345537708 Samuel Tapiro - 133467 Lior Tordjman - 346012065 Daniel Elbaz

Step A

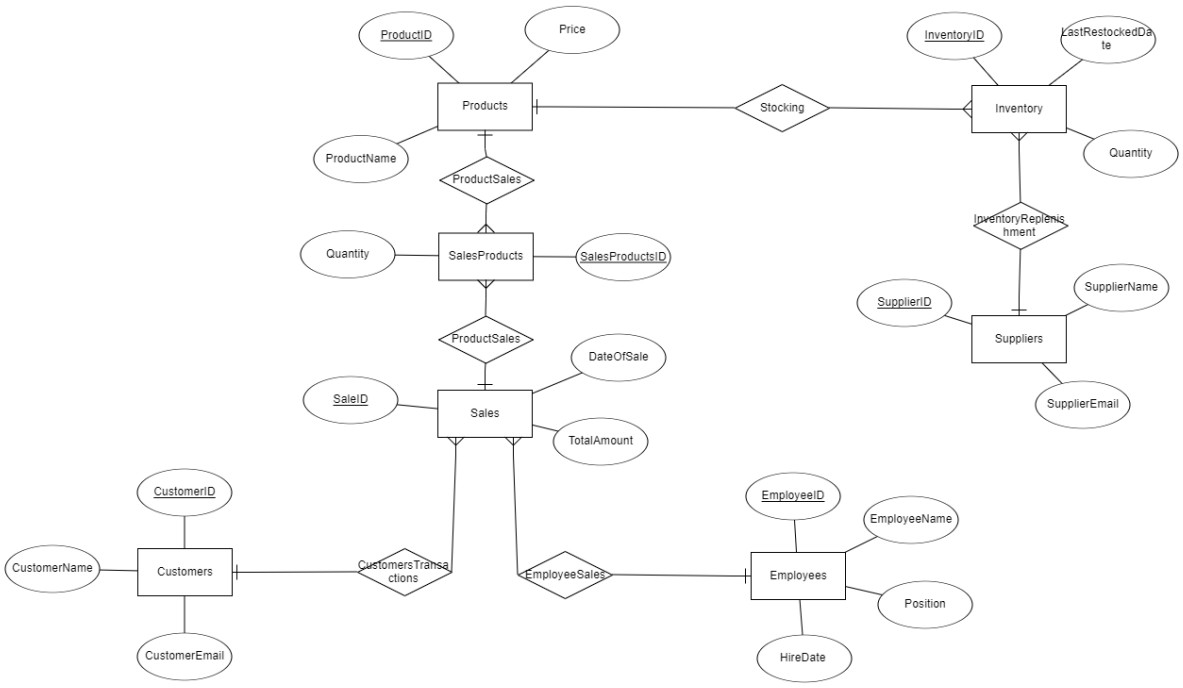
1. Introduction:

The purpose of this project is to design and implement a database system for a store, capturing essential data such as customer details, employee records, product information, sales transactions, inventory status, and supplier information. The primary functionalities include managing customer information, tracking sales, maintaining inventory levels, and ensuring smooth supplier interactions.

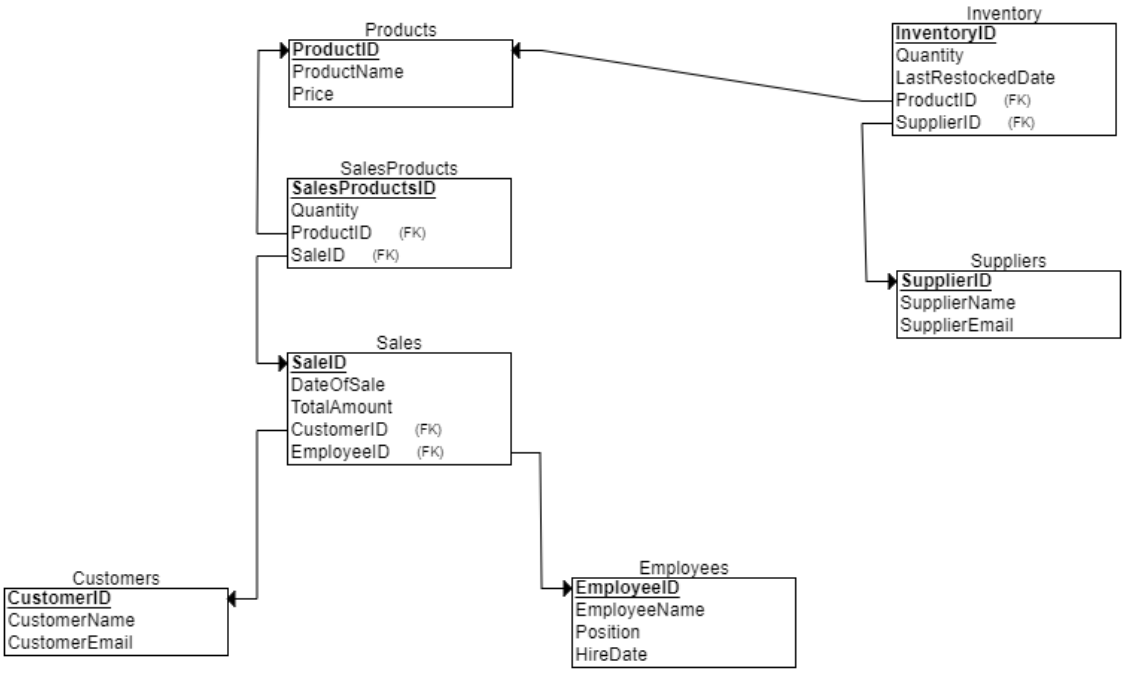
This report details the design, implementation, and testing of the database system, covering the creation of tables, data entry methods, and backup/recovery processes.

