

# מחלקת כספים ביקב

אסתר הדס וקנין 324966993

תהילה ישראלי 325119493

תוכן עניינים

- 3----- מבוא
- 4----- תרשים ERD מחלקת כספים ביקב —
- 5----- תרשים DSD מחלקת כספים ביקב—
- 6----- החלטות עיצוב ונימוקים—
- 7----- צילומי מסך של שלושת שיטות הכנסת הנתונים שנבחרו—

## מבוא:

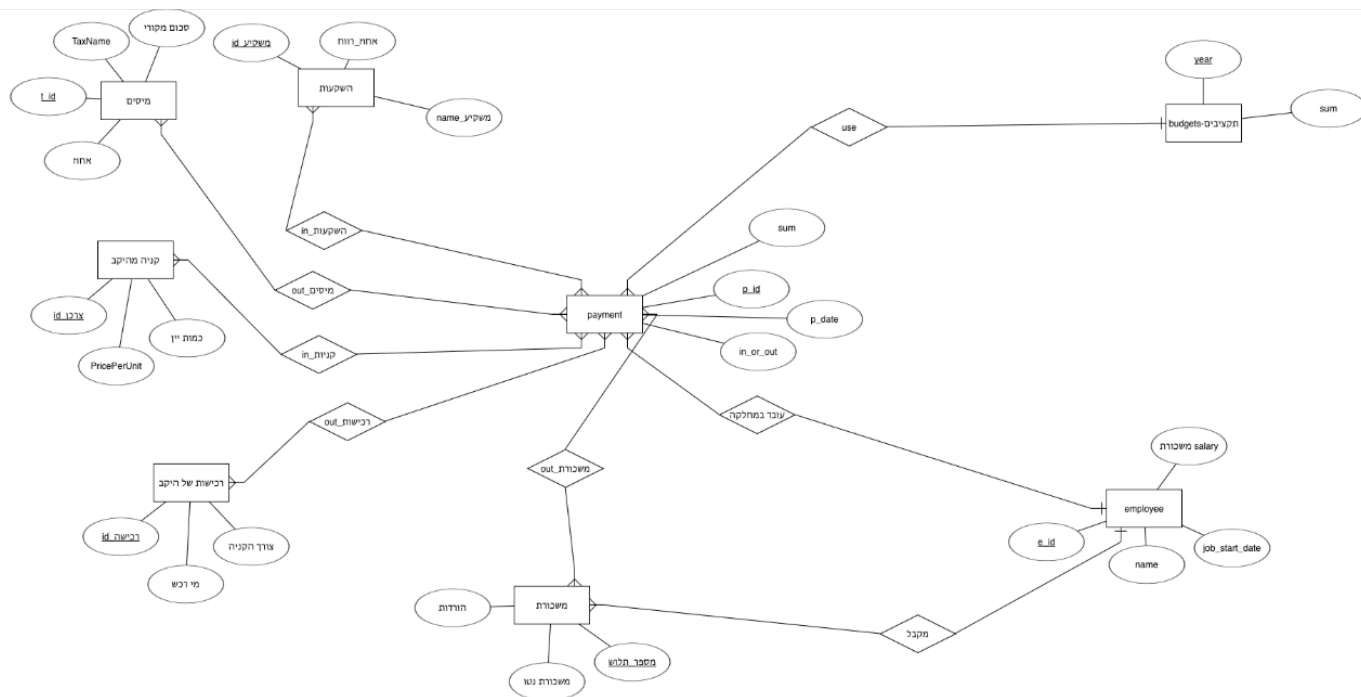
### תיאור הנתונים הנשמרים במערכת:

1. **טבלת תשלומים (payment)** - כוללת מזהה תשלום, סכום, תאריך, וסיווג האם מדובר בהוצאה או הכנסה.
2. **עובדים (employee)** - לכל עובד יש מזהה, שם, תאריך התחלת עבודה ושכר.
3. **תקציבים (budgets-תקציבים)** - כוללים שנת תקציב וסכום מוקצה.
4. **השקעות (השקעות)** - מקשרות בין גורם משקיע (משקיע) לסוגי השקעות, עם סכומי השקעה ותשואה.
5. **רכישות ורכיבי קנייה** - מעקב אחר רכישות, כולל מידע על הצורך ברכישה, מי רכש, וכמות.
6. **מיסים (מיסים)** - כללים על מיסים שונים, סכום מס, ואחוזים.
7. **משכורות (משכורות)** - כוללות משכורת נטו, מס הכנסה וכו'.

