	20404
9	7	rocky	no	20638
10	2	flat	no	20280
11	10	flat	yes	20607
12	9	rocky	yes	20165
13	3	smooth	no	20249
14	2	flat	yes	20061
15	9	smooth	no	20283
16	2	hilly	yes	20163
17	3	rocky	no	20306
18	4	flat	yes	20298
19	10	hilly	no	20281
20	6	smooth	yes	20058
21	2	smooth	yes	20234
22	8	rocky	yes	20601

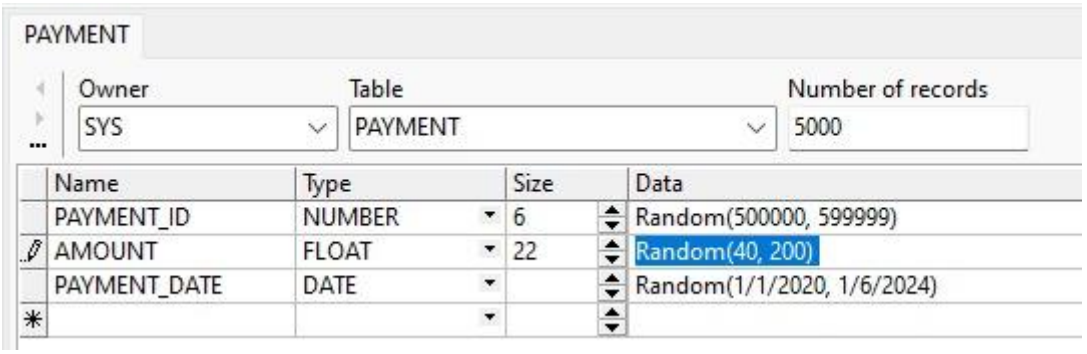
1 of 479 0:10 sys@XE AS SYSDBA [2:58:40 PM] 479 rows selected in 10.695 seconds

טבלת Employee_at_trail

	EMPLOYEE_ID	TRAIL_ID
1	2	20105
2	2	20300
3	2	20835
4	6	20969
5	8	20639
6	9	20306
7	9	21010
8	11	20831
9	11	21221
10	12	20141
11	12	20700
12	16	20974
13	16	21225
14	17	20322
15	18	20322
16	19	20730
17	19	20808
18	22	21045
19	22	21114
20	24	20136
21	24	20708
22	24	21170

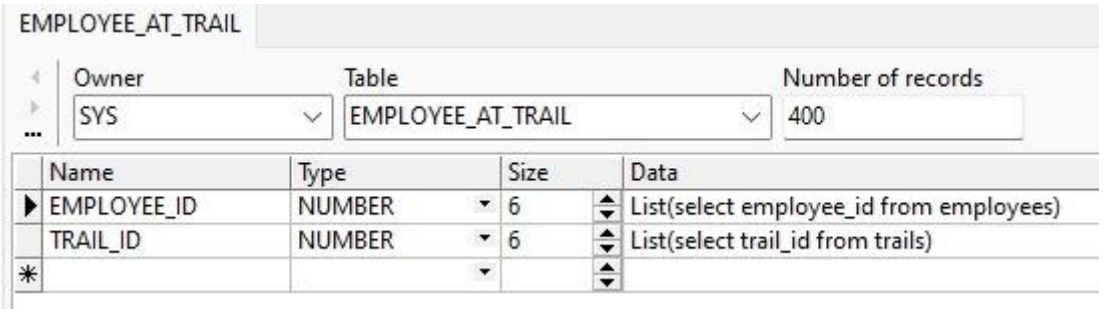
1 of 400 0:10 sys@XE AS SYSDBA [2:58:59 PM] 400 rows selected in 10.922 seconds

:data generator


PAYMENT

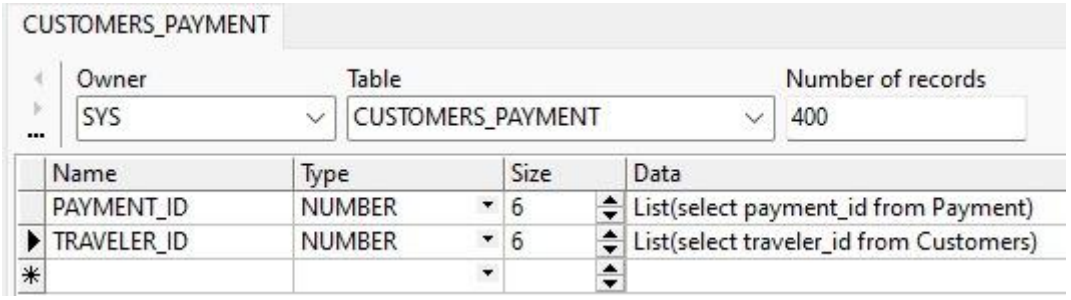
	Owner	Table	Number of records
	SYS	PAYMENT	5000

	Name	Type	Size	Data
	PAYMENT_ID	NUMBER	6	Random(500000, 599999)
	AMOUNT	FLOAT	22	Random(40, 200)
	PAYMENT_DATE	DATE		Random(1/1/2020, 1/6/2024)
*				


EMPLOYEE_AT_TRAIL

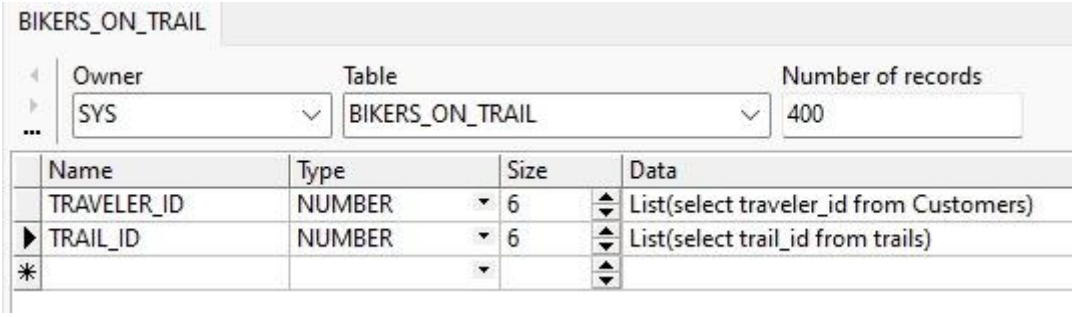
	Owner	Table	Number of records
	SYS	EMPLOYEE_AT_TRAIL	400

	Name	Type	Size	Data
▶	EMPLOYEE_ID	NUMBER	6	List(select employee_id from employees)
	TRAIL_ID	NUMBER	6	List(select trail_id from trails)
*				


CUSTOMERS_PAYMENT

	Owner	Table	Number of records
	SYS	CUSTOMERS_PAYMENT	400

	Name	Type	Size	Data
	PAYMENT_ID	NUMBER	6	List(select payment_id from Payment)
▶	TRAVELER_ID	NUMBER	6	List(select traveler_id from Customers)
*				


BIKERS_ON_TRAIL

	Owner	Table	Number of records
	SYS	BIKERS_ON_TRAIL	400

	Name	Type	Size	Data
	TRAVELER_ID	NUMBER	6	List(select traveler_id from Customers)
▶	TRAIL_ID	NUMBER	6	List(select trail_id from trails)
*				

HIKERS_ON_TRAIL

Owner	Table	Number of records
SYS	HIKERS_ON_TRAIL	400

Name	Type	Size	Data
TRAIL_ID	NUMBER	6	List(select trail_id from Trails)
TRAVELER_ID	NUMBER	6	List(select traveler_id from Customers)

BIKING_TRAILS

Owner	Table	Number of records
SYS	BIKING_TRAILS	400

Name	Type	Size	Data
LENGTH	NUMBER	5	Random(1, 10)
TERRAIN_DESCRIPTION	VARCHAR2	255	List('rocky', 'smooth', 'hilly', 'flat')
RENTAL_ON_SITE	VARCHAR2	50	List('yes', 'no')
TRAIL_ID	NUMBER	6	List(select trail_id from Trails)

HIKING_TRAILS

Owner	Table	Number of records
SYS	HIKING_TRAILS	400

Name	Type	Size	Data
DURATION	FLOAT	22	Random(1, 15)
FAMILY_FRIENDLY	VARCHAR2	50	List('yes', 'no')
ACCESSIBILITY_INFO	VARCHAR2	255	List('wheelchair accessible', 'not wheelchair accessible')
TRAIL_ID	NUMBER	6	List(select trail_id from Trails)

CUSTOMERS

Owner	Table	Number of records
SYS	CUSTOMERS	3000

Name	Type	Size	Data
TRAVELER_ID	NUMBER	6	Random(300000, 399999)
TRAVELER_NAME	VARCHAR2	255	FirstName ' ' LastName
TRAVELER_AGE	NUMBER	3	Random(1, 100)
CONTACT_INFO	NUMBER		'05' [0] '-' [000] '-' [0000]
NUMBER_OF_TRAVELEI	NUMBER	4	Random(1, 100)
TRIP_DATE	DATE		Random(1/1/2020, 1/6/2024)

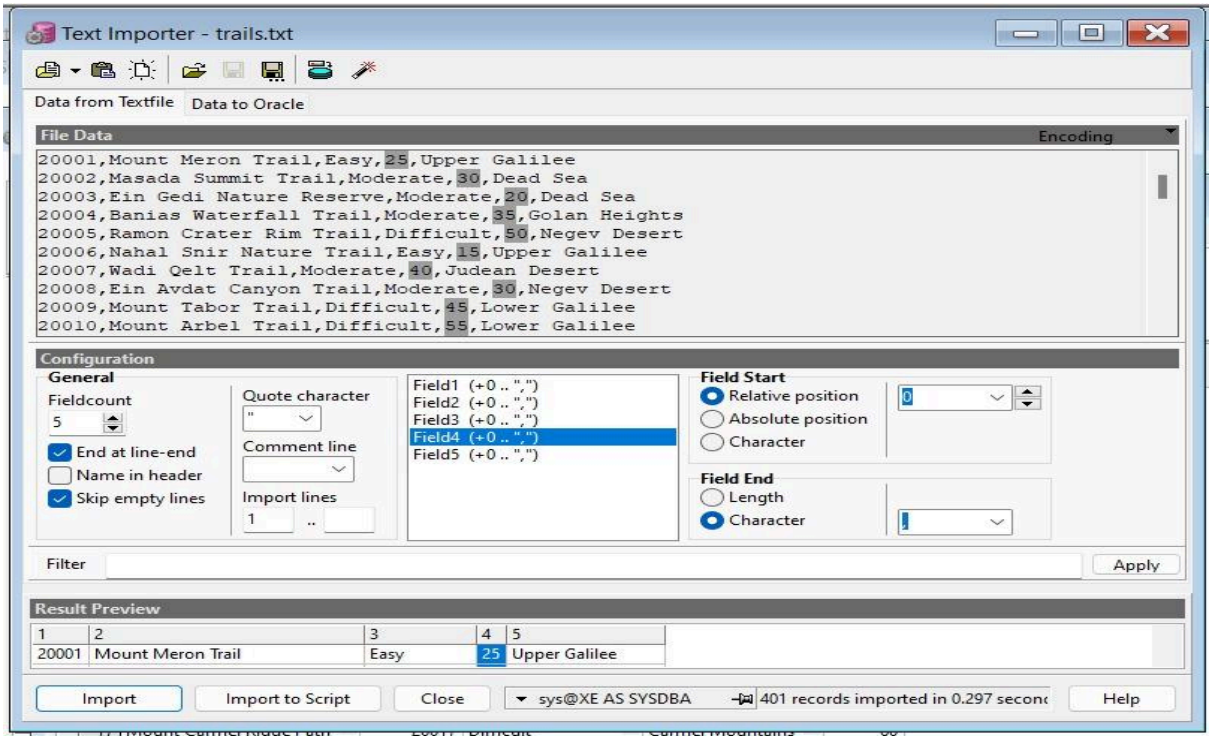
העלאת נתונים מקובץ txt.

