

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.

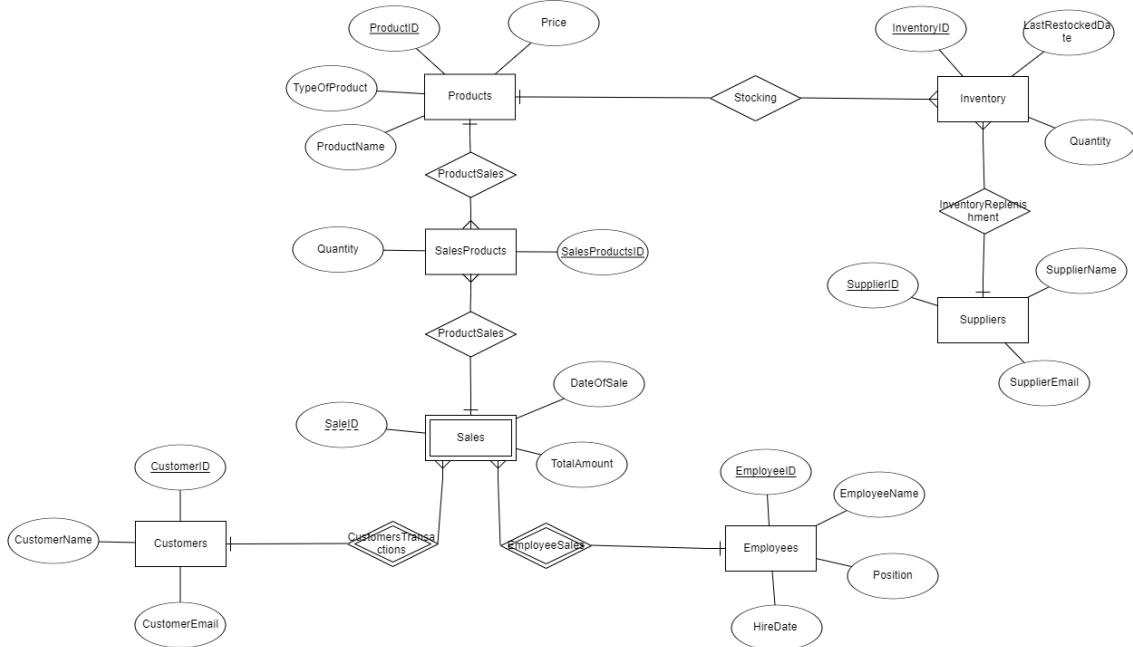
This report details the design, implementation, and testing of the database system, covering the creation of tables, data entry methods, and backup/recovery processes.

Entities and Their Functions:

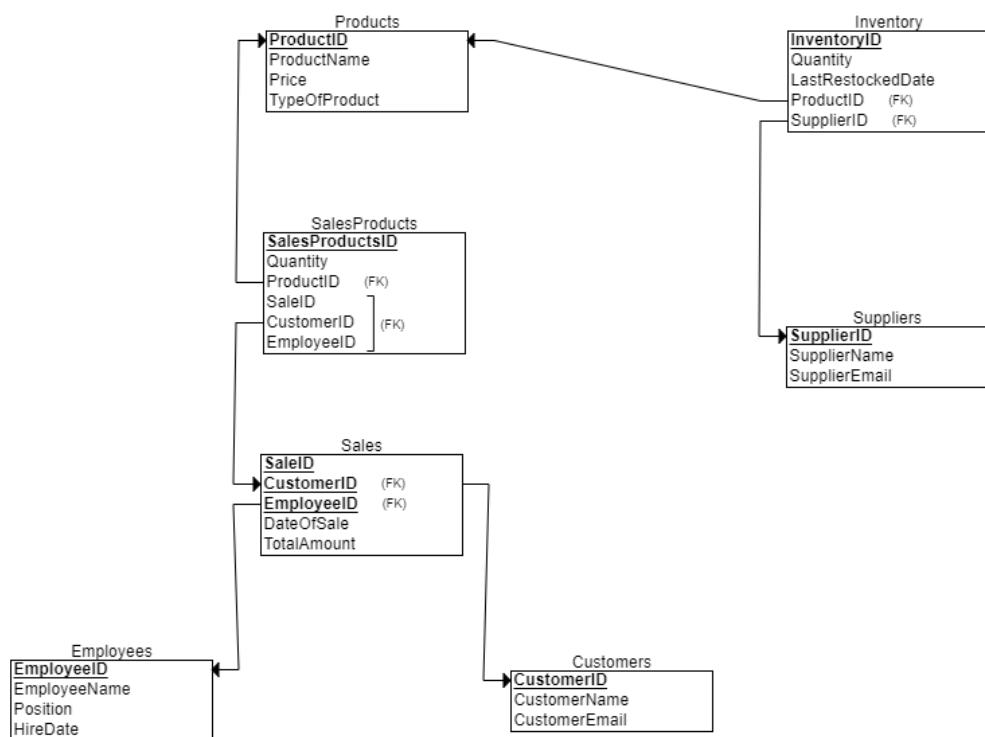
- **Customers:** This entity stores information about the customers of the store, including their unique ID, name, and email address.
- **Employees:** This entity holds data about the store's employees, including their unique ID, name, position, and hire date.
- **Products:** This entity captures details about the products available in the store, including their unique ID, name, type, and price.
- **Suppliers:** This entity maintains information about the suppliers who provide products to the store, including their unique ID, name, and email address.
- **Inventory:** This entity tracks the stock of products in the store, including the unique inventory ID, quantity, last restocked date, product ID, and supplier ID.
- **Sales:** As a weak entity identified by CustomerID and EmployeeID, this entity records sales transactions, including the unique sale ID, date of sale, total amount, customer ID, and employee ID involved in the sale.
- **SalesProducts:** This junction table manages the many-to-many relationship between sales and products, including the unique sales-products ID, sale ID, customer ID, employee ID, product ID, and quantity of each product sold in a sale.