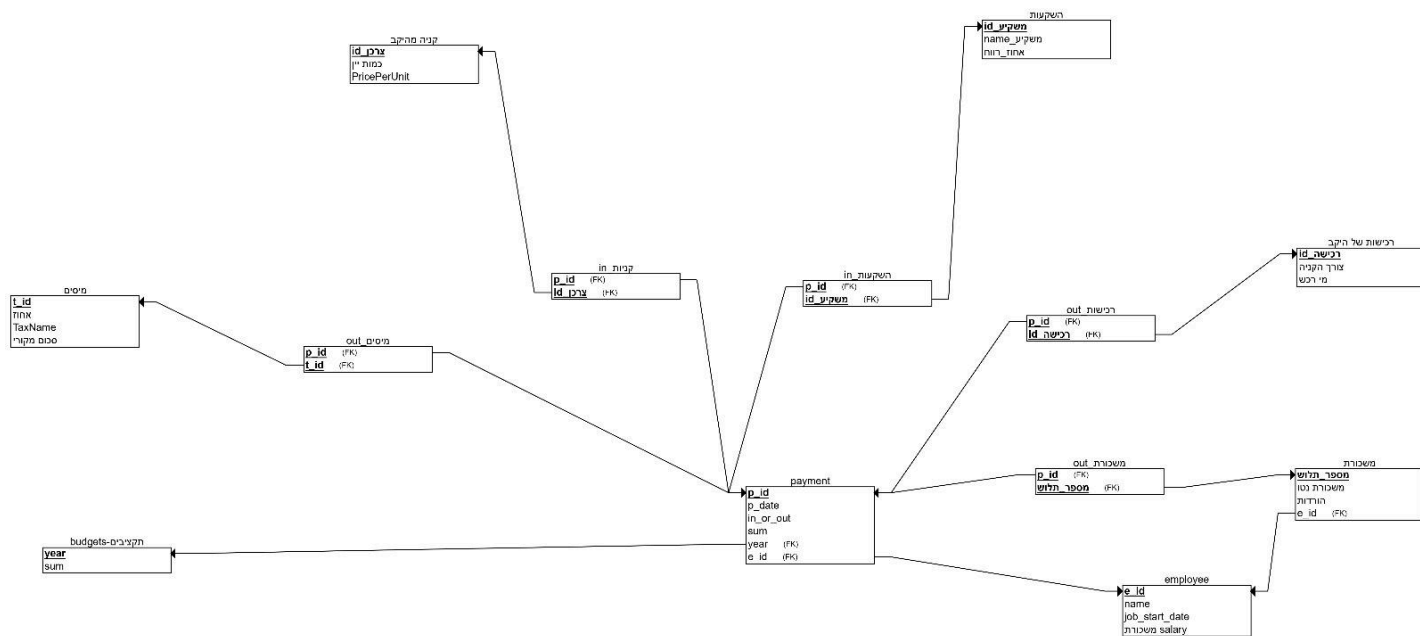
### הפונקציונליות העיקרית במערכת:

- מעקב אחר כל תשלום כולל קישור להשקעות, משכורות ורכישות.
- חישוב משכורות נטו לעובדים לאחר ניכוי מסים.
- ניהול תקציבים ובדיקת הוצאות מול הכנסות.
- מעקב אחר רכישות - מי רכש, מה נקנה, וכמות.
- חישוב מסים לפי חוקים והגדרות מותאמות.

## תרשים ERD מחלקת כספים ביקב



## תרשים DSD מחלקת כספים ביקב



ב"ה

### החלטות עיצוב ונימוקים:

1. **ריכוז התשלומים בישות אחת (payment)** - כל התשלומים במערכת (משכורות, רכישות, השקעות) מנוהלים דרך ישות מרכזית אחת. זה מאפשר גישה אחידה לכל סוגי ההוצאות וההכנסות.
2. **קישור ישיר בין תשלומים לרכישות, משכורות והשקעות** - לכל תשלום ניתן לשייך את מקורו (לדוגמה: רכישה, משכורת, השקעה), מה שמאפשר מעקב ברור אחר סוגי התנועות הכספיות.
3. **פיצול ישות הרכישה** - הרכישות מחולקות לרכישה כללית ורכישת רכיב ספציפי, כך שניתן לפרט כל רכישה לרמת הפריטים שנקנו, הכמות והמחיר ליחידה.
4. **ניהול משכורות בנפרד עם חישובי מס** - הפרדת המשכורות מאפשרת לחשב משכורת נטו על בסיס מס הכנסה וחוקים שונים.
5. **שימוש בישות תקציב (budgets-תקציבים)** - מאפשר מעקב אחר הגבלת הוצאות והשוואה מול הסכומים המתוקצבים לכל שנה.
6. **ניהול מסים בנפרד** - במקום לשמור אחוזי מס בתוך תשלומים, ישות נפרדת למסים מאפשרת לעדכן חוקים משתנים ללא שינוי מבנה בסיס הנתונים.