Setup | data-iOYa6X | data-ybibfU | data-pLRKR | trails | README | README | exercism_my

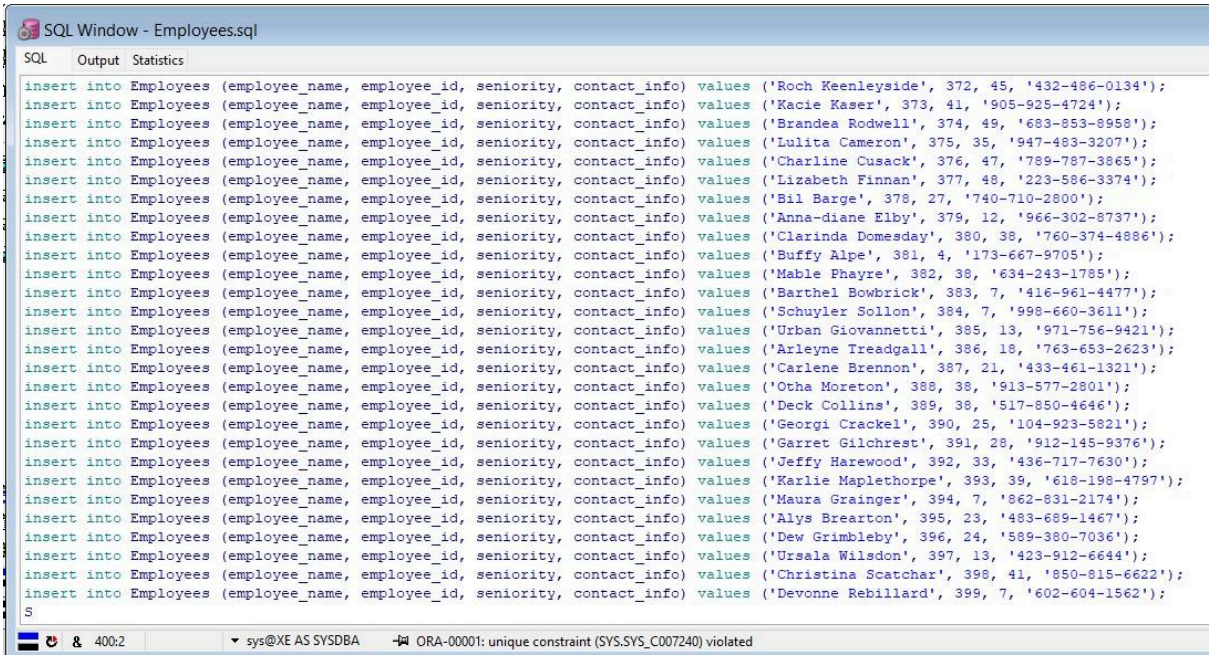
File Edit View

```
20001,Mount Meron Trail,Easy,25,Upper Galilee, Israel
20002,Masada Summit Trail,Moderate,30,Dead Sea, Israel
20003,Ein Gedi Nature Reserve,Moderate,20,Dead Sea, Israel
20004,Banias Waterfall Trail,Moderate,35,Golan Heights, Israel
20005,Ramon Crater Rim Trail,Difficult,50,Negev Desert, Israel
20006,Nahal Snir Nature Trail,Easy,15,Upper Galilee, Israel
20007,Wadi Qelt Trail,Moderate,40,Judean Desert, Israel
20008,Ein Avdat Canyon Trail,Moderate,30,Negev Desert, Israel
20009,Mount Tabor Trail,Difficult,45,Lower Galilee, Israel
20010,Mount Arbel Trail,Difficult,55,Lower Galilee, Israel
20011,Tel Dan Nature Reserve,Easy,20,Upper Galilee, Israel
20012,En Gedi Waterfalls Trail,Moderate,35,Dead Sea, Israel
20013,Sataf Nature Trail,Easy,10,Jerusalem Hills, Israel
20014,Yehudiya Forest Nature R,Moderate,25,Golan Heights, Israel
20015,Ein Hod Artists' Trail,Easy,20,Carmel Mountains, Israel
20016,Nahal Kziv Trail,Moderate,30,Upper Galilee, Israel
20017,Mount Carmel Ridge Path,Difficult,60,Carmel Mountains, Israel
20018,Ein Gedi Ein Gedi Oasis,Moderate,35,Dead Sea, Israel
20019,Banias River Trail,Easy,15,Golan Heights, Israel
20020,Har HaAri (Lion Mountai,Difficult,50,Lower Galilee, Israel
20021,Nahal Amud Trail,Moderate,40,Upper Galilee, Israel
20022,Wadi Zin Trail,Moderate,25,Negev Desert, Israel
20023,Yatir Forest Trail,Easy,10,Negev Desert, Israel
20024,Makhtesh Ramon,Difficult,55,Negev Desert, Israel
20025,Tzippori Forest Nature,Moderate,30,Lower Galilee, Israel
20026,Nahal Alexander Park,Easy,20,Sharon Plain, Israel
```

Ln 1, Col 1 24,035 characters



הכנסת נתונים דרך mockaroo:


SQL Window - Employees.sql

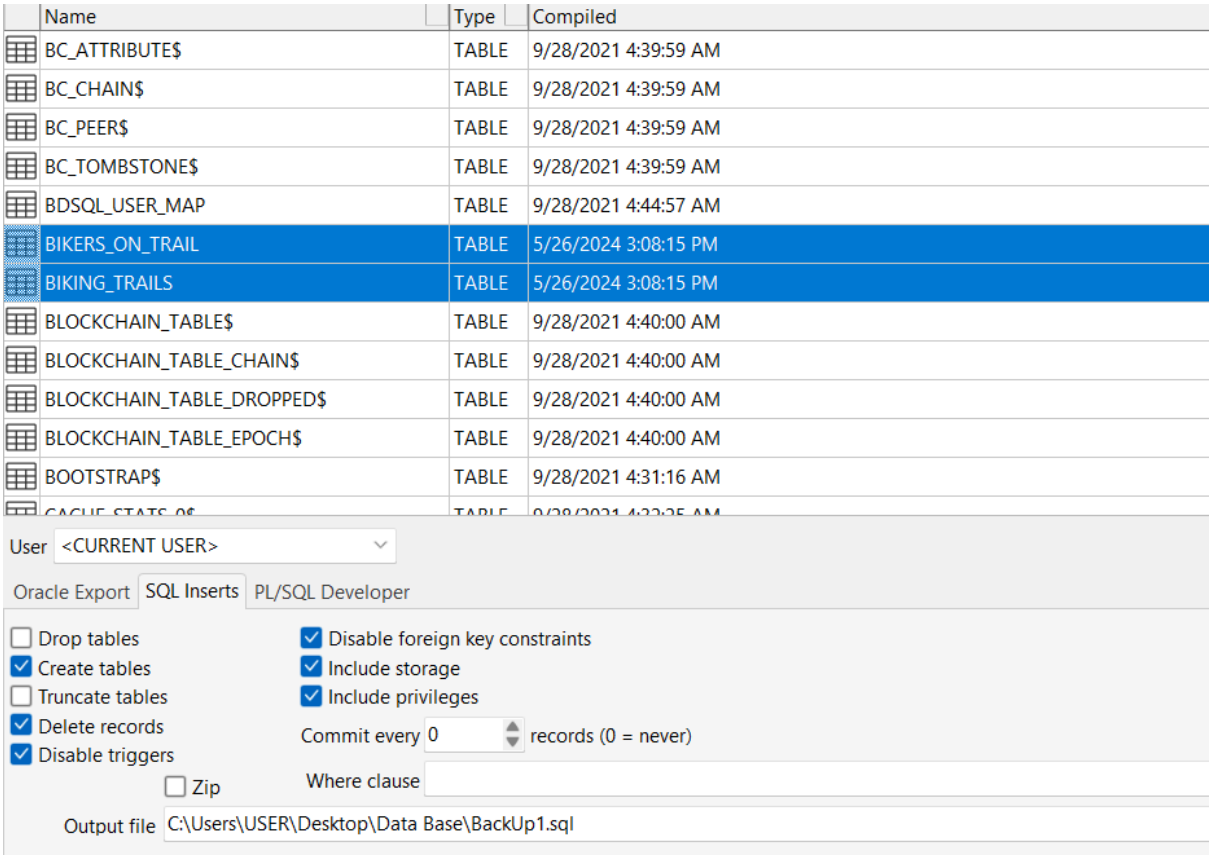
SQL Output Statistics

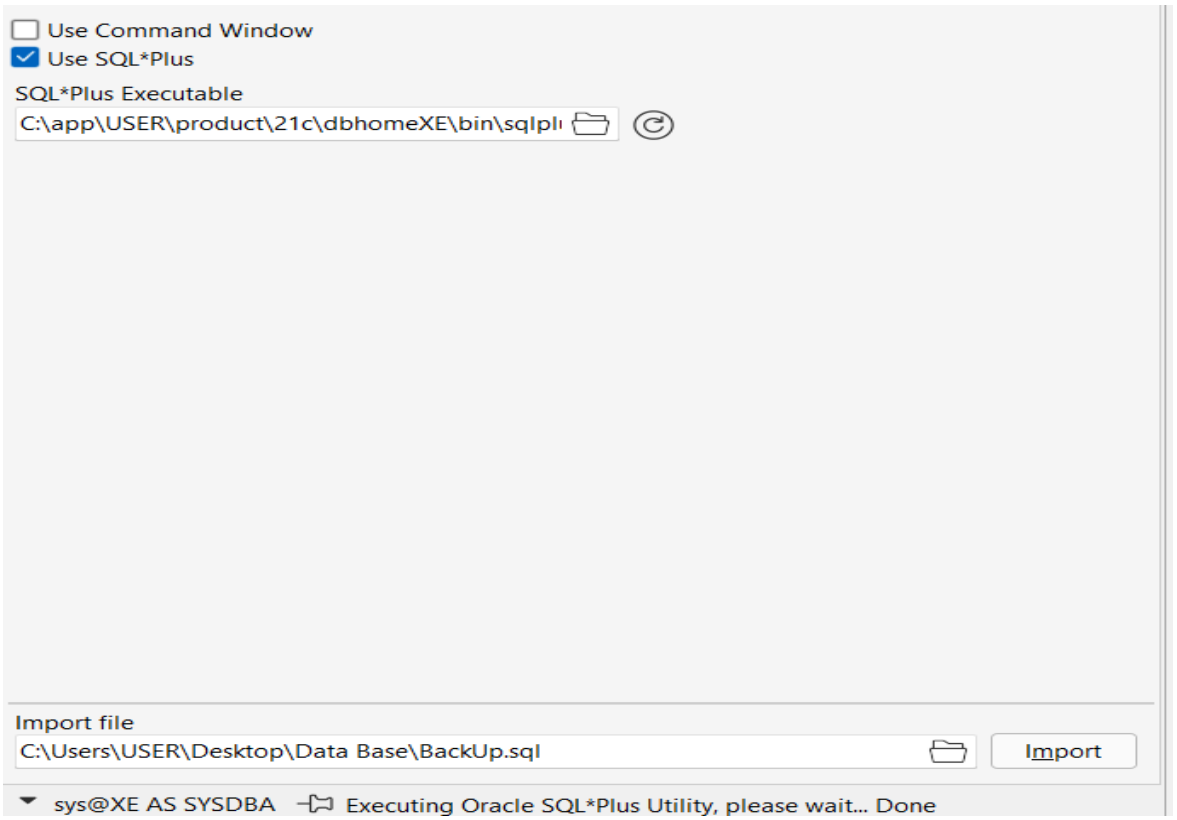
```
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Roch Keenleyside', 372, 45, '432-486-0134');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Kacie Kaser', 373, 41, '905-925-4724');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Brandea Rodwell', 374, 49, '683-853-8958');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Lulita Cameron', 375, 35, '947-483-3207');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Charline Cusack', 376, 47, '789-787-3865');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Lizabeth Finnan', 377, 48, '223-586-3374');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Bil Barge', 378, 27, '740-710-2800');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Anna-diane Elby', 379, 12, '966-302-8737');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Clarinda Domesday', 380, 38, '760-374-4886');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Buffy Alpe', 381, 4, '173-667-9705');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Mable Phayre', 382, 38, '634-243-1785');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Barthel Bowbrick', 383, 7, '416-961-4477');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Schuyler Sollon', 384, 7, '998-660-3611');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Urban Giovannetti', 385, 13, '971-756-9421');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Arleyne Treadgall', 386, 18, '763-653-2623');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Carlene Brennon', 387, 21, '433-461-1321');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Otha Moreton', 388, 38, '913-577-2801');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Deck Collins', 389, 38, '517-850-4646');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Georgi Crackel', 390, 25, '104-923-5821');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Garret Gilchrest', 391, 28, '912-145-9376');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Jeffy Harewood', 392, 33, '436-717-7630');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Karlie Maplethorpe', 393, 39, '618-198-4797');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Maura Grainger', 394, 7, '862-831-2174');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Alys Brearton', 395, 23, '483-689-1467');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Dew Grimbleby', 396, 24, '589-380-7036');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Ursala Wilsdon', 397, 13, '423-912-6644');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Christina Scatchar', 398, 41, '850-815-6622');
insert into Employees (employee_name, employee_id, seniority, contact_info) values ('Devonne Rebillard', 399, 7, '602-604-1562');
S
```

400:2 sys@XE AS SYSDBA ORA-00001: unique constraint (SYS.SYS_C007240) violated

תהליך הגייבוי:

Export:





```
prompt PL/SQL Developer import file
prompt Created on Sunday, 26 May 2024 by Esther Malka
set feedback off
set define off
prompt Creating TRAILS...
create table TRAILS
(
  trail_name       VARCHAR2(255) not null,
  trail_id         NUMBER(6) not null,
  difficulty_level VARCHAR2(50) not null,
  location         VARCHAR2(255) not null,
  price            FLOAT not null
)
;
alter table TRAILS
  add primary key (TRAIL_ID)
 ;

prompt Creating BIKING_TRAILS...
create table BIKING_TRAILS
(
  length              NUMBER(5) not null,
  terrain_description VARCHAR2(255),
  rental_on_site      VARCHAR2(50) not null,
  trail_id            NUMBER(6) not null
)
;
alter table BIKING_TRAILS
  add primary key (TRAIL_ID)
  ;
alter table BIKING_TRAILS
  add foreign key (TRAIL_ID)
  references TRAILS (TRAIL_ID);

prompt Creating CUSTOMERS...
create table CUSTOMERS
(
  traveler_id         NUMBER(6) not null,
  traveler_name       VARCHAR2(255) not null,
  traveler_age        NUMBER(3),
  contact_info        INTEGER,
  number_of_travelers NUMBER(4) not null,
  trip_date           DATE not null
)
;
alter table CUSTOMERS
  add primary key (TRAVELER_ID)
```

שלב ב

שאילתות:

שאילתה 1:

שאילתה זו מחשבת את הסכום הכולל של הכסף שכל שביל הרוויח מרוכבי אופניים ומטיילים. היא מסננת את השבילים שיש בהם לפחות תשלום אחד וממיינת את התוצאות לפי שם השביל.

```
--calculating the amount of money evrery trail made
SELECT trail_name,
       COALESCE(
           (SELECT SUM(P.amount)
            FROM Bikers_on_trail B
            JOIN Customers_payment CP ON B.traveler_id = CP.traveler_id
            JOIN Payment P ON CP.payment_id = P.payment_id
            WHERE B.trail_id = T.trail_id), 0)
       +
       COALESCE(
           (SELECT SUM(P.amount)
            FROM Hikers_on_trail H
            JOIN Customers_payment CP ON H.traveler_id = CP.traveler_id
            JOIN Payment P ON CP.payment_id = P.payment_id
            WHERE H.trail_id = T.trail_id), 0)
       AS total_amount
FROM Trails T
WHERE (SELECT COUNT(*)
       FROM Bikers_on_trail B
       JOIN Customers_payment CP ON B.traveler_id = CP.traveler_id
       JOIN Payment P ON CP.payment_id = P.payment_id
       WHERE B.trail_id = T.trail_id) > 0
   OR
      (SELECT COUNT(*)
       FROM Hikers_on_trail H
       JOIN Customers_payment CP ON H.traveler_id = CP.traveler_id
       JOIN Payment P ON CP.payment_id = P.payment_id
       WHERE H.trail_id = T.trail_id) > 0
ORDER BY trail name ASC;
```

19:30 sys@XE AS SYSDBA [7:56:10 PM] 5 rows selected in 0.366 seconds (more...)

	TRAIL_NAME	TOTAL_AMOUNT
1	Ein Avdat Canyon Trail	185
2	Har Amasa Summit Trail	73
3	Har Avital Summit Trail	109
4	Har Gilboa Summit Trail	181
5	Har Horshan	2

שאילתה 2:

השאילתה מחשבת את סכום הטיולים לכל שביל במהלך ה-12 חודשים האחרונים על ידי ספירת מספר המטיילים בודדים עבור כל סוג של טיול – הליכה ורכיבה על אופניים. לאחר מכן, השאילתה מוצאת את השביל שסך המטיילים בו הכי גבוה.

```
SELECT
    trail_name,
    MAX(total_hikers + total_bikers) AS total_hikers_and_bikers
FROM (
    SELECT
        T.trail_name,
        (
            SELECT COUNT(DISTINCT HOT.traveler_id)
            FROM Hikers_on_trail HOT
            JOIN Hiking_Trails HT ON HOT.trail_id = HT.trail_id
            JOIN Customers C ON HOT.traveler_id = C.traveler_id
            WHERE HT.trail_id = T.trail_id
            AND C.trip_date >= ADD_MONTHS(SYSDATE, -12)
        ) AS total_hikers,
        (
            SELECT COUNT(DISTINCT BOT.traveler_id)
            FROM Bikers_on_trail BOT
            JOIN Biking_Trails BT ON BOT.trail_id = BT.trail_id
            JOIN Customers C ON BOT.traveler_id = C.traveler_id
            WHERE BT.trail_id = T.trail_id
            AND C.trip_date >= ADD_MONTHS(SYSDATE, -12)
        ) AS total_bikers
    FROM
        Trails T
) subquery
GROUP BY
    trail_name
ORDER BY
    total hikers and bikers DESC;
```

27:11 sys@XE AS SYSDBA [7:56:15 PM] 10 rows selected in 0.045 seconds (more...)

	TRAIL_NAME	TOTAL_HIKERS_AND_BIKERS
1	Nahal Zivon	3
2	Adulam Park Trail	2
3	Nahal Kziv Trail	2
4	Nahal Kadesh Trail	2
5	Mount Tavor Ridge Trail	2

שאילתה 3:

השאילתה מחשבת את מחיר השבילים בממוצע לפי מיקומם. היא מבחרת את כל המחירים של השבילים בכל מיקום ומחשבת את הממוצע, ולבסוף מקבצת את התוצאות לפי המיקום ומסדרת אותן.

```
--Average price of trails by location
SELECT
    location,
    CEIL(AVG(price)) AS average_price
FROM (
    SELECT
        T.location,
        T.price
    FROM
        Trails T
    WHERE EXISTS (
        SELECT 1
        FROM Biking_Trails BT
        WHERE BT.trail_id = T.trail_id
    )
    UNION ALL
    SELECT
        T.location,
        T.price
    FROM
        Trails T
    WHERE EXISTS (
        SELECT 1
        FROM Hiking_Trails HT
        WHERE HT.trail_id = T.trail_id
    )
) TrailPricesByLocation
GROUP BY
    location
```

	LOCATION	AVERAGE_PRICE
1	Carmel Coast	9
2	Carmel Mountains	15
3	Dead Sea Region	16
4	Eilat Mountains	10
5	Ein Gedi	28

שאילתה 4:

השאילתה מחשבת את מספר הנוסעים על כל שבילי הטיולים לפי תאריך הטיול, ומסדרת את התוצאות לפי התאריך ואת מספר הנוסעים בסדר יורד.

```sql
-- Total number of travelers on hiking and biking trails by date
SELECT
    T.trail_name,
    C.trip_date,
    SUM(C.number_of_travelers) AS total_travelers
FROM
    Trails T
JOIN
    (
        SELECT
            B.trail_id,
            B.traveler_id
        FROM
            Bikers_on_trail B
        UNION
        SELECT
            H.trail_id,
            H.traveler_id
        FROM
            Hikers_on_trail H
    ) All_Trails ON T.trail_id = All_Trails.trail_id
JOIN
    Customers C ON All_Trails.traveler_id = C.traveler_id
GROUP BY
    T.trail_name,
    C.trip_date
ORDER BY
    C.trip_date DESC,
    total_travelers DESC;
```

	TRAIL_NAME	TRIP_DATE	TOTAL_TRAVELERS
1	Nahal Betzet Trail	6/29/2024	8
2	Nahal Gilbon Trail	6/29/2024	8
3	Nahal Mishmar HaYarden Trail	6/28/2024	6
4	Nahal Keshet Trail	6/23/2024	10
5	Har Negev Summit Trail	6/22/2024	9

שאילתת מחיקה 1:

שאילתת מחיקה מטבלה אבא ובן, מחיקת כל העובדים מעל גיל 65.

```
DELETE FROM employee_at_trail t
WHERE t.employee_id IN (
    SELECT e.employee_id
    FROM employees e
    WHERE e.age > 65
    AND (
        SELECT COUNT(*)
        FROM employee_at_trail t2
        WHERE t2.employee_id = e.employee_id
    ) < 5
);

DELETE FROM employees e
WHERE age > 65
AND (
    SELECT COUNT(*)
    FROM employee_at_trail t
    WHERE t.employee_id = e.employee_id
) <5;
```