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.
- Sales as a Weak Entity: The Sales entity was designed as a weak entity, identified by Customers and Employees, to better reflect the relationship between sales transactions and the individuals involved.

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

The image displays two side-by-side screenshots of the PL/SQL Developer application interface. Both screenshots show the 'Data' tab selected in the top menu bar, which contains various tools like Explain Plan, Code Assistant, and Data Generator.

Top Screenshot (CUSTOMERS Table):

- Connections:** Shows 'Imported Fixed Users' and 'Imported History'.
- Objects:** Shows 'CUSTOMERS' table with columns: CUSTOMERID, CUSTOMERNAME, and CUSTOMEREMAIL.
- Table Data:** Displays rows of generated data for the CUSTOMERS table, such as:

CUSTOMERID	CUSTOMERNAME	CUSTOMEREMAIL
374	Eugene Langella	eugene.langella@hospitalsolutions.com
375	Harrison Delta	harrison.delta@componentgraphics.jp
376	Claude McLachlan	claude.mclachlan@gra.ca
377	Carl Keitel	c.keitel@restaurantpartners.dk
378	Donald Hatchet	donal.hatchet@accesssystems.cr
379	Andrae Epps	a.epps@als.com
380	Peter Goodman	peter@canterburypark.ch
381	Lorraine Brock	lbrock@dataprise.ca
382	Robert Mulroney	robert.mulroney@invisioncom.de
383	Lizzy Norton	lizzy@deutschtelekom.com
384	Claude Union	claude@priorityleasing.uk
385	Bob Curtis	bob.c@gillette.cl
386	Claude Nelson	c.nelson@ass.com
387	Maria Sherman	maria@capstone.com
388	Veruca Gyllenhaal	veruca.gyllenhaal@hardwoodwholesalers.com
389	Avenged de Lancia	avenged@ads.com
390	Nancy Paquin	n.paquin@newtoninteractive.it
391	Nik Weber	nik.weber@gentraysystems.com
392	Stevie Holly	stevie.holly@saltgroup.com
393	Lou Kirkwood	lou.kirkwood@vendinuniversal.id
394	Rascal Dawson	rascal.dawson@bestbuy.ca
395	Alfred Fichtner	alfred.fichtner@valleyoaksystems.jp
396	Olympia Judd	olympia.judd@atg.de
397	Vienna Chambers	vienna@scheringplough.uk
398	Kyle Rudd	kyle.rudd@technica.com
399	Anjelica Leary	anjelica@teoco.nl
- Clipboard:** Shows generated SQL code for creating the CUSTOMERS table and inserting data into it.