2. ERD and DSD Diagrams:

ERD Diagram



DSD Diagram



3. Design decisions :

Many-to-Many Relationship: We chose to represent the many-to-many relationship between Sales and Products using a junction table called SalesProducts to accurately reflect real-world scenarios where multiple products can be part of a single sale.

Foreign Keys: Each table with foreign key constraints helps maintain data integrity and establish clear relationships between entities.

Data Types: INT was used for numeric fields, VARCHAR2 for text fields, and DATE for date fields, ensuring appropriate data representation and storage efficiency.

4. SQL Commands :

- *createTables.sql*

This script contains the SQL commands to create all the necessary tables for the database, including Customers, Employees, Products, Suppliers, Inventory, Sales, and SalesProducts. Each table is defined with its respective columns and data types, and foreign key constraints are set to maintain data integrity and establish relationships between tables.

- *dropTables.sql*

This script provides the SQL commands to drop all the tables created by the createTables.sql script. It ensures that tables are dropped in the correct order, taking into account the dependencies between them, and includes the CASCADE CONSTRAINTS option to handle foreign key constraints smoothly.

- *insertTables.sql*

This script includes SQL commands to insert data into each of the tables in the database. We provided sample data for Customers, Employees, Products, Suppliers, Inventory, Sales, and SalesProducts tables to facilitate the testing and demonstration of the database's functionality.

- *selectAll.sql*

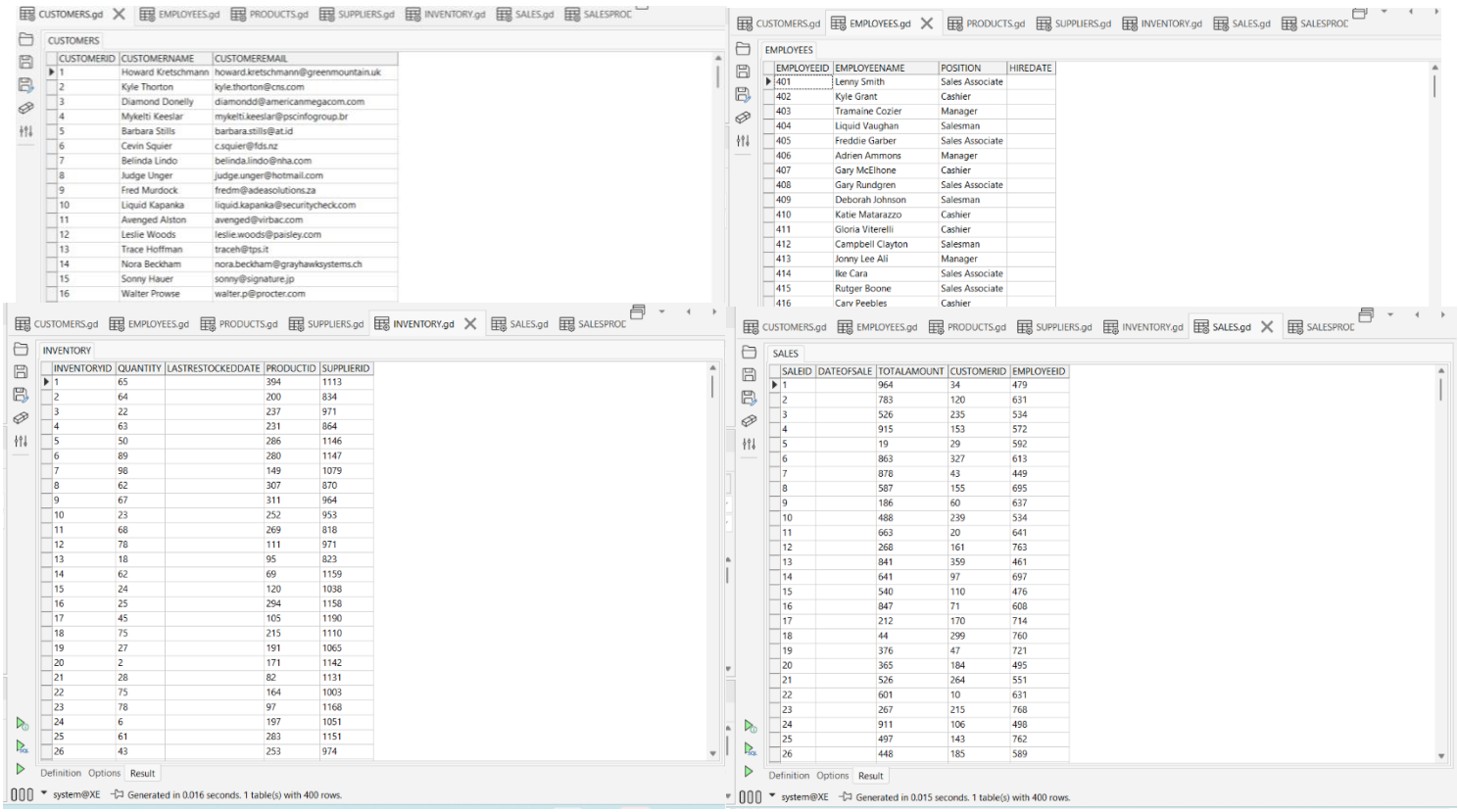
This script contains SQL commands to retrieve and display all data from each of the tables in the database. It is used to verify that the data has been correctly inserted and to provide an overview of the current state of the database.