לפני מחיקה:

Employee_at_trail:

	EMPLOYEE_ID	TRAIL_ID
1	2	20105
2	2	20300
3	2	20835
4	6	20969
5	8	20639
6	9	20306
7	9	21010
8	11	20831

1 of 400 sys@XE AS SYSDBA [11:15:55 PM] 400 rows selected in 0.275 seconds

אחרי:

	EMPLOYEE_ID	TRAIL_ID
1	2	20105
2	2	20300
3	2	20835
4	6	20969
5	8	20639
6	9	20306

1 of 313 sys@XE AS SYSDBA [11:19:27 PM] 313 rows selected in 0.481 seconds

לפני מחיקה:

Employees:

	EMPLOYEE_NAME	EMPLOYEE_ID	SENIORITY	CONTACT_INFO	AGE
1	Faith Ticehurst	1	2	695-994-8103	38
2	Eugene Lythgoe	2	45	406-809-9168	61
3	Meridel Yackiminie	3	18	109-680-4599	18
4	Davin Cusworth	4	44	889-138-1922	22
5	Willem Abrahamson	5	31	583-787-0290	29
6	Dedie Davydochkin	6	30	213-750-8261	62
7	Riccardo Orring	7	1	389-194-9923	26
8	Gabbi Cerman	8	24	504-130-5202	53
9	Elia Snoxill	9	46	721-547-7894	62

1 of 399 sys@XE AS SYSDBA [11:16:43 PM] 399 rows selected in 0.535 seconds

אחרי:

	EMPLOYEE_NAME	EMPLOYEE_ID	SENIORITY	CONTACT_INFO	AGE
1	Faith Ticehurst	1	2	695-994-8103	38
2	Eugene Lythgoe	2	45	406-809-9168	61
3	Meridel Yackiminie	3	18	109-680-4599	18
4	Davin Cusworth	4	44	889-138-1922	22
5	Willem Abrahamson	5	31	583-787-0290	29
6	Dedie Davydochkin	6	30	213-750-8261	62

1 of 301 sys@XE AS SYSDBA [11:19:24 PM] 301 rows selected in 0.251 seconds

] 98 rows deleted in 0.015 seconds

הרצה:

87 rows deleted in 0.019 seconds

שאילתת מחיקה 2:

מחיקת תשלומים שלא נעשו השנה, מטבלת האבא ובן.

```
DELETE FROM customers_payment cp
       WHERE cp.payment_id IN (
              SELECT p.payment_id
              FROM payment p
              WHERE EXTRACT(YEAR FROM p.payment_date) != EXTRACT(YEAR FROM SYSDATE));


--DeletePayment that wasnt in this past year
DELETE FROM payment p
WHERE EXTRACT(YEAR FROM p.payment_date) != EXTRACT(YEAR FROM SYSDATE);
```

לפני מחיקה:

Payment:

Select customers_payment | Select payment

	PAYMENT_ID	AMOUNT	PAYMENT_DATE
1	567962	2	8/26/2020
2	572495	3	12/21/2021
3	507570	2	12/24/2022
4	518548	4	5/4/2023
5	554578	4	10/7/2023
6	567886	4	12/3/2023

1 of 6006 0:04 sys@XE AS SYSDBA [11:43:48 PM] 6006 rows selected in 4.017 seconds

אחרי:

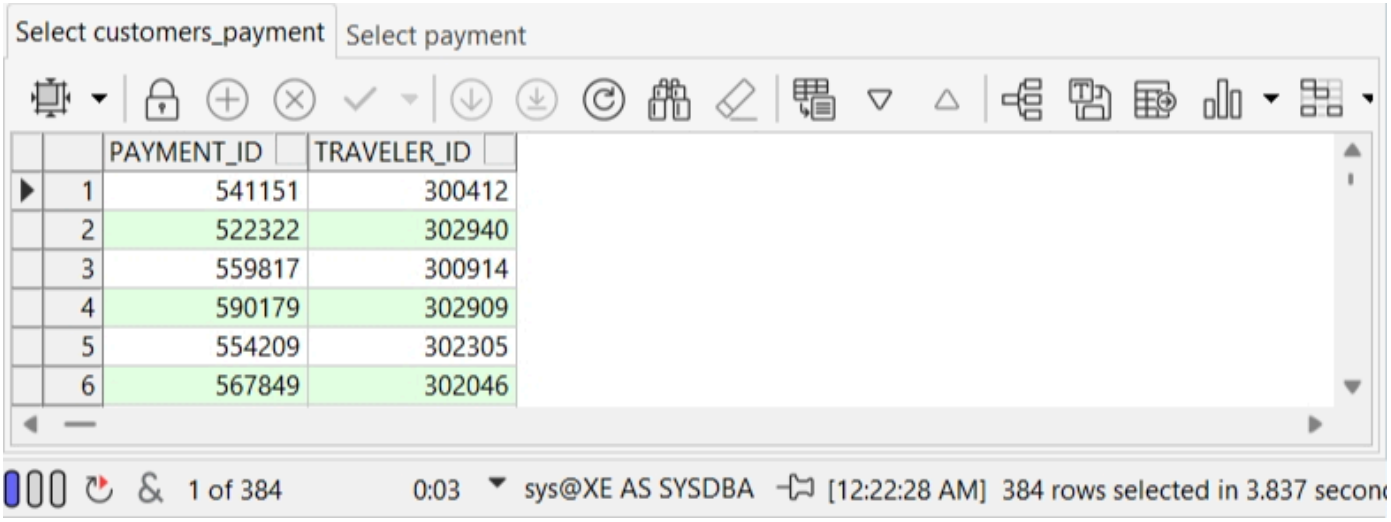
Select customers_payment | Select payment

	PAYMENT_ID	AMOUNT	PAYMENT_DATE
1	558268	2	2/9/2024
2	599002	2	4/20/2024
3	559329	4	1/14/2024
4	501008	1	1/7/2024
5	570869	3	4/3/2024
6	585613	3	3/25/2024
7	571338	3	5/12/2024

1 of 573 sys@XE AS SYSDBA [12:24:06 AM] 573 rows selected in 0.370 seconds

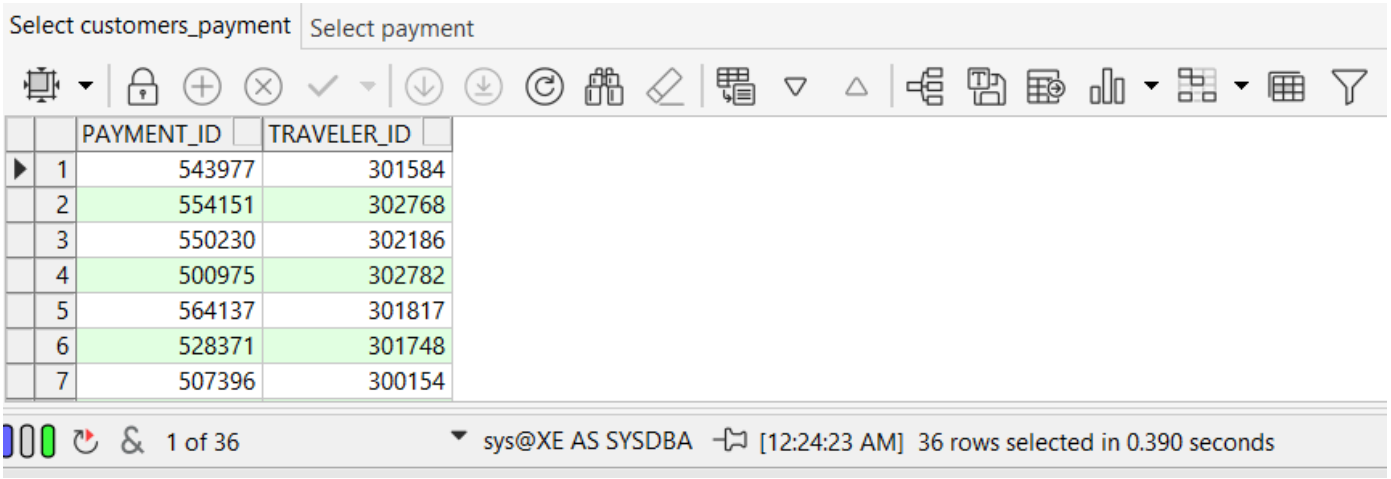
לפני מחיקה:

customer_payment



	PAYMENT_ID	TRAVELER_ID
1	541151	300412
2	522322	302940
3	559817	300914
4	590179	302909
5	554209	302305
6	567849	302046

אחרי:



	PAYMENT_ID	TRAVELER_ID
1	543977	301584
2	554151	302768
3	550230	302186
4	500975	302782
5	564137	301817
6	528371	301748
7	507396	300154

הרצאה:

5433 rows deleted in 1.002 seconds

348 rows deleted in 0.052 seconds

שאילתת עדכון 1:

עדכון גיל הנוסע בטבלת `customers` לפי ההפרש בין השנה הנוכחית לשנת הלידה, בהתחשב אם החודש והיום הנוכחיים לפני תאריך הלידה המדויק.

```
UPDATE customers c
SET c.traveler_age = EXTRACT(YEAR FROM SYSDATE) - EXTRACT(YEAR FROM c.birth_date)
                   - CASE
                       WHEN EXTRACT(MONTH FROM SYSDATE) < EXTRACT(MONTH FROM c.birth_date) THEN 1
                       WHEN EXTRACT(MONTH FROM SYSDATE) = EXTRACT(MONTH FROM c.birth_date)
                       AND EXTRACT(DAY FROM SYSDATE) < EXTRACT(DAY FROM c.birth_date) THEN 1
                       ELSE 0
                     END;
```

לפני עדכון:

	TRAVELER_ID	TRAVELER_NAME	TRAVELER_AGE	CONTACT_INFO	NUMBER_OF_TRAVELERS	TRIP_DATE	BIRTH_DATE
1	300000	Donald Jones	24	054-488-0205	93	4/16/2023	8/8/1942
2	300001	Garland Apple	11	058-816-2462	18	8/18/2021	2/24/1977
3	300002	Crispin Bacon	64	050-209-0491	17	10/4/2020	3/29/1954
4	300003	Madeline Ontiveros	28	059-948-4620	20	8/22/2023	2/23/1998
5	300004	Edgar Suvari	44	050-983-1092	9	6/6/2023	12/11/1962
6	300005	Elijah Fehr	55	052-457-6593	72	9/27/2021	1/15/1954
7	300006	Heath Kirkwood	64	056-506-6452	51	3/31/2023	9/20/1943

אחרי:

	TRAVELER_ID	TRAVELER_NAME	TRAVELER_AGE	CONTACT_INFO	NUMBER_OF_TRAVELERS	TRIP_DATE	BIRTH_DATE
1	300000	Donald Jones	81	054-488-0205	93	4/16/2023	8/8/1942
2	300001	Garland Apple	47	058-816-2462	18	8/18/2021	2/24/1977
3	300002	Crispin Bacon	70	050-209-0491	17	10/4/2020	3/29/1954
4	300003	Madeline Ontiveros	26	059-948-4620	20	8/22/2023	2/23/1998
5	300004	Edgar Suvari	61	050-983-1092	9	6/6/2023	12/11/1962
6	300005	Elijah Fehr	70	052-457-6593	72	9/27/2021	1/15/1954
7	300006	Heath Kirkwood	80	056-506-6452	51	3/31/2023	9/20/1943
8	300007	Dermot Linney	79	054-357-6556	34	1/18/2023	9/12/1944