Bottom Screenshot (EMPLOYEES Table):

- Connections:** Shows 'Imported Fixed Users' and 'Imported History'.
- Objects:** Shows 'EMPLOYEES' table with columns: EMPLOYEEID, EMPLOYEENAME, POSITION, and HIREDATE.
- Table Data:** Displays rows of generated data for the EMPLOYEES table, such as:

EMPLOYEEID	EMPLOYEENAME	POSITION	HIREDATE
401	Rascal Diesel	Cashier	25/10/2021
402	Liam Kingsley	Sales Associate	04/04/2022
403	Wade Webb	Cashier	04/05/2021
404	Victor Benson	Cashier	02/08/2020
405	John Jolie	Manager	21/01/2020
406	Earl Epps	Sales Associate	03/05/2020
407	Hilton Adams	Manager	10/07/2022
408	Antonio Lizzy	Product Manager	27/01/2021
409	Mary-Louise Bracco	Cashier	18/07/2021
410	Rhett Gellar	Manager	06/02/2020
411	Rick Perry	Product Manager	24/09/2022
412	Julie Strathairn	Sales Associate	23/02/2023
413	Lydia Borgenine	Cashier	13/04/2023
414	Don Janney	Sales Associate	16/04/2021
415	Wendy LuPone	Cashier	24/01/2021
416	Kris Bruce	Product Manager	19/02/2022
417	Cathy Frampton	Manager	30/11/2021
418	Gaby Everett	Product Manager	01/04/2023
419	Devon Levine	Manager	04/03/2021
420	Belinda Maguire	Cashier	01/11/2023
421	Bret Cale	Manager	06/07/2023
422	Jennifer Goldwyn	Product Manager	31/08/2022
423	Kyra Austin	Sales Associate	22/12/2020
424	Spencer Spiner	Manager	07/04/2022
425	Martin Culkin	Product Manager	17/04/2021
426	Aaron Swinton	Cashier	30/04/2023
- Clipboard:** Shows generated SQL code for creating the EMPLOYEES table and inserting data into it.

345537708 Samuel Tapiro - 133467 Lior Tordjman - 346012065 Daniel Elbaz

system@XE - PL/SQL Developer - PRODUCTS.gd

The screenshot shows the PL/SQL Developer interface with the 'PRODUCTS' table selected. The table has columns: PRODUCTID, PRODUCTNAME, TYPEOFPRODUCT, and PRICE. The data consists of 25 rows of product information.

PRODUCTID	PRODUCTNAME	TYPEOFPRODUCT	PRICE
10000	Phone	Sports Equipment	443
10001	Skin Care	Beauty	19
10002	Phone	Pet Supplies	150
10003	Skin Care	Pet Supplies	313
10004	Dog Food	Beauty	450
10005	Dog Food	Beauty	11
10006	Dog Food	Pet Supplies	371
10007	Phone	Electronics	444
10008	Sports Shoes	Pet Supplies	239
10009	Dog Food	Sports Equipment	427
10010	Skin Care	Beauty	345
10011	Laptop	Beauty	313
10012	Dog Food	Electronics	388
10013	Laptop	Sports Equipment	379
10014	Laptop	Electronics	433
10015	Laptop	Sports Equipment	20
10016	Laptop	Electronics	332
10017	Dog Food	Beauty	89
10018	Dog Food	Beauty	49
10019	Laptop	Pet Supplies	487
10020	Dog Food	Sports Equipment	314
10021	Skin Care	Beauty	375
10022	Skin Care	Electronics	478
10023	Phone	Beauty	102
10024	Phone	Pet Supplies	68
10025	Laptop	Pet Supplies	421

system@XE - PL/SQL Developer - SUPPLIERS.gd

The screenshot shows the PL/SQL Developer interface with the 'SUPPLIERS' table selected. The table has columns: SUPPLIERID, SUPPLIERNAME, and SUPPLIEREMAIL. The data consists of 26 rows of supplier information.