5. Data Entry Methods

Method 1: Data Generator

We used the Data Generator tool in PL/SQL Developer to generate realistic data for each table.

Screenshots



CUSTOMERS

CUSTOMERID	CUSTOMERNAME	CUSTOMEREMAIL
1	Howard Kretschmann	howard.kretschmann@greenmountain.uk
2	Kyle Thorton	kyle.thorton@cns.com
3	Diamond Donelly	diamondd@americanmegacom.com
4	Mykelti Keeslar	mykelti.keeslar@pscinfogroup.br
5	Barbara Stills	barbara.stills@at.id
6	Cevin Squier	c.squier@fds.nz
7	Belinda Lindo	belinda.lindo@nha.com
8	Judge Unger	judge.unger@hotmail.com
9	Fred Murdock	fredm@adeasolutions.za
10	Liquid Kapanka	liquid.kapanka@securitycheck.com
11	Avenged Alston	avenged@virbac.com
12	Leslie Woods	leslie.woods@paisley.com
13	Trace Hoffman	traceh@tps.it
14	Nora Beckham	nora.beckham@grayhawksystems.ch
15	Sonny Hauer	sonny@signature.jp
16	Walter Prowse	walter.p@procter.com

EMPLOYEES

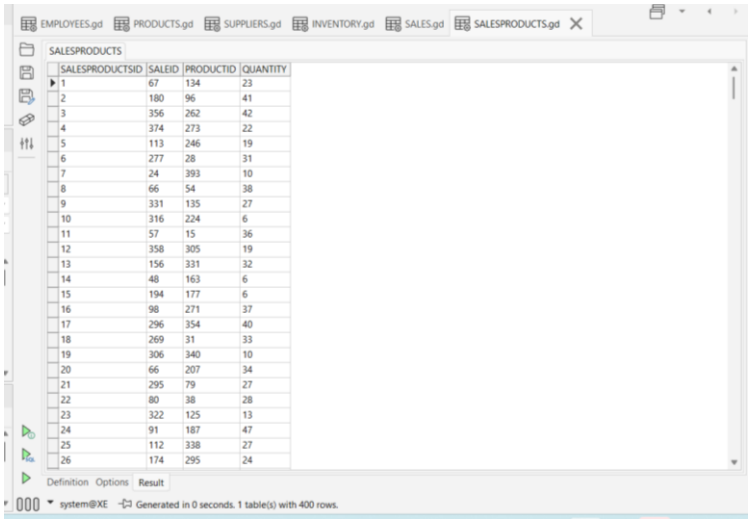
EMPLOYEEID	EMPLOYEENAME	POSITION	HIREDATE
401	Lenny Smith	Sales Associate	
402	Kyle Grant	Cashier	
403	Tramaine Cozier	Manager	
404	Liquid Vaughan	Salesman	
405	Freddie Garber	Sales Associate	
406	Adrien Ammons	Manager	
407	Gary McElhone	Cashier	
408	Gary Rundgren	Sales Associate	
409	Deborah Johnson	Salesman	
410	Katie Matarazzo	Cashier	
411	Gloria Viterelli	Cashier	
412	Campbell Clayton	Salesman	
413	Jonny Lee Ali	Manager	
414	Ike Cara	Sales Associate	
415	Rutger Boone	Sales Associate	
416	Cary Peebles	Cashier	

INVENTORY

INVENTORYID	QUANTITY	LASTRESTOCKEDDATE	PRODUCTID	SUPPLIERID
1	65		394	1113
2	64		200	834
3	22		237	971
4	63		231	864
5	50		286	1146
6	89		280	1147
7	98		149	1079
8	62		307	870
9	67		311	964
10	23		252	953
11	68		269	818
12	78		111	971
13	18		95	823
14	62		69	1159
15	24		120	1038
16	25		294	1158
17	45		105	1190
18	75		215	1110
19	27		191	1065
20	2		171	1142
21	28		82	1131
22	75		164	1003
23	78		97	1168
24	6		197	1051
25	61		283	1151
26	43		253	974