ב"ה

צילומי מסך של שלושת שיטות הכנסת הנתונים שנבחרו.

## 1. mackoroo

Import/Export data - table 'budgets'

General

Options

Columns

Import/Export

✓ Import

Export

Filename

C:\Users\tehil\Desktop\בבסנת\בפרויקט\ג\סמסטר\תהילה\שנה\לימודים\mac

Format

csv

Encoding

UTF8

i

?

Close

Reset

✓ OK

Import/Export data - table 'investments'

General

Options

Columns

Import/Export

✓ Import

Export

Filename

C:\Users\tehil\Desktop\בבסנת\בפרויקט\ג\סמסטר\תהילה\שנה\לימודים\mac

Format

csv

Encoding

UTF8

i

?

Close

Reset

✓ OK

ב"ה

## 2. GENERATEDATA:

```
DROP TABLE IF EXISTS "Purchase_from_the_winery";

CREATE TABLE "Purchase_from_the_winery" (
  id_Consumer integer NULL,
  Quantity_of_wine integer NULL,
  PricePerUnit integer NULL
);

INSERT INTO Purchase_from_the_winery (id_Consumer,Quantity_of_wine ,PricePerUnit)
VALUES
  (1,16,101) .
```

Query	Query History
1	DROP TABLE IF EXISTS "taxes";
2	
3	CREATE TABLE "taxes" (
4	t_id integer NULL,
5	percent integer NULL,
6	TaxName TEXT default NULL,
7	principal_amount integer NULL
8	);
9	
10	INSERT INTO taxes (t_id,percent ,TaxName,principal_amount)
11	VALUES

ב"ה

## 3. script in python



colab.research.google.com

Untitled0.ipynb - Colab

File Edit View Insert Runtime Tools Help

Commands + Code + Text

```
import pandas as pd
import random
from datetime import datetime, timedelta

# פונקציה ליצירת תאריך אקראי בטווח נתון
def random_date(start_date, end_date):
    delta_days = (end_date - start_date).days
    random_days = random.randint(0, delta_days)
    return start_date + timedelta(days=random_days)

# מספר עובדים
NUM_EMPLOYEES = 400
NUM_PAYMENTS = 400
NUM_SALARIES = 400
NUM_PURCHASES_FROM = 400
NUM_PURCHASES_FOR = 400

# רשימת שמות לדוגמה
first_names = ['John', 'Jane', 'Alex', 'Chris', 'Mary', 'Jessica', 'Michael', 'David', 'Sarah', 'Daniel']
last_names = ['Smith', 'Johnson', 'Williams', 'Brown', 'Jones', 'Garcia', 'Miller', 'Davis', 'Rodriguez', 'Martinez']

# פונקציה ליצירת שם מלא
def generate_full_name():
    first_name = random.choice(first_names)
    last_name = random.choice(last_names)
    return f'{first_name} {last_name}'

# יצירת טבלת employee
employee_data = []
employee_ids = list(range(100, 100 + NUM_EMPLOYEES))
for e_id in employee_ids:
    employee_data.append({
        'e_id': e_id,
        'name': generate_full_name(),
        'job_start_date': random_date(datetime(2010, 1, 1), datetime(2020, 1, 1)),
        'last_salary': random.randint(7000, 25000)
```

Allocating runtime

The image shows a screenshot of the pgAdmin 4 web interface. On the left, the 'Object Explorer' pane shows a tree view of database objects. Under 'Tables (13)', the table 'in\_purchases\_from' is selected. The main pane displays the 'Import/Export data - table 'in\_purchases\_from'' dialog box. The 'General' tab is active, showing 'Import' selected under 'Import/Export', the filename '/Users/yhd/Downloads/script in python/in\_Purchases\_from-4.csv', the format 'csv', and the encoding 'Select an item...'. Below the dialog, a table view of 'in\_purchases\_from' is shown with columns: p\_id [PK] numeric (3), p\_date date, in\_or\_out character varying (3), p\_sum integer, p\_year integer, and e\_id numeric (9). The table contains 10 rows of data. The status bar at the bottom indicates 'Total rows: 400' and 'Query complete 00:00:00.107'. The macOS dock is visible at the very bottom.

pgAdmin 4

finnance/postgres@PostgreSQL 17\*

Import/Export data - table 'employee'