SUPPLIERID	SUPPLIERNAME	SUPPLIEREMAIL
801	Laura Lewis	laura.l@amerisourcefunding.br
802	Donal Raybon	donal.r@eastmankodak.com
803	France Ward	france.ward@pragmatechsoftware.de
804	Ned Davidson	ned.davidson@linensdirect.si
805	William Uggams	william.uggams@aco.ie
806	Lynette Coe	lynnette.coe@pharmafab.ca
807	Brad Bonneville	brad.bonneville@priorityexpress.com
808	Lou Ledger	lou@fsc.com
809	Daniel Brooks	daniel.brooks@capitolbancorp.com
810	Lloyd Playboys	lloyd.playboys@yeb.com
811	Geoff Shannon	geoff@ecopy.com
812	Diane Kier	diane.kier@conquestsystems.de
813	Cliff Rhodes	cliff.r@solipsys.com
814	Geoffrey Chinlund	geoffrey@flavones
815	Janice Smith	janice@rnm.au
816	Maureen Mattea	mmattea@reckittbenckiser.jp
817	Loren Cassel	loren.cassel@httpprint.com
818	Pierce Beck	pierce.beck@bioste.com
819	Goldie Mould	goldie.mould@ecom.de
820	Jimmie Burns	jimmie.b@mission.pt
821	Lucinda Cromwell	lcromwell@upb.com
822	Chloe Vaughan	chloe.vaughan@msdw.com
823	Frankie Head	frankie.h@tama.fr
824	Nickel Warwick	nickel@bayer.za
825	Stewart Renfro	stewart.renfro@dpd.de
826	Lennie Fiorentino	lennie.fiorentino@bigdough.com

345537708 Samuel Tapiro - 133467 Lior Tordjman - 346012065 Daniel Elbaz

PL/SQL Developer - INVENTORY.gd

INVENTORY

INVENTORYID	QUANTITY	LASTRESTOCKEDDATE	PRODUCTID	SUPPLIERID
20000	915	30/03/2023	10000	801
20001	284	30/08/2022	10001	802
20002	131	16/12/2023	10002	803
20003	802	26/07/2020	10003	804
20004	839	05/04/2022	10004	805
20005	920	03/09/2020	10005	806
20006	309	23/04/2021	10006	807
20007	472	18/03/2020	10007	808
20008	517	05/06/2020	10008	809
20009	157	02/04/2023	10009	810
20010	886	31/10/2021	10010	811
20011	668	02/12/2020	10011	812
20012	752	18/08/2023	10012	813
20013	603	07/08/2023	10013	814
20014	774	22/09/2023	10014	815
20015	55	22/03/2022	10015	816
20016	767	11/05/2020	10016	817
20017	950	10/12/2020	10017	818
20018	711	02/12/2023	10018	819
20019	518	20/04/2020	10019	820
20020	903	06/10/2023	10020	821
20021	505	08/02/2023	10021	822
20022	492	17/09/2022	10022	823
20023	971	13/03/2020	10023	824
20024	886	25/11/2020	10024	825
20025	791	30/04/2022	10025	826

SALES

SALEID	DATEOFSALE	TOTALAMOUNT	CUSTOMERID	EMPLOYEEID
30000	07/05/2020	49	1	401
30001	20/12/2020	3159	2	402
30002	28/08/2020	2929	3	403
30003	25/10/2022	4420	4	404
30004	05/11/2021	3901	5	405
30005	20/08/2023	2579	6	406
30006	25/08/2023	4060	7	407
30007	02/12/2021	145	8	408
30008	31/10/2022	3880	9	409
30009	29/04/2021	4598	10	410
30010	02/02/2023	1003	11	411
30011	14/07/2021	4876	12	412
30012	03/08/2022	3275	13	413
30013	07/04/2022	4723	14	414
30014	03/05/2021	806	15	415
30015	26/01/2023	4880	16	416
30016	31/07/2023	1796	17	417
30017	10/08/2022	2049	18	418
30018	21/05/2023	3505	19	419
30019	09/02/2022	3928	20	420
30020	26/08/2022	640	21	421
30021	13/09/2022	964	22	422
30022	06/09/2023	4844	23	423
30023	11/04/2023	1445	24	424
30024	05/07/2023	372	25	425
30025	11/08/2022	3185	26	426

INVENTORY Clipboard

```

CREATE TABLE Customers (
    CustomerID INT PRIMARY KEY,
    CustomerName VARCHAR2(100),
    CustomerEmail VARCHAR2(100)
);

SELECT * FROM Customers;

CREATE TABLE Sales (
    SaleID INT PRIMARY KEY,
    CustomerID INT,
    EmployeeID INT,
    TotalAmount NUMBER
);

SELECT * FROM Sales;

CREATE TABLE Products (
    ProductID INT PRIMARY KEY,
    ProductName VARCHAR2(100),
    Description VARCHAR2(255),
    UnitPrice NUMBER
);

SELECT * FROM Products;

CREATE TABLE Suppliers (
    SupplierID INT PRIMARY KEY,
    SupplierName VARCHAR2(100),
    ContactName VARCHAR2(100),
    Address VARCHAR2(255),
    City VARCHAR2(50),
    PostalCode VARCHAR2(20),
    Phone VARCHAR2(20)
);

SELECT * FROM Suppliers;

CREATE TABLE Staff (
    StaffID INT PRIMARY KEY,
    StaffName VARCHAR2(100),
    HireDate DATE,
    JobTitle VARCHAR2(50),
    ManagerID INT,
    DepartmentID INT
);

SELECT * FROM Staff;

CREATE TABLE Inventory (
    InventoryID INT PRIMARY KEY,
    ProductID INT,
    SupplierID INT,
    Quantity NUMBER,
    LastStockedDate DATE
);

SELECT * FROM Inventory;