SALES

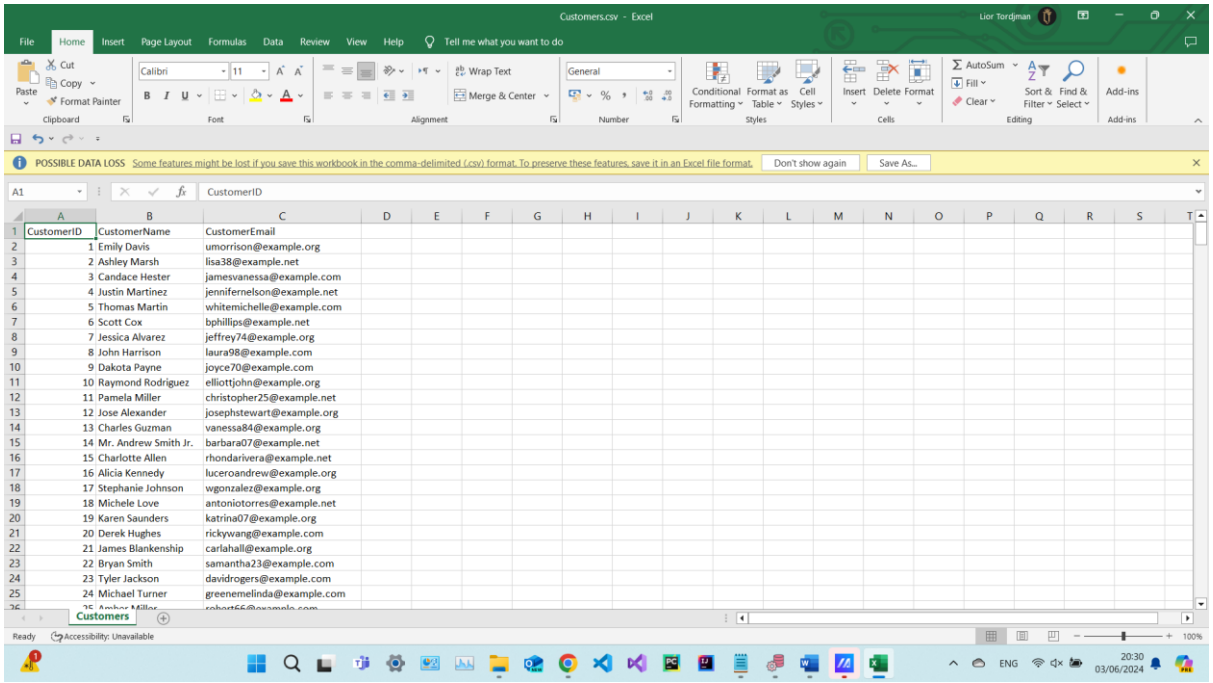
SALEID	DATEOFSALE	TOTALAMOUNT	CUSTOMERID	EMPLOYEEID
1		964	34	479
2		783	120	631
3		526	235	534
4		915	153	572
5		19	29	592
6		863	327	613
7		878	43	449
8		587	155	695
9		186	60	637
10		488	239	534
11		663	20	641
12		268	161	763
13		841	359	461
14		641	97	697
15		540	110	476
16		847	71	608
17		212	170	714
18		44	299	760
19		376	47	721
20		365	184	495
21		526	264	551
22		601	10	631
23		267	215	768
24		911	106	498
25		497	143	762
26		448	185	589



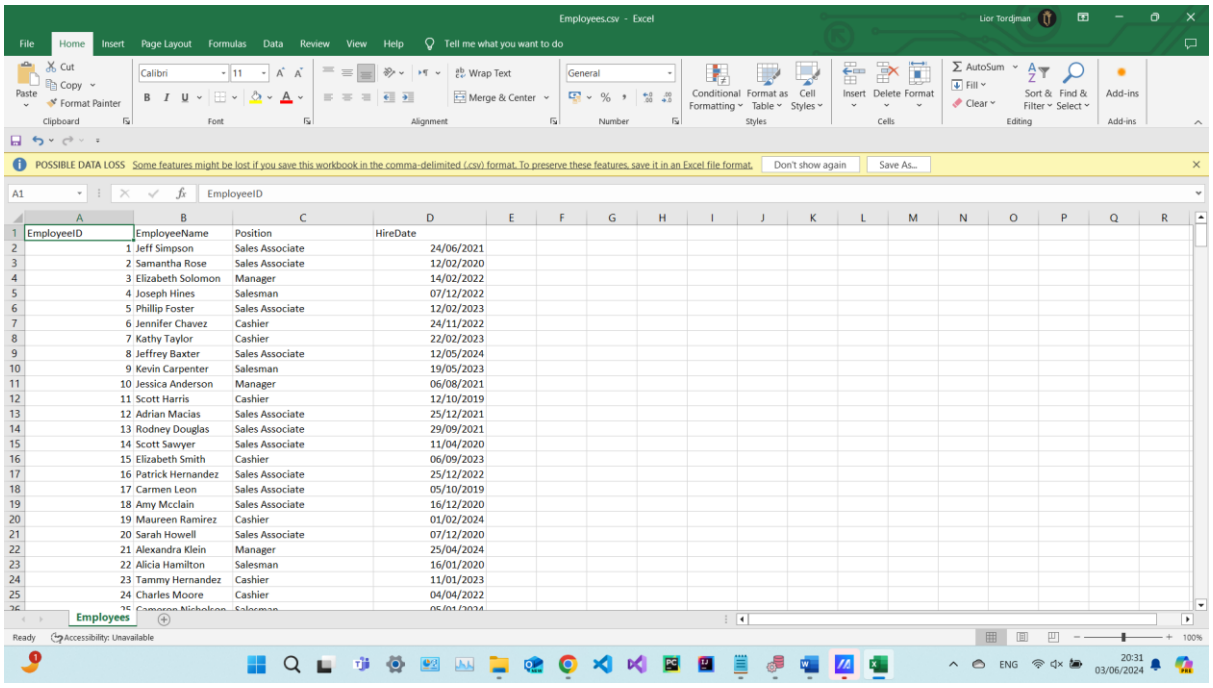
Method 2: CSV Files

We used Python scripts to generate CSV files for each table with 400 records.

