

מINI פרויקט בבסיסי נתונים – תרגיל 4

ראשית, התחלנו ביצירת נתונים לטבלה הראשונה שלנו – profile.
לצורך זה התמשכנו באתר [mockaroo](https://mockaroo.com).

The screenshot shows the Mockaroo web application's schema configuration page. At the top, there are tabs for SCHEMAS, DATASETS, MOCK APIs, SCENARIOS, and PROJECTS. Below the tabs, a message encourages generating up to 1,000 rows of realistic test data in CSV, JSON, SQL, and Excel formats. The main area displays a table of fields and their configurations:

Field Name	Type	Options
profile_id	Sequence	start at: 9001 step: 1 repeat: 1 restart at: blank: 0 % Σ X
first_name	First Name	blank: 0 % Σ X
last_name	Last Name	blank: 0 % Σ X
email	Email Address	blank: 0 % Σ X
gender	Gender (abbrev)	blank: 0 % Σ X
pass	Password	blank: 0 % Σ X

Below the table, there are buttons for "ADD ANOTHER FIELD", "# Rows: 10000", "Format: SQL", "Table Name: profile", and "include CREATE TABLE". An "Append Dataset:" dropdown is also present. At the bottom, there are links for "Follow @mockaroodev", "DOWNLOAD DATA", "PREVIEW", "SAVE THIS SCHEMA", and "MORE ▾".

יצרנו 10,000 רשומות בקובץ csv ועשינו לקובץ import:

The screenshot shows a Microsoft Excel spreadsheet with data from the 'profile' schema. The columns are labeled PROFILE_ID, FIRST_NAME, LAST_NAME, EMAIL, PASS, and GENDER. The data consists of 24 rows of sample data, such as Terrence Gilley, Richard Alston, Leo Sweeney, Taye Klugh, Kenneth Evanswood, Lorraine Callow, Bridgette Fisher, and many others.

	PROFILE_ID	FIRST_NAME	LAST_NAME	EMAIL	PASS	GENDER
1	6997	Terrence	Gilley	terrence.gilley@yumbrands.com	887880404	F
2	6998	Richard	Alston	richard.a@httpprint.ca	229526463	O
3	6999	Leo	Sweeney	leo.sweeney@provideservice.com	707536111	M
4	7000	Taye	Klugh	tklugh@directdata.uk	728477299	F
5	7001	Kenneth	Evanswood	kenneth.evanswood@apprius.com	828246673	F
6	7002	Lorraine	Callow	lorraine.callow@pinnaclestaffing.lt	985576344	M
7	7003	Bridgette	Fisher	bridgette.f@healthscribe.com	601869273	F
8	6881	Gin	Calle	gin.calle@ptg.com	206373139	O
9	6882	Chalee	Belle	chalee.belle@northhighland.br	735106242	M
10	6883	Hazel	Kotto	hazel.kotto@visionarysystems.ch	529107382	F
11	6884	First	Glover	first.g@refinery.com	468508279	O
12	6885	Jodie	Rhymes	jodie.r@lms.com	786589939	M
13	6886	Marie	Griggs	mrigggs@electricalsolutions.com	805261409	F
14	6887	Maggie	Sevigny	msevigny@ibfh.br	587230922	O
15	6888	Regina	Thorton	regina@ccb.com	491590791	O
16	6889	Wesley	Ledger	w.ledger@kingland.be	255143046	F
17	6890	Armand	Dukakis	armand@providentbancorp.com	189400825	M
18	6891	Kyle	Root	k.root@johnkeeler.com	824168823	M
19	6892	Chely	Dourif	chelyd@qestrel.com	217946050	M
20	6893	Mary	Spector	mary.spector@venoco.be	471738661	M
21	6894	Angela	Moody	angela.tarna.com	937700658	M
22	6895	Harriet	Kelly	harriet.kelly@kitba.nl	889654542	F
23	6896	Rosie	Atlas	rosie.atlas@lempproducts.it	317430747	O
24	6897	Jimmy	Starr	jimmy.starr@integratedamerica.nl	432663749	F

לאחר מכן יצרנו את הטבלה `post_type` והוספנו לה שני סוגי פוטיטים, רגיל וקבוצה.

	POST_TYPE_CODE	POST_TYPE_NAME	
▶	1	NORMAL	...
	2	GROUP	...

ואז כתבנו סקריפט בפייתון כדי ליצור פקודות `insert data` ולמלא את הטבלה `.post`.

```

for x in f:
    c += 1
    x = x.replace("date", "post_date")
    x = x.replace("content", "post_content")
    counter += 1
    count += 1
    if count % 1001 == 0:
        count = 1
    y = x
    profile_id = random.randint(1, 10000)
    arr[profile_id] += 1
    post_id = arr[profile_id]
    visibility = random.randint(0 , 1)
    likes = random.randint(0, 2000)
    type = random.randint(0,1)
    group = 'NULL'
    if type == 0:
        group = random.randint(1, 500)
    lst = y.split("values")
    prop = lst[1].split(',')
    post_id_new = prop[0].replace(str(counter), str(post_id))
    profile_id_new = prop[1].replace(str(count), str(profile_id))
    date_new = change_format_date(prop[2].split('')[1])
    visibility_new = prop[3].replace(str(count), str(visibility))
    likes_new = prop[4].replace(str(count), str(likes))
    type_new = prop[6].replace(str(count), str(type))
    group_new = prop[7].replace(str(count), str(group))
    y = "values".join([lst[0],'.join([post_id_new, profile_id_new, date_new, visibility_new, likes_new, prop[5], type_new, group_new])])
    p.write(y)
    p.write(y)

```

יצרנו 10,000 פקודות `insert data` והרכינו בSQL.