2:1 [23] 0:01 sys@XE AS SYSDBA [12:32:54 AM] 3000 rows selected in 1.340 seconds

הרצאה:

3000 rows updated in 0.064 seconds

שאילתת עדכון 2:

עדכון מצב פתיחה של מסלול בגלל המלחמה, לפי כמה נתונים.
המיקום של המסלול הוא ב"גליל העליון".
משך המסלול (מסלולי הליכה) עולה על 10 שעות.
אורך המסלול (מסלולי אופניים) עולה על 8 קילומטרים.

```
UPDATE trails t
SET t.open_closed = 'closed'
WHERE t.location = 'Upper Galilee'
      OR t.trail_id IN (
          SELECT ht.trail_id
          FROM Hiking_Trails ht
          WHERE ht.duration > 10
      )
      OR t.trail_id IN (
          SELECT bt.trail_id
          FROM Biking_Trails bt
          WHERE bt.length > 8
);
```

לפני עדכון:

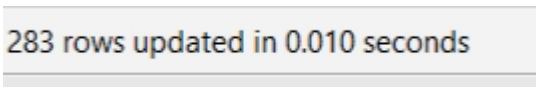
	TRAIL_NAME	TRAIL_ID	DIFFICULTY_LEVEL	LOCATION	PRICE	OPEN_CLOSED
1	Mount Meron Trail	20001	Easy	Upper Galilee	25	open
2	Masada Summit Trail	20002	Moderate	Dead Sea	30	open
3	Ein Gedi Nature Reserve	20003	Moderate	Dead Sea	20	open
4	Banias Waterfall Trail	20004	Moderate	Golan Heights	35	open
5	Ramon Crater Rim Trail	20005	Difficult	Negev Desert	50	open
6	Nahal Snir Nature Trail	20006	Easy	Upper Galilee	15	open
7	Wadi Qelt Trail	20007	Moderate	Judean Desert	40	open
8	Ein Avdat Canyon Trail	20008	Moderate	Negev Desert	30	open

15:12 sys@XE AS SYSDBA [12:38:57 AM] 956 rows selected in 0.327 seconds

אחרי:

	TRAIL_NAME	TRAIL_ID	DIFFICULTY_LEVEL	LOCATION	PRICE	OPEN_CLOSED
1	Mount Meron Trail	20001	Easy	Upper Galilee	25	closed
2	Masada Summit Trail	20002	Moderate	Dead Sea	30	open
3	Ein Gedi Nature Reserve	20003	Moderate	Dead Sea	20	open
4	Banias Waterfall Trail	20004	Moderate	Golan Heights	35	open
5	Ramon Crater Rim Trail	20005	Difficult	Negev Desert	50	closed
6	Nahal Snir Nature Trail	20006	Easy	Upper Galilee	15	closed
7	Wadi Qelt Trail	20007	Moderate	Judean Desert	40	open
8	Ein Avdat Canyon Trail	20008	Moderate	Negev Desert	30	closed
9	Mount Tabor Trail	20009	Difficult	Lower Galilee	45	open

הרצה:

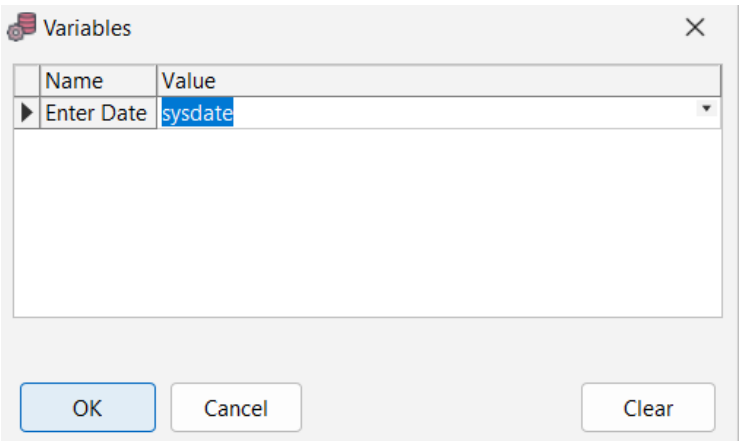


שאילתות עם פרמטרים:

1.

שאילתא מביאה את נתוני המטיילים שיש להם יום הולדת ביום הנתון.

```
--params query customers birthday
SELECT
    customer.traveler_id,
    customer.traveler_name,
    customer.traveler_age,
    customer.contact_info,
    customer.trip_date,
    TO_CHAR(SYSDATE, 'YYYY') - TO_CHAR(customer.trip_date, 'YYYY') AS trip_years_ago,
    customer.birth_date
FROM
    Customers customer
WHERE
    TO_CHAR(&<name="Enter Date" type="date" default=sysdate>, 'DD-MM') = TO_CHAR(customer.birth_date, 'DD-MM')
ORDER BY
    customer.birth_date;
```

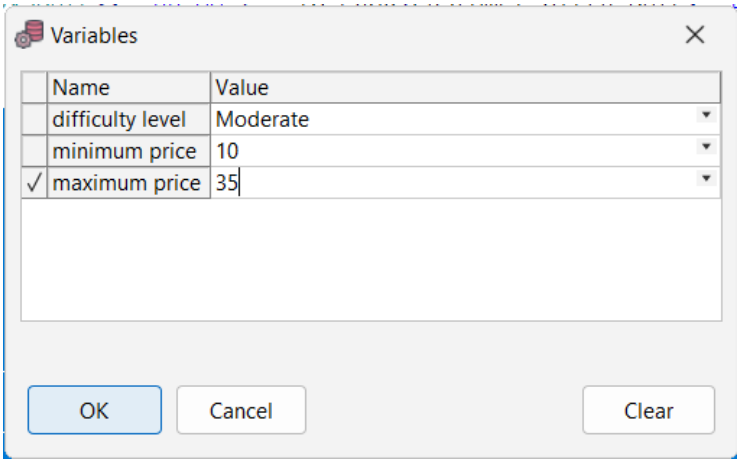


	TRAVELER_ID	TRAVELER_NAME	TRAVELER_AGE	CONTACT_INFO	TRIP_DATE	TRIP_YEARS_AGO	BIRTH_DATE
1	300555	Ricardo Blackmore	87	058-032-1910	5/28/2021	3	6/16/1929
2	301208	Chuck Heron	92	054-289-8987	11/28/2023	1	6/16/1934
3	301695	Woody Lynne	25	052-833-2607	1/14/2020	4	6/16/1936
4	301612	Gina McLean	17	052-806-2610	9/30/2022	2	6/16/1951
5	301962	Fisher Mueller-Stahl	67	051-968-3962	1/31/2022	2	6/16/1965
6	301528	Rachael Vince	49	059-041-5873	7/14/2023	1	6/16/1984

.2

שאילתא מביאה את הנתונים על המסלולים לפי רמת קושי ומחיר.

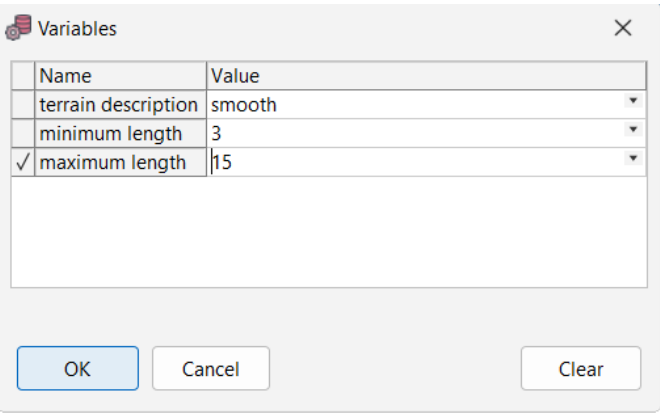
```
--params query trails by level and price
SELECT
    t.trail_name,
    t.location,
    t.price
FROM
    Trails t
WHERE
    t.difficulty_level = &<name="difficulty level" list="Easy, Moderate, Difficult"
    type="string">
    and t.price between &<name="minimum price" type=integer>
 and &<name="maximum price" type=integer>
ORDER BY  t.price;
```



	TRAIL_NAME	LOCATION	PRICE
1	Ein Gedi Nature Reserve	Dead Sea	20
2	Wadi Zin Trail	Negev Desert	25
3	Tavor Stream Trail	Lower Galilee	25
4	Yehudiya Forest Nature R	Golan Heights	25
5	Nahal Dishon Trail	Upper Galilee	30

.3
שאילתא מביאה נתונים על מסלולי רכיבה לפי סוג פני השטח ואורך המסלול.

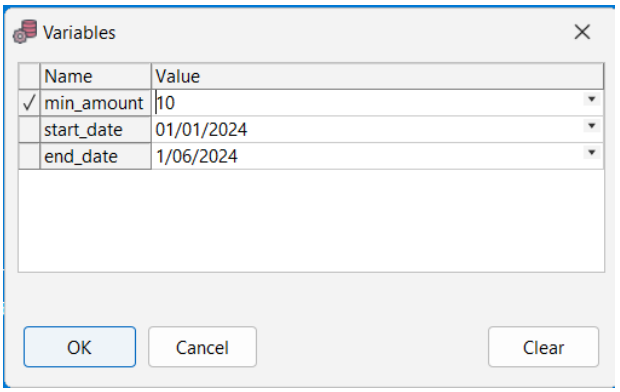
```
SELECT
    t.trail_name,
    t.location,
    b.length,
    b.terrain_description,
    t.price
FROM
    Biking_Trails b
JOIN
    Trails t ON b.trail_id = t.trail_id
WHERE
    b.terrain_description = &<name="terrain description" list="flat, hilly, smooth, rocky" type="string">
    AND b.length BETWEEN &<name="minimum length" type="integer"> AND &<name="maximum length" type="integer">
ORDER BY
    b.length;
```



	TRAIL_NAME	LOCATION	LENGTH	TERRAIN_DESCRIPTION	PRICE
1	Nahal Zippori Trail	Western Galilee	3	smooth	60
2	Nahal Hatzbani Trail	Upper Galilee	3	smooth	60
3	Mount Tabor Summit Trail	Mount Tabor	3	smooth	45
4	Mount Tabor Trail	Mount Tabor	3	smooth	45
5	Nahal Levanon Trail	Lower Galilee	3	smooth	55

.4
שאילתא מחזירה נתונים של כל התשלומים מסכום מסוים בין תאריכים ספציפיים.