Screenshots

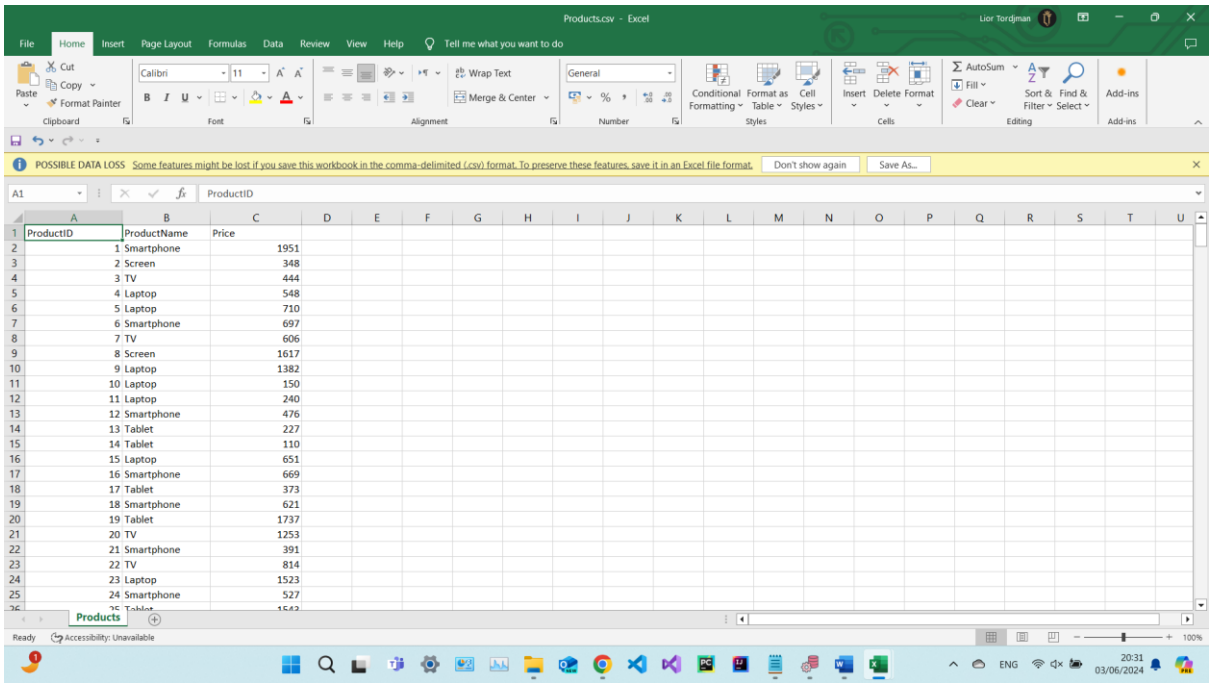


345537708 Samuel Tapiro - 133467 Lior Tordjman - 346012065 Daniel Elbaz



Employees.csv - Excel

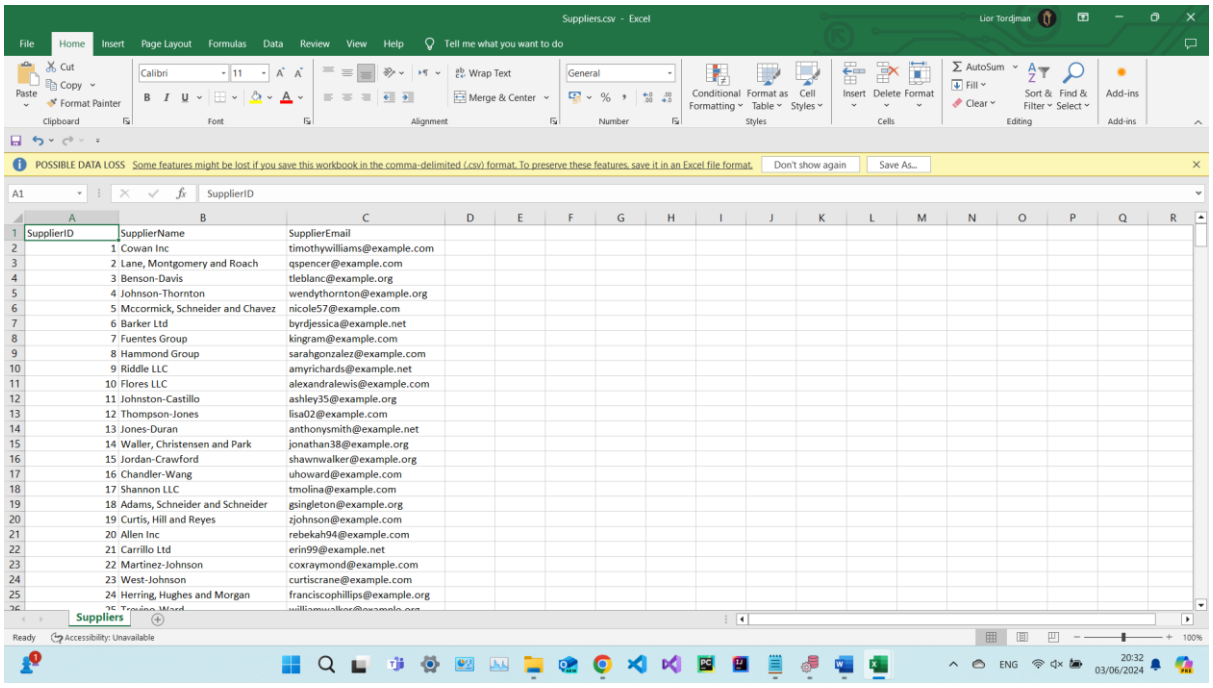
EmployeeID	EmployeeName	Position	HireDate
1	Jeff Simpson	Sales Associate	24/06/2021
2	Samantha Rose	Sales Associate	12/02/2020
3	Elizabeth Solomon	Manager	14/02/2022
4	Joseph Hines	Salesman	07/12/2022
5	Phillip Foster	Sales Associate	12/02/2023
6	Jennifer Chavez	Cashier	24/11/2022
7	Kathy Taylor	Cashier	22/02/2023
8	Jeffrey Baxter	Sales Associate	12/05/2024
9	Kevin Carpenter	Salesman	19/05/2023
10	Jessica Anderson	Manager	06/08/2021
11	Scott Harris	Cashier	12/10/2019
12	Adrian Macias	Sales Associate	25/12/2021
13	Rodney Douglas	Sales Associate	29/09/2021
14	Scott Sawyer	Sales Associate	11/04/2020
15	Elizabeth Smith	Cashier	06/09/2023