```

PL/SQL Developer - SALES.gd

SALES

SALEID	DATEOFSALE	TOTALAMOUNT	CUSTOMERID	EMPLOYEEID
30000	07/05/2020	49	1	401
30001	20/12/2020	3159	2	402
30002	28/08/2020	2929	3	403
30003	25/10/2022	4420	4	404
30004	05/11/2021	3901	5	405
30005	20/08/2023	2579	6	406
30006	25/08/2023	4060	7	407
30007	02/12/2021	145	8	408
30008	31/10/2022	3880	9	409
30009	29/04/2021	4598	10	410
30010	02/02/2023	1003	11	411
30011	14/07/2021	4876	12	412
30012	03/08/2022	3275	13	413
30013	07/04/2022	4723	14	414
30014	03/05/2021	806	15	415
30015	26/01/2023	4880	16	416
30016	31/07/2023	1796	17	417
30017	10/08/2022	2049	18	418
30018	21/05/2023	3505	19	419
30019	09/02/2022	3928	20	420
30020	26/08/2022	640	21	421
30021	13/09/2022	964	22	422
30022	06/09/2023	4844	23	423
30023	11/04/2023	1445	24	424
30024	05/07/2023	372	25	425
30025	11/08/2022	3185	26	426

SALES Clipboard

```

CREATE TABLE Customers (
    CustomerID INT PRIMARY KEY,
    CustomerName VARCHAR2(100),
    CustomerEmail VARCHAR2(100)
);

SELECT * FROM Customers;

CREATE TABLE Sales (
    SaleID INT PRIMARY KEY,
    CustomerID INT,
    EmployeeID INT,
    TotalAmount NUMBER
);

SELECT * FROM Sales;

CREATE TABLE Products (
    ProductID INT PRIMARY KEY,
    ProductName VARCHAR2(100),
    Description VARCHAR2(255),
    UnitPrice NUMBER
);

SELECT * FROM Products;

CREATE TABLE Suppliers (
    SupplierID INT PRIMARY KEY,
    SupplierName VARCHAR2(100),
    ContactName VARCHAR2(100),
    Address VARCHAR2(255),
    City VARCHAR2(50),
    PostalCode VARCHAR2(20),
    Phone VARCHAR2(20)
);

SELECT * FROM Suppliers;

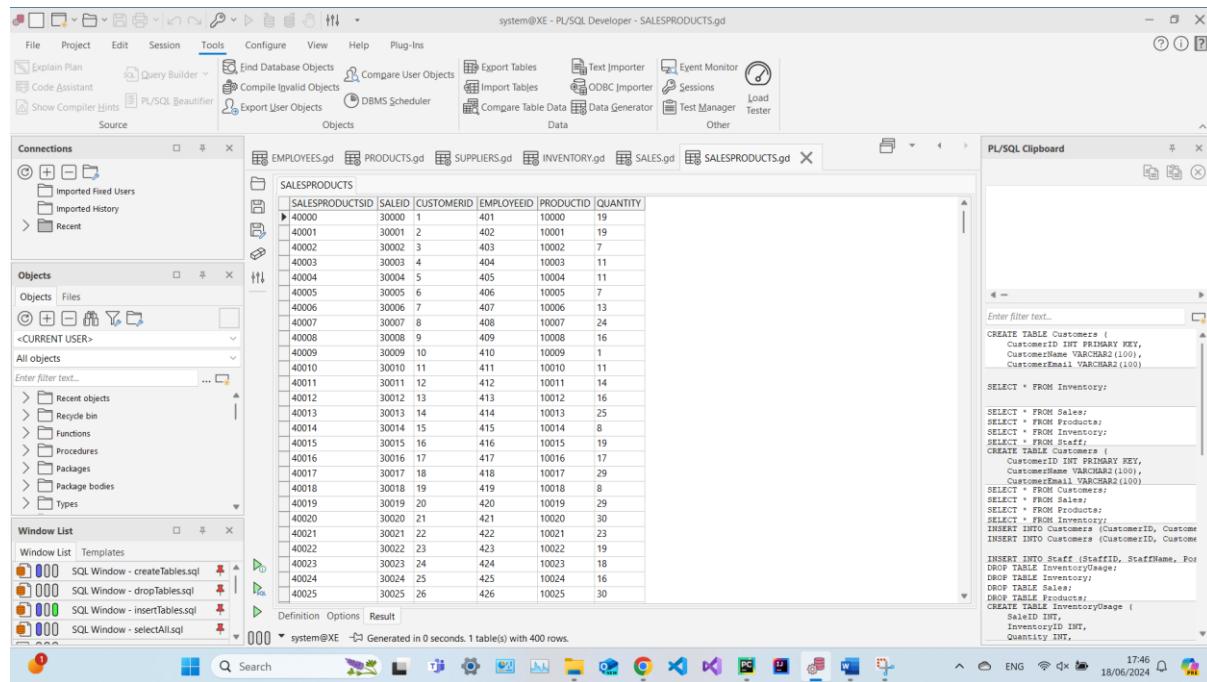
CREATE TABLE Staff (
    StaffID INT PRIMARY KEY,
    StaffName VARCHAR2(100),
    HireDate DATE,
    JobTitle VARCHAR2(50),
    ManagerID INT,
    DepartmentID INT
);

SELECT * FROM Staff;

CREATE TABLE Inventory (
    InventoryID INT PRIMARY KEY,
    ProductID INT,
    SupplierID INT,
    Quantity NUMBER,
    LastStockedDate DATE
);

SELECT * FROM Inventory;