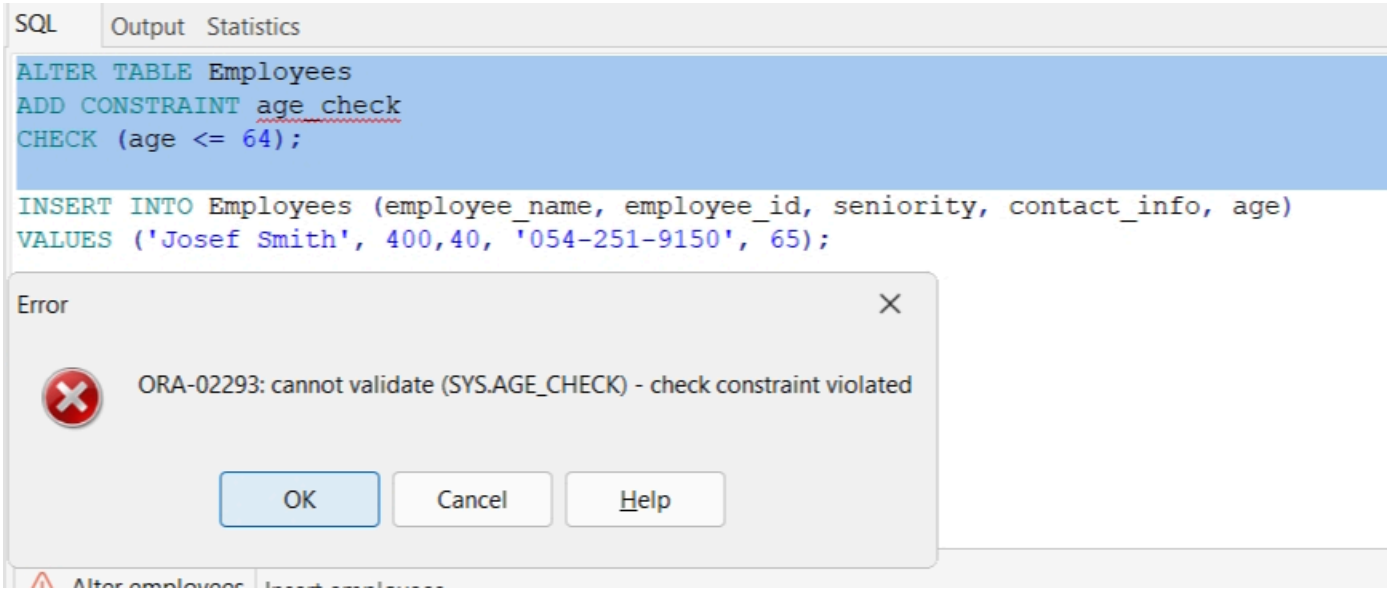
```
-- all payments made from min price between dates
SELECT
    cp.payment_id,
    c.traveler_name,
    c.contact_info,
    p.amount,
    p.payment_date
FROM
    Customers_payment cp
JOIN
    Payment p ON cp.payment_id = p.payment_id
JOIN
    Customers c ON cp.traveler_id = c.traveler_id
WHERE
    p.amount >= &<name="min_amount" type="integer">
    AND p.payment_date BETWEEN TO_DATE(&<name="start_date"  type=string
 default="01/01/2024"> ,'DD/MM/YYYY')
                          AND TO_DATE(&<name="end_date"  type=string
 default="1/06/2024"> ,'DD/MM/YYYY')
ORDER BY
    p.payment_date DESC;
```



	PAYMENT_ID	TRAVELER_NAME	CONTACT_INFO	AMOUNT	PAYMENT_DATE
1	501577	Roddy Byrne	057-694-3978	194	5/29/2024
2	500975	Kitty Hobson	056-816-4643	95	5/28/2024
3	567146	Carlene Heald	054-670-9150	80	5/22/2024
4	583774	Claude Winans	051-867-0807	88	5/21/2024
5	548467	Renee Paul	054-505-2170	191	5/16/2024
6	576610	Lori Finn	052-044-5828	172	5/12/2024

אילוצים:

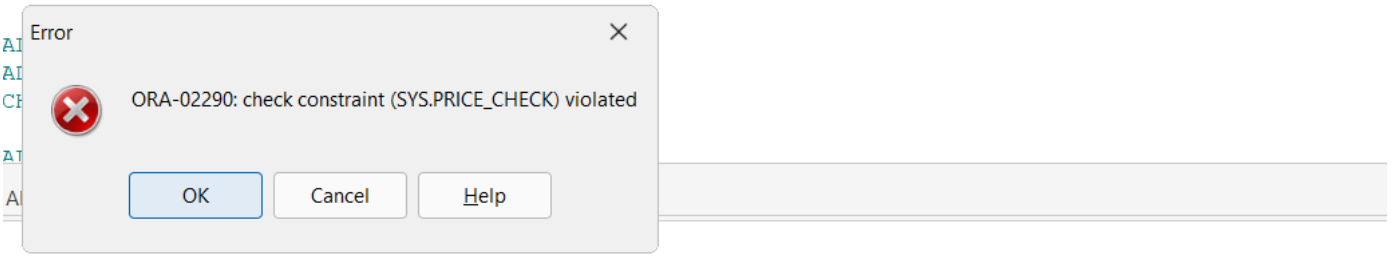
.1



.2

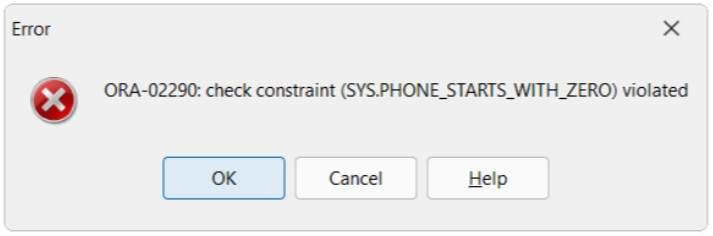
```
ALTER TABLE Trails
ADD CONSTRAINT price_check
CHECK (price >= 0);
```

```
INSERT INTO Trails (trail_name, trail_id, difficulty_level, location, price, open_closed)
VALUES ('Nachal Achziv', 21239, ' Moderate', 'Northern District HaZafon', -5, 'Open');
```



.3

```
ALTER TABLE Customers
ADD CONSTRAINT phone_starts_with_zero
CHECK (contact_info LIKE '0%');

INSERT INTO Customers (traveler_id, traveler_name, traveler_age, contact_info, number_of_travelers, trip_date)
VALUES (326550, 'Nikki Smith', 15, '453-729-1771', 3,TO_DATE('2023-12-09', 'YYYY-MM-DD'));
```



שלב ג

תכנות:

א. פונקציות-

1. חישוב דירוג רמת קושי של מסלול.

תיאור: הפונקציה get_trail_mean_rating נועדה להחזיר נתונים על דירוג רמת הקושי של מסלול מסוים על בסיס דירוגים שקיבלו מהמטיילים במסלול. הפונקציה מקבלת מזהה מסלול (p_trail_id) כפרמטר ומחזירה Ref Cursor המכיל את הדירוג הממוצע, הדירוג הגבוה ביותר והדירוג הנמוך ביותר עבור המסלול.

הקוד:

```
CREATE OR REPLACE FUNCTION get_trail_mean_rating(p_trail_id IN trails.trail_id%TYPE)
RETURN SYS_REFCURSOR
IS
    -- Variables to hold rating details
    v_sum_rating NUMBER := 0;
    v_count_rating NUMBER := 0;
    v_mean_rating NUMBER;
    v_highest_rating NUMBER := 0;
    v_lowest_rating NUMBER := 5; -- Assuming rating scale is 1 to 5

    -- Ref Cursor to return details
    v_ref_cursor SYS_REFCURSOR;

BEGIN
    -- Calculate the sum, count, highest, and lowest rating from Hikers_on_trail
    FOR rec IN (SELECT rating FROM Hikers_on_trail WHERE trail_id = p_trail_id) LOOP
        v_sum_rating := v_sum_rating + rec.rating;
        v_count_rating := v_count_rating + 1;

        IF rec.rating > v_highest_rating THEN
            v_highest_rating := rec.rating;
        END IF;

        IF rec.rating < v_lowest_rating THEN
            v_lowest_rating := rec.rating;
        END IF;
    END LOOP;

    -- Calculate the sum, count, highest, and lowest rating from Bikers_on_trail
    FOR rec IN (SELECT rating FROM Bikers_on_trail WHERE trail_id = p_trail_id) LOOP
        v_sum_rating := v_sum_rating + rec.rating;
        v_count_rating := v_count_rating + 1;

        IF rec.rating > v_highest_rating THEN
            v_highest_rating := rec.rating;
        END IF;

        IF rec.rating < v_lowest_rating THEN
            v_lowest_rating := rec.rating;
        END IF;
    END LOOP;
```

```
    -- Calculate the mean rating
    IF v_count_rating > 0 THEN
        v_mean_rating := CEIL(v_sum_rating / v_count_rating); -- Round up to the nearest integer
    ELSE
        v_mean_rating := NULL;
    END IF;

    -- Open a REF CURSOR to return the trail details
    OPEN v_ref_cursor FOR
    SELECT p_trail_id AS trail_id,
           v_mean_rating AS mean_rating,
           v_highest_rating AS highest_rating,
           v_lowest_rating AS lowest_rating
    FROM DUAL;

    RETURN v_ref_cursor;

EXCEPTION
    -- Handle the case where no ratings are found for the given trail_id
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('No data found for the given trail ID.');
        RETURN NULL;
    -- Handle other unexpected errors
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('An unexpected error occurred while calculating the mean rating: ' || SQLERRM);
        RETURN NULL;
END get_trail_mean_rating;
```

דוגמא:

```
1  begin
2    -- Call the function
3    :result := get_trail_mean_rating(p_trail_id => :p_trail_id);
4  end;
```

	Variable	Type	Value
▶	result	Cursor	<Cursor>
	p_trail_id	Integer	20003
*			

	TRAIL_ID	MEAN_RATING	HIGHEST_RATING	LOWEST_RATING
▶ 1	20003	4	5	3

ב. פרוצדורות-

1. תקציר משכורת חודשי עבור עובד.

תיאור: בפרוצדורה זו, אנו מאפשרים לעובדים לקבל תקציר של המשכורת החודשית שלהם. התקציר כולל מידע על המשרות שבהן עבדו, מספר הימים שעבדו והמשכורת הכוללת עבור החודש הנוכחי.