16	Patrick Hernandez	Sales Associate	25/12/2022
17	Carmen Leon	Sales Associate	05/10/2019
18	Amy Mcclain	Sales Associate	16/12/2020
19	Maureen Ramirez	Cashier	01/02/2024
20	Sarah Howell	Sales Associate	07/12/2020
21	Alexandra Klein	Manager	25/04/2024
22	Alicia Hamilton	Salesman	16/01/2020
23	Tammy Hernandez	Cashier	11/01/2023
24	Charles Moore	Cashier	04/04/2022



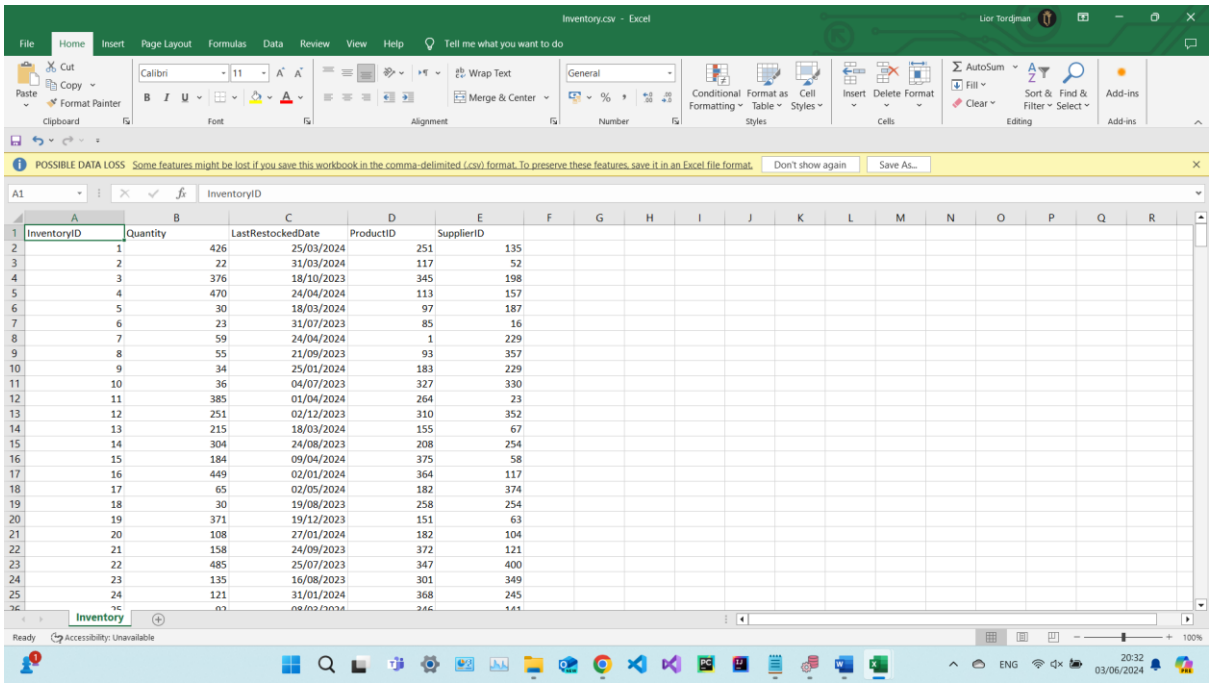
Products.csv - Excel

ProductID	ProductName	Price
1	Smartphone	1951
2	Screen	348
3	TV	444
4	Laptop	548
5	Laptop	710
6	Smartphone	697
7	TV	606
8	Screen	1617
9	Laptop	1382
10	Laptop	150
11	Laptop	240
12	Smartphone	476
13	Tablet	227
14	Tablet	110
15	Laptop	651
16	Smartphone	669
17	Tablet	373
18	Smartphone	621
19	Tablet	1737
20	TV	1253
21	Smartphone	391
22	TV	814
23	Laptop	1523
24	Smartphone	527