```



Method 2: CSV Files

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

Screenshots

The screenshot shows an Excel spreadsheet titled "Customers.csv - Excel". The table has columns: CustomerID, CustomerName, and CustomerEmail. The data consists of 400 rows of customer information, such as Michelle Brown, David Norris, Hannah Allen, etc., with their respective email addresses.

CustomerID	CustomerName	CustomerEmail
1	Michelle Brown	randallclark@example.net
2	David Norris	andrew28@example.com
3	Hannah Allen	sean14@example.org
4	Nicole Ewing	karenstevens@example.org
5	Daniel Wilson	jensenronald@example.net
6	Kevin Andrews	josephthomas@example.net
7	Devin Harris	sheilaclark@example.com
8	Dylan Proctor	warnerkenneth@example.com
9	Andrew Coleman	michellegarcia@example.com
10	David Morris	kcoffey@example.org
11	Kathryn Cook	christopher73@example.net
12	Teresa Gallagher	mitchelljerry@example.com
13	Tina Ward	jennifer91@example.org
14	Richard Woods	christopherreyes@example.com
15	Daniel Carpenter	rossleslie@example.com
16	William Rodgers	troy72@example.com
17	Robert Hall	wanda42@example.org
18	Mary Small	pdominguez@example.com
19	Miguel Griffin	mark60@example.org
20	Scott Moore	rogerherandez@example.org
21	Joseph Reynolds	millerjasmine@example.com
22	Ashley Winters	lisa01@example.com
23	Samuel Owens	richardquinn@example.com
24	Tyler Maynard	heather30@example.com
25	Linda Daniels	lindafish@example.com

345537708 Samuel Tapiro - 133467 Lior Tordjman - 346012065 Daniel Elbaz

Employees.csv - Excel

The screenshot shows the Microsoft Excel interface with the title bar "Employees.csv - Excel". The ribbon menu is visible with tabs like File, Home, Insert, Page Layout, Formulas, Data, Review, View, and Help. A message bar at the top says "POSSIBLE DATA LOSS Some features might be lost if you save this workbook in the comma-delimited (.csv) format. To preserve these features, save it in an Excel file format." Below the message is a "Don't show again" button and a "Save As..." button. The main content area displays a table with 26 rows of employee data. The columns are labeled A through S. Column A is "EmployeeID", column B is "EmployeeName", column C is "Position", and column D is "HireDate". The data includes various roles like HR Specialist, Manager, Sales Associate, IT Support Specialist, etc., with hire dates ranging from 01/01/2020 to 27/05/2024.