הקוד:

```
CREATE OR REPLACE PROCEDURE print_employee_salaries(employee_in IN Employees.employee_id%TYPE) IS
    -- Declare variables to hold the total salary and total days worked
    v_total_salary NUMBER := 0; -- Initialize total salary
    v_total_days_worked NUMBER := 0; -- Initialize total days worked
    v_first_row BOOLEAN := TRUE; -- Flag to print employee_id only once

 -- Declare a cursor to hold the employee salary data
    CURSOR emp_salaries IS
        SELECT employee_id,
               job,
               COUNT(work_date) AS days_worked,
               CASE job
                   WHEN 'cleaner' THEN COUNT(work_date) * (70 * 9)
                   WHEN 'security guard' THEN COUNT(work_date) * (100 * 9)
                   WHEN 'receptionist' THEN COUNT(work_date) * (40 * 9)
                   ELSE 0 -- Default case for other job types
               END AS salary
        FROM Employee_At_Trail
        WHERE employee_id = employee_in
        AND TO_CHAR(work_date, 'YYYY-MM') = TO_CHAR(SYSDATE, 'YYYY-MM')
        GROUP BY employee_id, job;

BEGIN
    -- Use a FOR loop to fetch rows from the cursor
    FOR emp_rec IN emp_salaries LOOP
        -- Print the employee ID once
        IF v_first_row THEN
            DBMS_OUTPUT.PUT_LINE('Employee ID: ' || emp_rec.employee_id);
            v_first_row := FALSE;
        END IF;
        -- Print the job, days worked, and salary
        DBMS_OUTPUT.PUT_LINE('Job: ' || emp_rec.job ||
                             ', Days Worked: ' || emp_rec.days_worked ||
                             ', Salary: ' || emp_rec.salary);
        -- Accumulate the total salary and total days worked
        v_total_salary := v_total_salary + emp_rec.salary;
        v_total_days_worked := v_total_days_worked + emp_rec.days_worked;
    END LOOP;

    -- Print the total salary and total days worked
    DBMS_OUTPUT.PUT_LINE('Total Salary: ' || v_total_salary);
    DBMS_OUTPUT.PUT_LINE('Total Days Worked: ' || v_total_days_worked);
EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('An unexpected error occurred while retrieving salaries: ' || SQLERRM);
END print_employee_salaries;
```

דוגמא:

```
select * from employee_at_trail where employee_id = 1 AND TO_CHAR(work_date, 'YYYY-MM') = TO_CHAR(SYSDATE, 'YYYY-MM');
```

	EMPLOYEE_ID	TRAIL_ID	WORK_DATE	JOB
1	1	20054	29/07/2024	cleaner
2	1	20078	06/07/2024	receptionist
3	1	20200	23/07/2024	cleaner
4	1	20290	25/07/2024	cleaner
5	1	20326	20/07/2024	security guard
6	1	20608	14/07/2024	security guard
7	1	20619	29/07/2024	security guard
8	1	20638	16/07/2024	cleaner
9	1	20648	21/07/2024	cleaner
10	1	20650	20/07/2024	security guard
11	1	20653	12/07/2024	receptionist
12	1	20722	18/07/2024	security guard
13	1	20839	24/07/2024	receptionist
14	1	20918	27/07/2024	cleaner
15	1	20935	16/07/2024	security guard
16	1	21005	12/07/2024	receptionist
17	1	21012	28/07/2024	receptionist
18	1	21064	19/07/2024	cleaner
19	1	21232	04/07/2024	cleaner

```
begin
  -- Call the procedure
  print_employee_salaries(employee_in => :employee_in);
end;
```

Test script | DBMS Output | Statistics | Profiler | Trace

Clear | Buffer size 10000 | Enabled

```
Employee ID: 1
Job: security guard, Days Worked: 6, Salary: 5400
Job: receptionist, Days Worked: 5, Salary: 1800
Job: cleaner, Days Worked: 8, Salary: 5040
Total Salary: 12240
```

ג. תוכניות ראשיות-

תיאור: התוכנית הראשית מאפשרת למשתמש לבחור בין שתי פונקציות: חישוב דירוג מסלול או הצגת תקציר משכורת של עובד. המשתמש מזין מזהה מסלול או מזהה עובד בהתאם לבחירה. בפונקציה הראשונה, get_trail_mean_rating, מחושב הדירוג הממוצע, הגבוה והנמוך ביותר למסלול המוגדר. בפרוצדורה השנייה, print_employee_salaries, מוצג תקציר של משכורת העובד עבור החודש הנוכחי כולל פרטים על המשרות, הימים שעבד והמשכורת הכוללת.

הקוד:

```
-- Main block to receive user input and call appropriate function/procedure
DECLARE
    -- Variable to hold the user's choice
    v_choice NUMBER;
    -- Variables to hold trail rating details
    v_trail_id trails.trail_id%TYPE;
    v_mean_rating NUMBER;
    v_highest_rating NUMBER;
    v_lowest_rating NUMBER;
    -- Ref Cursor to get results from the function
    v_ref_cursor SYS_REFCURSOR;
    -- Variable to hold employee ID for salary procedure
    v_employee_id Employees.employee_id%TYPE;

BEGIN
    -- Prompt user to choose an option
    DBMS_OUTPUT.PUT_LINE('Enter 1 to get trail mean rating');
    DBMS_OUTPUT.PUT_LINE('Enter 2 to print employee salaries');
    v_choice := &choice;

    -- Process the user's choice
    IF v_choice = 1 THEN
        -- Get trail ID from the user
        v_trail_id := &trail_id;

        -- Call the function to get the trail mean rating
        v_ref_cursor := get_trail_mean_rating(v_trail_id);

        -- Fetch and process the results from the Ref Cursor
        LOOP
            FETCH v_ref_cursor INTO v_trail_id, v_mean_rating, v_highest_rating, v_lowest_rating;
            EXIT WHEN v_ref_cursor%NOTFOUND;

            -- Print the details for verification
            DBMS_OUTPUT.PUT_LINE('Trail ID: ' || v_trail_id);
            DBMS_OUTPUT.PUT_LINE('Mean Rating: ' || v_mean_rating);
            DBMS_OUTPUT.PUT_LINE('Highest Rating: ' || v_highest_rating);
            DBMS_OUTPUT.PUT_LINE('Lowest Rating: ' || v_lowest_rating);
        END LOOP;

        -- Close the Ref Cursor
        CLOSE v_ref_cursor;
```

```
    ELSIF v_choice = 2 THEN
        -- Get employee ID from the user
        v_employee_id := &employee_id;

        -- Call the procedure to print employee salaries
        print_employee_salaries(v_employee_id);

    ELSE
        DBMS_OUTPUT.PUT_LINE('Invalid choice. Please enter 1 or 2.');
    END IF;

EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('An unexpected error occurred: ' || SQLERRM);
END;
```

דוגמא לאופציה הראשונה:

```
Enter 1 to get trail mean rating
Enter 2 to print employee salaries
---------------------------------
Trail ID: 20001
Mean Rating: 4
Highest Rating: 5
Lowest Rating: 1
```

דוגמא לאופציה השנייה:

```
Enter 1 to get trail mean rating
Enter 2 to print employee salaries
---------------------------------
Employee ID: 1
Job: security guard, Days Worked: 6, Salary: 5400
Job: receptionist, Days Worked: 5, Salary: 1800
Job: cleaner, Days Worked: 8, Salary: 5040
Total Salary: 12240
Total Days Worked: 19
```

פונקציה 2:

פונקציה GetEmployeeTrails מחזירה רשומה מסוג SYS_REFCURSOR המציגה את המסלולים שבהם העובד עם המספר מזהה employee_id_in השתתף. היא מחפשת את מספר המסלולים בהם העובד השתתף ומציגה את הנתונים שלהם, או מדפיסה הודעה במקרה של אי-מציאות מסלולים לעובד זה.

קוד:

```
CREATE OR REPLACE FUNCTION GetEmployeeTrails(employee_id_in IN NUMBER)
RETURN SYS_REFCURSOR
IS
    cur_result SYS_REFCURSOR;
    trail_count NUMBER;
BEGIN
    -- Count the number of trails for the given employee
    SELECT COUNT(*)
    INTO trail_count
    FROM Employee_at_trail
    WHERE employee_id = employee_id_in;

    -- Check if there are no trails
    IF trail_count = 0 THEN
        DBMS_OUTPUT.PUT_LINE('No trails found for employee with ID ' || employee_id_in);
        RETURN NULL;
    ELSE
        OPEN cur_result FOR
            SELECT e.employee_name, et.work_date, et.job, t.trail_name, t.location
            FROM Employees e
            JOIN Employee_at_trail et ON e.employee_id = et.employee_id
            JOIN Trails t ON et.trail_id = t.trail_id
            WHERE e.employee_id = employee_id_in;
    END IF;

    RETURN cur_result;
END;
```

דוגמה:

	Variable	Type	Value
▶	✓ result	Cursor	<Cursor>
	✓ employee_id_in	Float	2

```
begin
  -- Call the function
  :result := GetEmployeeTrails(employee_id_in => :employee_id_in);
end;
```

	EMPLOYEE_NAME	WORK_DATE	JOB	TRAIL_NAME	LOCATION
1	Eugene Lythgoe	7/5/2024	security guard	Nahal Amud Trail	Upper Galilee
2	Eugene Lythgoe	7/29/2024	cleaner	Gan HaShlosha National Park	Lower Galilee
3	Eugene Lythgoe	8/6/2024	security guard	Mount Arbel and Wadi Hamam	Lower Galilee
4	Eugene Lythgoe	8/8/2024	cleaner	Nahal Zavitan	Golan Heights
5	Eugene Lythgoe	7/21/2024	receptionist	Mount Bental Trail	Golan Heights
6	Eugene Lythgoe	7/29/2024	cleaner	Nahal Yishuv Trail	Upper Galilee
7	Eugene Lythgoe	8/18/2024	receptionist	Nahal Hatzbani Trail	Upper Galilee
8	Eugene Lythgoe	8/3/2024	receptionist	Nahal Sasa Trail	Western Galilee
9	Eugene Lythgoe	7/28/2024	receptionist	Nahal Tzalmon Trail	Upper Galilee
10	Eugene Lythgoe	7/11/2024	security guard	Nahal Zavitan Trail	Upper Galilee
11	Eugene Lythgoe	7/10/2024	receptionist	Nahal Hashofet Trail	Lower Galilee
12	Eugene Lythgoe	7/11/2024	cleaner	Nahal Hilazon Trail	Lower Galilee
13	Eugene Lythgoe	8/22/2024	cleaner	Nahal Kadesh Trail	Lower Galilee
14	Eugene Lythgoe	8/4/2024	receptionist	Nahal Keshet Trail	Western Galilee
15	Eugene Lythgoe	7/26/2024	receptionist	Mount Tabor Summit Trail	Mount Tabor
16	Eugene Lythgoe	8/8/2024	cleaner	En Gedi Nature Reserve Trail	Arad

פרוצדורה 2:

חישוב סוכם כולל ששילם מטייל.

הפרוצדורה TotalPayment מחשבת את סכום התשלום הכולל של לקוח עבור כל המסלולים שהוא השתתף בהם. היא מקבלת כקלט את מזהה הלקוח (`traveler_id_in`) ומדפיסה את הסכום הכולל שבוצע לפי כמות המסלולים ומחירם. אם הלקוח לא השתתף בשום מסלול.