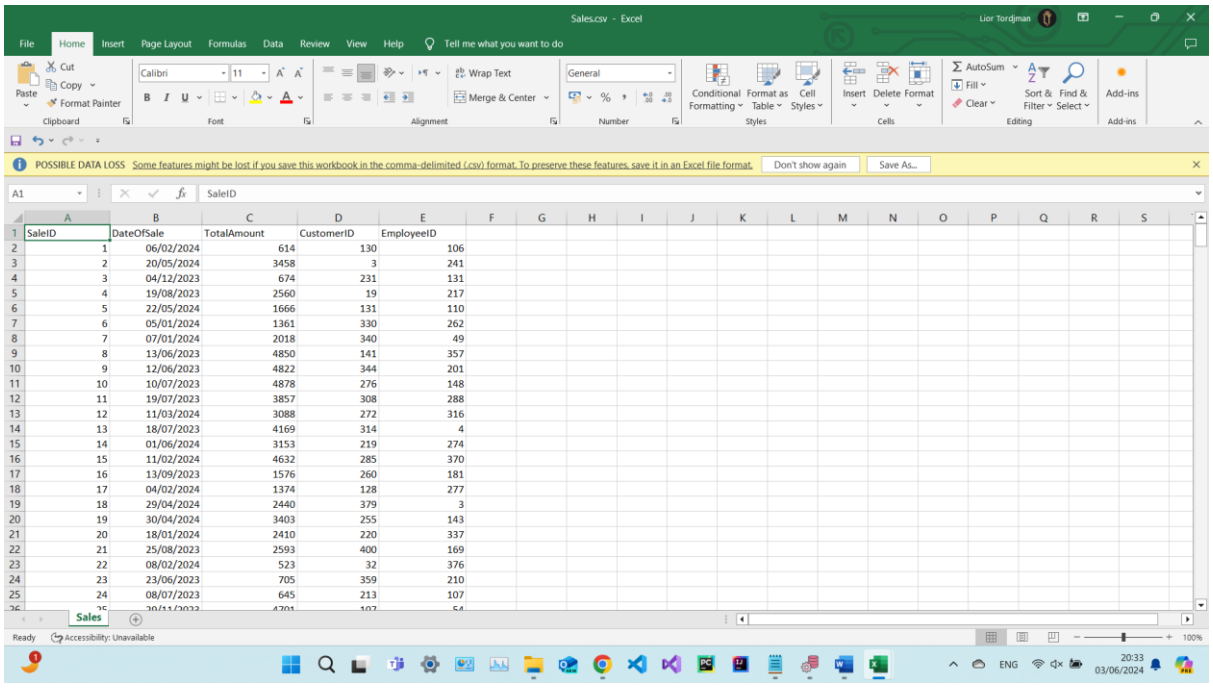
345537708 Samuel Tapiro - 133467 Lior Tordjman - 346012065 Daniel Elbaz