EmployeeID	EmployeeName	Position	HireDate
1	Cody Larson	HR Specialist	21/01/2022
2	Noah Dixon	HR Specialist	01/01/2020
3	Christopher Swanson	Manager	04/10/2020
4	Mark Murphy	Finance Analyst	05/04/2021
5	Tina Dean	Sales Associate	27/05/2024
6	Robert Perez	IT Support Specialist	14/05/2021
7	Sophia Cox	Finance Analyst	19/05/2023
8	Mitchell Hayes	HR Specialist	15/12/2021
9	Michelle Matthews	Sales Associate	06/10/2019
10	Michael Greene	Inventory Specialist	17/06/2022
11	James Dean	Marketing Coordinator	07/03/2021
12	Eric Lambert	Cashier	04/12/2022
13	Bradley Johnson	Sales Associate	18/06/2022
14	Jonathan Wood	HR Specialist	11/11/2022
15	Jessica Mora	Finance Analyst	18/08/2021
16	Jacqueline Johns	Cashier	14/07/2022
17	Heidi Reyes	Marketing Coordinator	05/10/2019
18	Katrina Dixon	Manager	21/03/2022
19	Ryan Wells	Marketing Coordinator	20/10/2023
20	Anthony Phillips	Finance Analyst	28/12/2019
21	Madison Morrison	Inventory Specialist	17/05/2023
22	Jennifer Vazquez	Product Manager	07/03/2022
23	Matthew Zuniga	Customer Service Represen	14/10/2021
24	Jeffrey Smith	Finance Analyst	21/12/2020
25	John Wallace	Sales Associate	06/06/2022

Products.csv - Excel

The screenshot shows the Microsoft Excel interface with the title bar "Products.csv - Excel". The ribbon menu is visible with tabs like File, Home, Insert, Page Layout, Formulas, Data, Review, View, and Help. A message bar at the top says "POSSIBLE DATA LOSS Some features might be lost if you save this workbook in the comma-delimited (.csv) format. To preserve these features, save it in an Excel file format." Below the message is a "Don't show again" button and a "Save As..." button. The main content area displays a table with 26 rows of product data. The columns are labeled A through U. Column A is "ProductID", column B is "ProductName", column C is "TypeOfProduct", and column D is "Price". The data includes various product types like Electronics, Accessories, and Computers, with prices ranging from 373 to 1798.

ProductID	ProductName	TypeOfProduct	Price
1	Laptop	Accessories	1349
2	Tablet	Electronics	1616
3	Smartphone	Electronics	1594
4	Desktop	Accessories	959
5	Router	Electronics	1095
6	Mouse	Accessories	165
7	Smartphone	Accessories	888
8	Mouse	Electronics	1063
9	Printer	Electronics	1408
10	Smartphone	Accessories	373
11	Monitor	Electronics	1946
12	Keyboard	Electronics	1549
13	Monitor	Electronics	1611
14	Tablet	Accessories	254
15	Monitor	Accessories	754
16	Smartphone	Accessories	659
17	Camera	Accessories	956
18	Printer	Accessories	1603
19	Monitor	Electronics	1226
20	Printer	Electronics	1520
21	Tablet	Accessories	1798
22	Router	Accessories	501
23	Desktop	Electronics	1470
24	Printer	Electronics	1402
25	Monitor	Accessories	220

345537708 Samuel Tapiro - 133467 Lior Tordjman - 346012065 Daniel Elbaz

Suppliers.csv - Excel

File Home Insert Page Layout Formulas Data Review View Help Tell me what you want to do

Possible data loss: Some features might be lost if you save this workbook in the comma-delimited (.csv) format. To preserve these features, save it in an Excel file format.

SupplierID	SupplierName	SupplierEmail
1	Holmes, Wright and Edwards	lopezsarah@example.org
2	Diaz-Johnson	jmedina@example.com
3	Mitchell, Bishop and Pugh	hjones@example.net
4	Wilkin, Brown and Fox	brandon79@example.org
5	Pearson-Wilson	randyfrye@example.com
6	Ward, Roberts and Carpenter	jenkinsara@example.net
7	Brown PLC	michellelopez@example.net
8	Cook-Monroe	erogers@example.com
9	Jones, Davis and Smith	hamptonolivia@example.org
10	Harris, Gilbert and Anthony	fsavage@example.org
11	Russell-Duncan	cooperalfred@example.net
12	Hanson Inc	smithchristopher@example.com
13	Beck, Moore and Gallagher	dallen@example.net
14	Hansen, Mason and Johnson	christopherjacobs@example.com
15	Buchanan Group	kellyjackson@example.com
16	Mooney-Brown	xfraizer@example.org
17	Hernandez and Sons	matthew52@example.org
18	Cummings, Thomas and Ford	griffinjoseph@example.com
19	Mills LLC	oreese@example.com
20	Gonzalez, Williams and Scott	wilsonangela@example.com
21	Mathis, Hernandez and Waters	danielmoses@example.net
22	Bennett Ltd	parkerkristen@example.net
23	Elliott and Sons	kimmorris@example.net
24	Fritz, Benjamin and Patterson	samuel16@example.org
25	O'Neal, Jackson and Clark	danielolivias@example.net