General Options Columns

Import/Export ☒ Import Export

Filename

Format

Encoding

Data Output

Showing rows: 1 to 400 Page No: 1 of 1

	p_id [PK] numeric (3)	p_date date	in_or_out character varying (3)	p_sum integer	p_year integer	e_id numeric (9)
1	200	2020-01-24	out	46170	2020	486
2	201	2024-05-04	out	45076	2024	128
3	202	2021-07-02	out	25351	2021	407
4	203	2024-06-19	in	23396	2024	497
5	204	2023-12-15	in	11036	2023	220
6	205	2023-04-22	in	10535	2023	242
7	206	2022-06-25	in	29701	2022	497
8	207	2022-11-04	in	22146	2022	135
9	208	2024-11-18	out	33344	2024	134
10	209	2023-02-24	in	47312	2023	191

Total rows: 400 Query complete 00:00:00.107

pgAdmin 4

finnance/postgres@PostgreSQL 17\*

Import/Export data - table 'out\_salary'

General Options Columns

Import/Export ☒ Import Export

Filename

Format

Encoding

Data Output

Showing rows: 1 to 400 Page No: 1 of 1

	p_id [PK] numeric (3)	p_date date	in_or_out character varying (3)	p_sum integer	p_year integer	e_id numeric (9)
1	200	2020-01-24	out	46170	2020	486
2	201	2024-05-04	out	45076	2024	128
3	202	2021-07-02	out	25351	2021	407
4	203	2024-06-19	in	23396	2024	497
5	204	2023-12-15	in	11036	2023	220
6	205	2023-04-22	in	10535	2023	242
7	206	2022-06-25	in	29701	2022	497
8	207	2022-11-04	in	22146	2022	135
9	208	2024-11-18	out	33344	2024	134
10	209	2023-02-24	in	47312	2023	191

Total rows: 400 Query complete 00:00:00.107

pgAdmin 4

Object Explorer

- FTS Configurations
- FTS Dictionaries
- FTS Parsers
- FTS Templates
- Foreign Tables
- Functions
- Materialized Views
- Operators
- Procedures
- Sequences
- Tables (13)
  - budgets
  - employee
  - in\_investments
  - in\_purchases\_from
  - investments
  - out\_purchase\_for\_
  - out\_salary
  - out\_taxes
  - payment
  - purchase\_from\_the
  - salary
  - taxes
  - winery\_purchases
- Trigger Functions
- Types
- Views
- Subscriptions
- postgres
  - Casts
  - Catalogs
  - Event Triggers

Query Editor

Import/Export data - table 'salary'

General Options Columns

Import/Export ☒ Import Export

Filename: /Users/yhd/Downloads/script in python/salary-4.csv

Format: csv

Encoding: Select an item...

Data Output

Showing rows: 1 to 400 Page No: 1 of 1

p_id [PK] numeric (3)	p_date date	in_or_out character varying (3)	p_sum integer	p_year integer	e_id numeric (9)	
1	200	2020-01-24	out	46170	2020	486
2	201	2024-05-04	out	45076	2024	128
3	202	2021-07-02	out	25351	2021	407
4	203	2024-06-19	in	23396	2024	497
5	204	2023-12-15	in	11036	2023	220
6	205	2023-04-22	in	10535	2023	242
7	206	2022-06-25	in	29701	2022	497
8	207	2022-11-04	in	22146	2022	135
9	208	2024-11-18	out	33344	2024	134
10	209	2023-02-24	in	47312	2023	191

Total rows: 400 Query complete 00:00:00.107

pgAdmin 4

Object Explorer

- FTS Configurations
- FTS Dictionaries
- FTS Parsers
- FTS Templates
- Foreign Tables
- Functions
- Materialized Views
- Operators
- Procedures
- Sequences
- Tables (13)
  - budgets
  - employee
  - in\_investments
  - in\_purchases\_from
  - investments
  - out\_purchase\_for\_
  - out\_salary
  - out\_taxes
  - payment
  - purchase\_from\_the
  - salary
  - taxes
  - winery\_purchases
- Trigger Functions
- Types
- Views
- Subscriptions
- postgres
  - Casts
  - Catalogs
  - Event Triggers

Query Editor

Import/Export data - table 'payment'

General Options Columns

Import/Export ☒ Import Export

Filename: /Users/yhd/Downloads/script in python/payment-4.csv

Format: csv

Encoding: Select an item...

Data Output

Showing rows: 1 to 400 Page No: 1 of 1

p_id [PK] numeric (3)	p_date date	in_or_out character varying (3)	p_sum integer	p_year integer	e_id numeric (9)	
162	361	2022-07-11	in	18533	2022	363
163	362	2023-06-12	in	23889	2023	495
164	363	2021-09-27	out	42792	2021	418
165	364	2021-03-02	out	37069	2021	455
166	365	2024-06-24	in	38836	2024	267
167	366	2024-12-27	in	12194	2024	255
168	367	2024-06-18	out	35001	2024	292
169	368	2020-08-23	out	41686	2020	292
170	369	2021-10-03	in	7363	2021	210

Total rows: 400 Query complete 00:00:00.148