הקוד:

```
CREATE OR REPLACE NONEDITIONABLE PROCEDURE TotalPayment(traveler_id_in IN NUMBER)
IS
    CURSOR trail_prices IS
        SELECT t.price, c.number_of_travelers
        FROM Customers c
        JOIN Bikers_on_trail bot ON c.traveler_id = bot.traveler_id
        JOIN Trails t ON bot.trail_id = t.trail_id
        WHERE c.traveler_id = traveler_id_in
        UNION ALL
        SELECT t.price, c.number_of_travelers
        FROM Customers c
        JOIN Hikers_on_trail hot ON c.traveler_id = hot.traveler_id
        JOIN Trails t ON hot.trail_id = t.trail_id
        WHERE c.traveler_id = traveler_id_in;

    total_payment NUMBER := 0;
    trail_rec trail_prices%ROWTYPE;
    trail_count NUMBER := 0;  -- Counter for the number of trails the traveler has participated in
    customer_name VARCHAR2(100);  -- Variable to hold the customer's name
    header_printed BOOLEAN := FALSE;  -- Flag to print the header only once

BEGIN
    -- Fetch the customer's name
    SELECT c.traveler_name
    INTO customer_name
    FROM Customers c
    WHERE c.traveler_id = traveler_id_in;
```

```
    -- Count the number of trails the traveler has participated in
    SELECT COUNT(*)
    INTO trail_count
    FROM (
        SELECT bot.traveler_id
        FROM Bikers_on_trail bot
        WHERE bot.traveler_id = traveler_id_in
        UNION ALL
        SELECT hot.traveler_id
        FROM Hikers_on_trail hot
        WHERE hot.traveler_id = traveler_id_in
    );

    IF trail_count = 0 THEN
        DBMS_OUTPUT.PUT_LINE('Customer with traveler_id ' || traveler_id_in || ' hasn''t gone on any trails.');
        RETURN;
    END IF;

    -- If there are trails, calculate total payment
    OPEN trail_prices;
    LOOP
        FETCH trail_prices INTO trail_rec;
        EXIT WHEN trail_prices%NOTFOUND;

        -- Print the payment summary header only once
        IF NOT header_printed THEN
            DBMS_OUTPUT.PUT_LINE('Payment Summary for ' || customer_name );
```

```
        -- Print the payment summary header only once
        IF NOT header_printed THEN
            DBMS_OUTPUT.PUT_LINE('Payment Summary for ' || customer_name );
            DBMS_OUTPUT.PUT_LINE(' ');
            header_printed := TRUE;
        END IF;

        -- Calculate and print each trail's payment
        DBMS_OUTPUT.PUT_LINE('Trail price: ' || trail_rec.price || ', Number of travelers: ' || trail_rec.number_of_travelers ||',
         Payment for this trail: ' || (trail_rec.price * trail_rec.number_of_travelers) || ' shekel');
        total_payment := total_payment + (trail_rec.price * trail_rec.number_of_travelers);
    END LOOP;
    CLOSE trail_prices;

    -- Print the total payment
    DBMS_OUTPUT.PUT_LINE(' ');
    DBMS_OUTPUT.PUT_LINE('Total payment is ' || total_payment || ' shekel');

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('No data found for traveler_id ' || traveler_id_in);
    WHEN OTHERS THEN
        RAISE_APPLICATION_ERROR(-20001, 'An error occurred: ' || SQLERRM);
END;
```

דוגמה:

```
1  begin
2     -- Call the procedure
3     TotalPayment(traveler_id_in => :traveler_id_in);
4  end;
```

	Variable	Type	Value
▶	traveler_id_in	Float	301473
*			

```
Payment Summary for Christina Beckham

Trail price: 50, Number of travelers: 94, Payment for this trail: 4700 shekel

Total payment is 4700 shekel
```

תוכנית ראשית 2:
התוכנית הראשית מאפשר למשתמש לבחור בין שתי אפשרויות:

1. בחירה 1: קבלת מסלולים של עובד על פי מספר זיהוי שלו.
2. בחירה 2: חישוב תשלום כולל ששולם עבור מסלולים שהמטייל השתתף בהם על פי מספר זיהוי של המטייל.

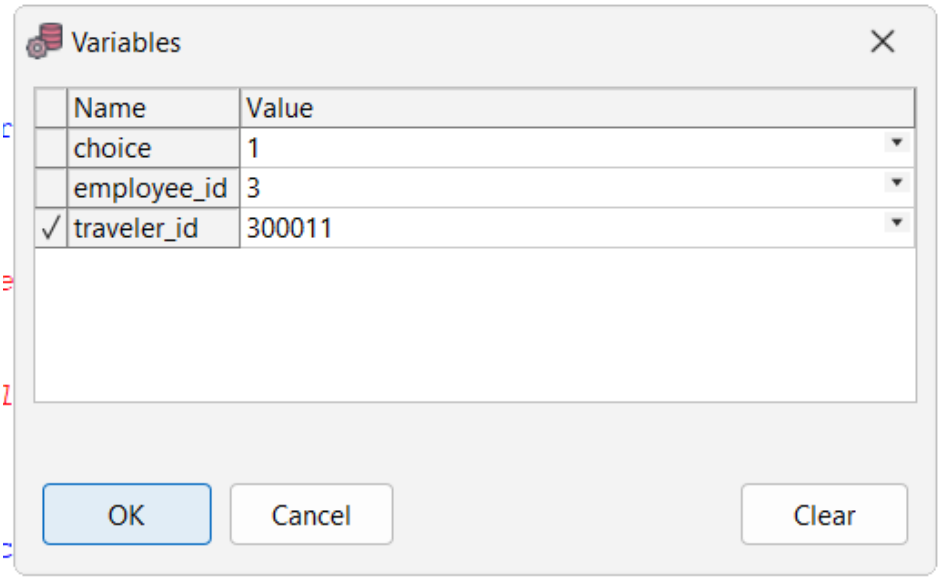
אם המשתמש יבחר באפשרות לא תקפה, התוכנית תדפיס הודעת שגיאה ותפסיק את הריצה שלה.

```
-- Main block to receive user input and call appropriate function/procedure
DECLARE
    -- Variable to hold the user's choice
    v_choice NUMBER;
    -- Ref Cursor to get results from the function
    v_ref_cursor SYS_REFCURSOR;
    -- Variables to hold employee ID and traveler ID
    v_employee_id Employees.employee_id%TYPE;
    v_traveler_id Customers.traveler_id%TYPE;
    -- Variables to hold the fetched data from the cursor
    v_employee_name Employees.employee_name%TYPE;
    v_work_date Employee_at_trail.work_date%TYPE;
    v_job Employee_at_trail.job%TYPE;
    v_trail_name Trails.trail_name%TYPE;
    v_location Trails.location%TYPE;
    -- Flag to check if the employee name has been printed
    v_name_printed BOOLEAN := FALSE;

BEGIN
    -- Prompt user to choose an option
    DBMS_OUTPUT.PUT_LINE('Enter 1 to get employee trails');
    DBMS_OUTPUT.PUT_LINE('Enter 2 to calculate total payment for a traveler');
    DBMS_OUTPUT.PUT_LINE('-----------------------------------');
    v_choice := &choice;

    -- Process the user's choice
    IF v_choice = 1 THEN
        -- Get employee ID from the user
        v_employee_id := &employee_id;
```

```
        -- Call the function to get the employee trails
        v_ref_cursor := GetEmployeeTrails(v_employee_id);

        -- Check if the cursor is null (meaning no trails found)
        IF v_ref_cursor IS NOT NULL THEN
            -- Fetch and process the results from the Ref Cursor
            LOOP
                FETCH v_ref_cursor INTO v_employee_name, v_work_date, v_job, v_trail_name, v_location;
                EXIT WHEN v_ref_cursor%NOTFOUND;

                -- Print the employee name only once
                IF NOT v_name_printed THEN
                    DBMS_OUTPUT.PUT_LINE('Employee Name: ' || v_employee_name);
                    v_name_printed := TRUE;
                END IF;

                -- Print the trail details
                DBMS_OUTPUT.PUT_LINE('Work Date: ' || v_work_date);
                DBMS_OUTPUT.PUT_LINE('Job: ' || v_job);
                DBMS_OUTPUT.PUT_LINE('Trail Name: ' || v_trail_name);
                DBMS_OUTPUT.PUT_LINE('Location: ' || v_location);
            END LOOP;

            -- Close the Ref Cursor
            CLOSE v_ref_cursor;
        ELSE
            DBMS_OUTPUT.PUT_LINE('No trails found for employee with ID ' || v_employee_id);
        END IF;

|
    ELSIF v_choice = 2 THEN
        -- Get traveler ID from the user
        v_traveler_id := &traveler_id;

        -- Call the procedure to calculate total payment for the traveler
        TotalPayment(v_traveler_id);

    ELSE
        DBMS_OUTPUT.PUT_LINE('Invalid choice. Please enter 1 or 2.');
    END IF;

EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('An unexpected error occurred: ' || SQLERRM);
END;
```

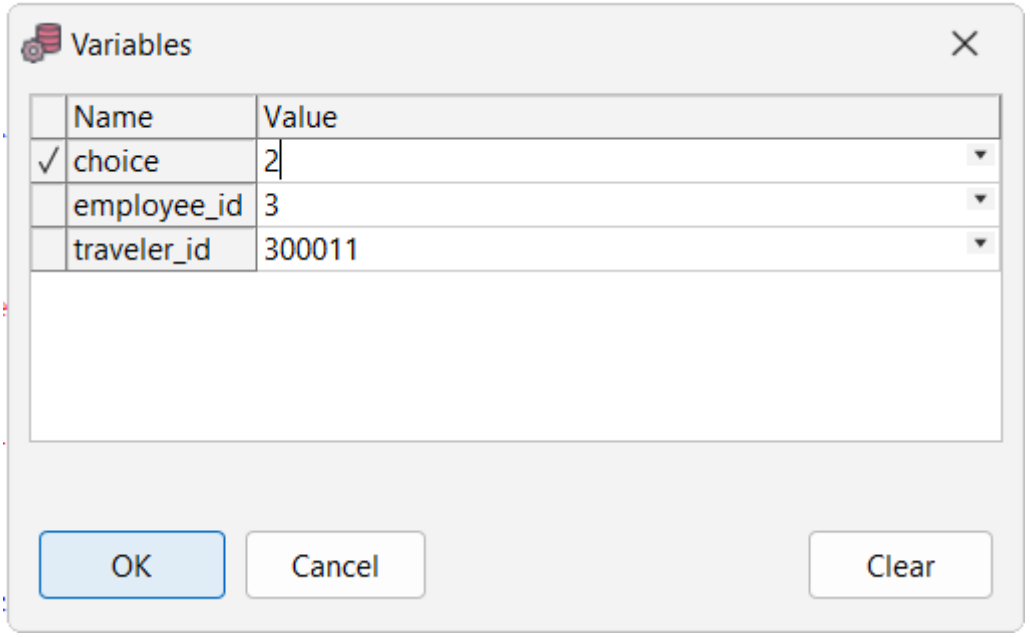
הרצה:

לחיצה על 1:




```
Enter 1 to get employee trails
Enter 2 to calculate total payment for a traveler
---------------------------------
Employee Name: Meridel Yackiminie
Work Date: 01-JUL-24
Job: security guard
Trail Name: Yam L'Yam Trail
Location: Upper Galilee
Work Date: 06-JUL-24
Job: receptionist
Trail Name: Tavor Stream Trail
Location: Lower Galilee
Work Date: 26-AUG-24
Job: cleaner
Trail Name: Ye'elim Stream
Location: Negev Desert
Work Date: 23-AUG-24
Job: cleaner
Trail Name: Nahal Zavitan Trail
Location: Golan Heights
Work Date: 14-AUG-24
Job: security guard
Trail Name: Nahal Amud Trail
Location: Upper Galilee
Work Date: 03-OCT-20
Job: security guard
Trail Name: Nahal El Al Trail
```

לחיצה על 2:




```
Enter 1 to get employee trails
Enter 2 to calculate total payment for a traveler
---------------------------------
Payment Summary for Noah Madonna

Trail price: 30, Number of travelers: 79, Payment for this trail: 2370 shekel
Trail price: 25, Number of travelers: 79, Payment for this trail: 1975 shekel

Total payment is 4345 shekel
```