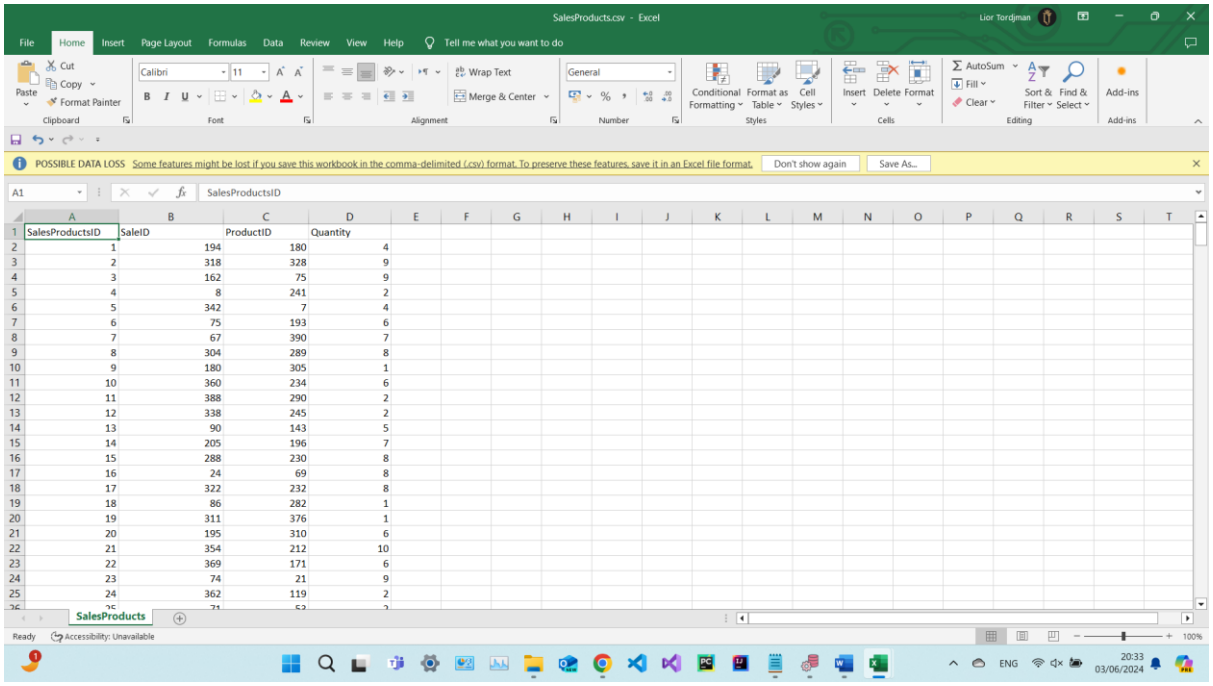
SupplierID	SupplierName	SupplierEmail
1	Cowan Inc	timothywilliams@example.com
2	Lane, Montgomery and Roach	qspencer@example.com
3	Benson-Davis	tleblanc@example.org
4	Johnson-Thornton	wendythornton@example.org
5	Mccormick, Schneider and Chavez	nicole57@example.com
6	Barker Ltd	byrdjessica@example.net
7	Fuentes Group	kingram@example.com
8	Hammond Group	sarahgonzalez@example.com
9	Riddle LLC	amyrichards@example.net
10	Flores LLC	alexandralewis@example.com
11	Johnston-Castillo	ashley35@example.org
12	Thompson-Jones	lisa02@example.com
13	Jones-Duran	anthonysmith@example.net
14	Waller, Christensen and Park	jonathan38@example.org
15	Jordan-Crawford	shawnwalker@example.org
16	Chandler-Wang	uhoward@example.com
17	Shannon LLC	tmolina@example.com
18	Adams, Schneider and Schneider	gsingleton@example.org
19	Curtis, Hill and Reyes	zjohnson@example.com
20	Allen Inc	rebekah94@example.com
21	Carrillo Ltd	erin99@example.net
22	Martinez-Johnson	coxraymond@example.com
23	West-Johnson	curtiscrane@example.com
24	Herring, Hughes and Morgan	franciscophillips@example.org



InventoryID	Quantity	LastRestockedDate	ProductID	SupplierID
1	426	25/03/2024	251	135
2	22	31/03/2024	117	52
3	376	18/10/2023	345	198
4	470	24/04/2024	113	157
5	30	18/03/2024	97	187
6	23	31/07/2023	85	16
7	59	24/04/2024	1	229
8	55	21/09/2023	93	357
9	34	25/01/2024	183	229
10	36	04/07/2023	327	330
11	385	01/04/2024	264	23
12	251	02/12/2023	310	352
13	215	18/03/2024	155	67
14	304	24/08/2023	208	254
15	184	09/04/2024	375	58
16	449	02/01/2024	364	117
17	65	02/05/2024	182	374
18	30	19/08/2023	258	254
19	371	19/12/2023	151	63
20	108	27/01/2024	182	104
21	158	24/09/2023	372	121
22	485	25/07/2023	347	400
23	135	16/08/2023	301	349
24	121	31/01/2024	368	245



SaleID	DateOfSale	TotalAmount	CustomerID	EmployeeID
1	06/02/2024	614	130	106
2	20/05/2024	3458	3	241
3	04/12/2023	674	231	131
4	19/08/2023	2560	19	217
5	22/05/2024	1666	131	110
6	05/01/2024	1361	330	262
7	07/01/2024	2018	340	49
8	13/06/2023	4850	141	357
9	12/06/2023	4822	344	201
10	10/07/2023	4878	276	148
11	19/07/2023	3857	308	288
12	11/03/2024	3088	272	316
13	18/07/2023	4169	314	4
14	01/06/2024	3153	219	274
15	11/02/2024	4632	285	370
16	13/09/2023	1576	260	181
17	04/02/2024	1374	128	277
18	29/04/2024	2440	379	3
19	30/04/2024	3403	255	143
20	18/01/2024	2410	220	337
21	25/08/2023	2593	400	169
22	08/02/2024	523	32	376
23	23/06/2023	705	359	210
24	08/07/2023	645	213	107



SalesProductsID	SaleID	ProductID	Quantity
1	194	180	4
2	318	328	9
3	162	75	9
4	8	241	2
5	342	7	4
6	75	193	6
7	67	390	7
8	304	289	8
9	180	305	1
10	360	234	6
11	388	290	2
12	338	245	2
13	90	143	5
14	205	196	7
15	288	230	8
16	24	69	8
17	322	232	8
18	86	282	1
19	311	376	1
20	195	310	6
21	354	212	10
22	369	171	6
23	74	21	9
24	362	119	2

Method 3: Programming with Python

We created Python scripts to automate the data insertion process, leveraging the Faker library to generate realistic data (in Programming folder on the GitHub).

6. Data Backup and Recovery

Backup Process

345537708 Samuel Tapiro - 133467 Lior Tordjman - 346012065 Daniel Elbaz

We used PL/SQL Developer to create a backup of the database using the SQL Insert method, ensuring that CREATE TABLE commands were included.

Screenshots