Inventory.csv - Excel

File Home Insert Page Layout Formulas Data Review View Help Tell me what you want to do

Possible data loss: Some features might be lost if you save this workbook in the comma-delimited (.csv) format. To preserve these features, save it in an Excel file format.

InventoryID	Quantity	LastRestockedDate	ProductID	SupplierID
1	92	29/07/2023	106	187
2	68	10/02/2024	326	101
3	66	30/08/2023	5	13
4	115	17/03/2024	299	202
5	294	15/08/2023	317	42
6	86	16/12/2023	93	141
7	465	16/12/2023	294	371
8	55	09/01/2024	136	242
9	246	20/02/2024	81	287
10	26	11/12/2023	228	99
11	14	01/03/2024	231	250
12	125	07/09/2023	109	160
13	36	13/10/2023	96	44
14	443	15/05/2024	179	393
15	62	29/12/2023	159	374
16	91	09/03/2024	95	131
17	154	30/06/2023	381	256
18	159	22/10/2023	211	249
19	372	11/11/2023	266	167
20	443	16/04/2024	26	61
21	108	29/01/2024	247	249
22	363	07/03/2024	15	210
23	253	22/02/2024	371	11
24	457	19/03/2024	24	114
25	201	27/03/2024	229	200

Sales.csv - Excel

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	SaleID	DateOfSale	TotalAmount	CustomerID	EmployeeID															
2	1	30/10/2023	436	211	21															
3	2	18/09/2023	1038	31	81															
4	3	11/10/2023	1282	267	384															
5	4	04/08/2023	3260	351	81															
6	5	30/08/2023	1986	131	228															
7	6	27/03/2024	3759	306	303															
8	7	27/05/2024	3047	341	285															
9	8	07/06/2024	4578	396	345															
10	9	20/10/2023	125	218	165															
11	10	19/06/2023	3345	153	77															
12	11	27/07/2023	1679	13	358															
13	12	01/01/2024	4723	220	87															
14	13	31/07/2023	4707	39	115															
15	14	12/03/2024	1722	193	84															
16	15	14/10/2023	3925	203	92															
17	16	25/09/2023	4644	309	322															
18	17	05/01/2024	268	143	121															
19	18	19/12/2023	1482	307	327															
20	19	19/11/2023	1447	361	102															
21	20	16/08/2023	4007	349	56															
22	21	21/12/2023	2712	206	50															
23	22	07/11/2023	3895	167	138															
24	23	27/12/2023	1777	106	97															
25	24	21/08/2023	330	375	349															
26	25	20/02/2024	2085	17	21															

SalesProducts.csv - Excel

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	
1	SalesProductsID	SaleID	CustomerID	EmployeeID	ProductID	Quantity													
2	1	133	173	321	46	5													
3	2	342	304	190	91	8													
4	3	31	186	247	154	6													
5	4	89	232	290	375	6													
6	5	145	106	377	60	8													
7	6	330	335	80	27	3													
8	7	366	157	170	104	6													
9	8	350	56	137	187	4													
10	9	114	85	182	251	8													
11	10	5	307	394	104	10													
12	11	253	63	128	355	8													
13	12	123	4	217	26	3													
14	13	368	389	97	147	5													
15	14	309	42	34	336	4													
16	15	14	158	340	16	9													
17	16	247	230	116	194	4													
18	17	74	175	318	332	4													
19	18	25	211	360	261	3													
20	19	113	198	4	55	7													
21	20	397	232	70	321	10													
22	21	67	308	42	351	3													
23	22	310	124	233	308	2													
24	23	127	289	20	239	4													
25	24	39	216	324	399	1													
26	25	14	202	60	202	0													

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

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