	POST_ID	PROFILE_ID	POST_DATE	VISIBILITY_CODE	NUMBER_OF_LIKES	POST_CONTENT	POST_TYPE_CODE	GROUP_ID
▶	1	2462	10/05/2020	0	563	vehicula condimentum	...	1
2	1	176	15/04/2020	1	1587	metus vitae ipsum aliquam	...	0
3	1	8102	01/12/2020	1	47	eu interdum eu tincidunt in	...	1
4	1	1649	26/01/2021	1	1026	in hac	...	1
5	1	6786	17/10/2020	1	911	habitasse plates	...	1
6	1	5831	08/03/2021	1	1068	etiam pretium iaculis justo in	...	0
7	1	9330	26/01/2021	1	543	sed lacus	...	0
8	1	5146	27/06/2020	0	1242	arcu libero rutrum ac lobortis	...	0
9	1	2729	01/08/2020	1	1870	mauris vulputate elementum nullam varius	...	0
10	1	2001	06/03/2021	0	826	ipsum aliquam non mauris morbi	...	0
11	1	3169	04/05/2020	1	1652	tempus sit amet	...	1
12	1	9326	29/04/2020	0	1237	muis etiam	...	1
13	1	734	21/05/2020	0	865	nulla nisl	...	0
14	1	9445	21/05/2020	1	1544	ultricies eu nibh quisque	...	1
15	1	7248	30/10/2020	0	1454	lacinia sapien quis libero nullam	...	1
16	1	9143	31/03/2021	0	1599	diam vitae	...	1
17	1	1818	29/03/2021	1	1792	prin leo odio porttitor id	...	1
18	1	9247	19/03/2021	1	1533	nisi et nibh in	...	0
19	1	6615	23/09/2020	0	744	magna bibendum imperdiet nullam orci	...	1
20	2	8253	26/12/2020	1	813	porta volutpat erat quisque	...	0
21	1	885	11/05/2020	0	1194	donec quis orci	...	1
22	1	9232	29/07/2020	1	913	sapien cum	...	0
23	1	3812	23/12/2020	1	1087	Faucibus orci luctus et ultrices	...	0
24	1	9480	08/07/2020	0	1800	dolor sit amet consectetur adipiscing	...	1

בסוף השתמשנו ב- **Data generator** של SqSAd כדי למלא את ה таблицה **.post_comment**

יצרנו אותו 10,000 רשומות עבור ה таблицה.

	COMMENT_ID	POST_ID	PROFILE_ID	COMMENT_CONTENT
▶ 1	1	1	1	1125 Morbi porttitor lorem id ligula. Suspendisse ornare consequat lectus. In e...
2	1	1	1	8901 Mauris sit amet eros. Suspendisse accumsan tortor quis turpis. Sed ante. ...
3	1	1	1	7938 Vivamus metus arcu, adipiscing molestie, hendrerit at, vulputate vitae, ni...
4	1	2	2836 Aenean auctor gravida sem. Praesent id massa id nisl venenatis lacinia. A...	
5	1	1	5473 Nulla ac enim. In tempor, turpis nec euismod scelerisque, quam turpis ac...	
6	1	1	9705 Maecenas tristique, est et tempus semper, est quam pharetra magna, ac ...	
7	1	1	9207 In hac habitasse platea dictumst. Etiam faucibus cursus urna. Ut tellus. N...	
8	1	1	5184 Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere i...	
9	1	1	8969 Vivamus metus arcu, adipiscing molestie, hendrerit at, vulputate vitae, ni...	
10	1	1	1167 Vivamus metus arcu, adipiscing molestie, hendrerit at, vulputate vitae, ni...	
11	1	1	6488 Proin interdum mauris non ligula pellentesque ultrices. Phasellus id sapi...	
12	1	1	5313 Nunc rhoncus dui vel sem. Sed sagittis. Nam congue, risus semper porta ...	
13	1	1	9146 Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ...	
14	1	1	232 Donec ut dolor. Morbi vel lectus in quam fringilla rhoncus. Mauris enim ...	
15	1	1	8985 Nunc rhoncus dui vel sem. Sed sagittis. Nam congue, risus semper porta ...	
16	1	1	9001 Suspendisse potenti. Nullam porttitor lacus at turpis. Donec posuere met...	
17	1	1	1402 Phasellus id sapien in sapien iaculis congue. ...	
18	1	1	4649 Aenean lectus. Pellentesque eget nunc. Donec quis orci eget orci vehicul...	
19	1	1	4245 Phasellus sit amet erat. Nulla tempus. Vivamus in felis eu sapien cursus v...	
20	1	1	8210 Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ...	
21	1	1	3378 Nulla ut erat id mauris vulputate elementum. Nullam varius. Nulla facilis ...	
22	1	1	5689 Proin eu mi. Nulla ac enim. In tempor, turpis nec euismod scelerisque, qu...	
23	1	1	9944 In blandit ultrices enim. Lorem ipsum dolor sit amet, consectetur adipis...	
24	1	1	8679 In sagittis dui vel nisl. Duis ac nibh. Fusce lacus purus, aliquet at, feugiat ...	