

# Community Loan Scheme Database Project Code First Girls Project Presentation

Isobel Olney

# Overview Of Database Purpose

Where I live, I have seen several community loan schemes startup.

These schemes allow people who face financial difficulties the ability to hire everyday items that they would otherwise not be able to utilise. For example - a lawnmower or pressure washer.

I thought it would be an interesting project to replicate this kind of loan scheme.

# Required Project Activities

I have completed the six fundamental activities and from the advanced requirements I have created:

- a sample stored procedure
- a sample view that uses three base tables
- an example query that uses GROUP BY

- Create relational DB of your choice with minimum 5 tables
- Set Primary and Foreign Key constraints to create relations between the tables
- Using any type of the joins create a view that combines multiple tables in a logical way
- In your database, create a stored function that can be applied to a query in your DB
- Prepare an example query with a subquery to demonstrate how to extract data from your DB for analysis
- Create DB diagram where all table relations are shown

## ADVANCED REQUIREMENTS

- In your database, create a stored procedure and demonstrate how it runs
- In your database, create a trigger and demonstrate how it runs
- In your database, create an event and demonstrate how it runs
- Create a view that uses at least 3-4 base tables; prepare and demonstrate a query that uses the view to produce a logically arranged result set for analysis.
- Prepare an example query with group by and having to demonstrate how to extract data from your DB for analysis

# Database Table Schema and Sample Data

```
1 • use community_loan_scheme ;  
2  
3 • select * from customers ;  
4 • select * from goods ;  
5 • select * from products ;  
◊ 26:3 |
```

Result Grid   Filter Rows:  Search   Edit: Export/Import:

id	name	email	phone_number	address	created_at	updated_at
1	Thomas Jones	thomas.jones@tjs.com	+44 163 356 789	79 Elm Street, Newport, NP10 1AA	2023-10-17 18:36:48	2023-10-17 18
2	Mary Evans	mary.evans@marys.com	+44 179 256 123	101 Birch Street, Swansea, SA1 1AA	2023-10-17 18:36:48	2023-10-17 18
3	David Williams	david.williams@dais.com	+44 120 293 445	12 Oak Street, Cardiff, CF10 1AA	2023-10-17 18:36:48	2023-10-17 18
4	Sarah Davies	sarah.davies@sarahdavies.com	+44 163 356 789	45 Elm Street, Newport, NP10 1AA	2023-10-17 18:36:48	2023-10-17 18
5	John Roberts	john.roberts@jrs.com	+44 120 2956 789	56 Ash Street, Cardiff, CF10 1AA	2023-10-17 18:36:48	2023-10-17 18
6	Susan Morgan	susan.morgan@morgans.com	+44 123 456 789	67 Conifer Road, Newport, NP10 1AA	2023-10-17 18:36:48	2023-10-17 18
7	David Lewis	david.lewis@davelewis.com	+44 123 456 789	30 Park Street, Newport, NP10 1AA	2023-10-17 18:36:48	2023-10-17 18
8	Anne Hughes	anne.hughes@hughesforrest.com	+44 129 206 789	89 Oak Street, Cardiff, CF10 1AA	2023-10-17 18:36:48	2023-10-17 18
9	Peter Griffiths	peter.griffiths@griffoes.com	+44 163 356 789	12 Birch Street, Newport, NP10 1AA	2023-10-17 18:36:48	2023-10-17 18
NULL	NULL	NULL	NULL	NULL	NULL	NULL

```
4 • select * from goods ;  
5 • select * from products ;  
◊ 22:4 |
```

Result Grid   Filter Rows:  Search   Edit:

id	product_...	status	created_at	updated_at
1	1	available	2023-10-17 18:51:45	2023-10-17 18:51:45
2	2	available	2023-10-17 18:51:45	2023-10-17 18:51:45
3	3	available	2023-10-17 18:51:45	2023-10-17 18:51:45
4	4	available	2023-10-17 18:51:45	2023-10-17 18:51:45
5	5	available	2023-10-17 18:51:45	2023-10-17 18:51:45
6	6	available	2023-10-17 18:51:45	2023-10-17 18:51:45
7	7	available	2023-10-17 18:51:45	2023-10-17 18:51:45
8	8	available	2023-10-17 18:51:45	2023-10-17 18:51:45
9	9	available	2023-10-17 18:51:45	2023-10-17 18:51:45
10	10	available	2023-10-17 18:51:45	2023-10-17 18:51:45
11	11	available	2023-10-17 18:51:45	2023-10-17 18:51:45
12	12	available	2023-10-17 18:51:45	2023-10-17 18:51:45
13	13	available	2023-10-17 18:51:45	2023-10-17 18:51:45
14	14	available	2023-10-17 18:51:45	2023-10-17 18:51:45
15	15	available	2023-10-17 18:51:45	2023-10-17 18:51:45
NULL	NULL	NULL	NULL	NULL

# Database Table Schema and Sample Data

```
6 • select * from loans ;
7 • select * from suppliers ;

Result Grid Filter Rows: Search Edit: Export/Import: 

id customer_id good_id loan_start_da... loan_end_date status created_at updated_at
1 1 1 2023-10-17 2023-10-31 active 2023-10-17 18:51:52 2023-10-17 18:51:52
2 2 2 2023-10-18 2023-10-30 active 2023-10-17 18:51:52 2023-10-17 18:51:52
3 3 3 2023-10-19 2023-10-29 active 2023-10-17 18:51:52 2023-10-17 18:51:52
4 4 4 2023-10-20 2023-10-28 active 2023-10-17 18:51:52 2023-10-17 18:51:52
5 5 5 2023-10-21 2023-10-27 active 2023-10-17 18:51:52 2023-10-17 18:51:52
6 6 6 2023-10-22 2023-10-26 active 2023-10-17 18:51:52 2023-10-17 18:51:52
7 7 7 2023-10-23 2023-10-25 active 2023-10-17 18:51:52 2023-10-17 18:51:52
8 8 8 2023-10-24 2023-10-24 active 2023-10-17 18:51:52 2023-10-17 18:51:52
9 9 9 2023-10-25 2023-10-23 active 2023-10-17 18:51:52 2023-10-17 18:51:52
10 1 2 2023-11-17 2023-11-30 active 2023-10-17 19:54:45 2023-10-17 19:54:45
11 1 3 2023-11-18 2023-11-30 active 2023-10-17 19:54:45 2023-10-17 19:54:45
12 3 4 2023-11-19 2023-11-29 active 2023-10-17 19:54:45 2023-10-17 19:54:45
13 3 6 2023-11-20 2023-11-28 active 2023-10-17 19:54:45 2023-10-17 19:54:45
14 3 7 2023-11-21 2023-11-27 active 2023-10-17 19:54:45 2023-10-17 19:54:45
15 6 5 2023-11-22 2023-11-26 active 2023-10-17 19:54:45 2023-10-17 19:54:45
16 6 3 2023-11-23 2023-11-25 active 2023-10-17 19:54:45 2023-10-17 19:54:45
17 8 2 2023-11-24 2023-11-24 active 2023-10-17 19:54:45 2023-10-17 19:54:45
18 9 1 2023-11-25 2023-11-26 active 2023-10-17 19:54:45 2023-10-17 19:54:45
19 1 2 2023-12-17 2023-12-24 pend... 2023-10-17 20:45:25 2023-10-17 20:45:25

Result Grid
```

5 \* select \* from products ;

25:5

**Result Grid** Filter Rows:  Search Edit: Export/Import:

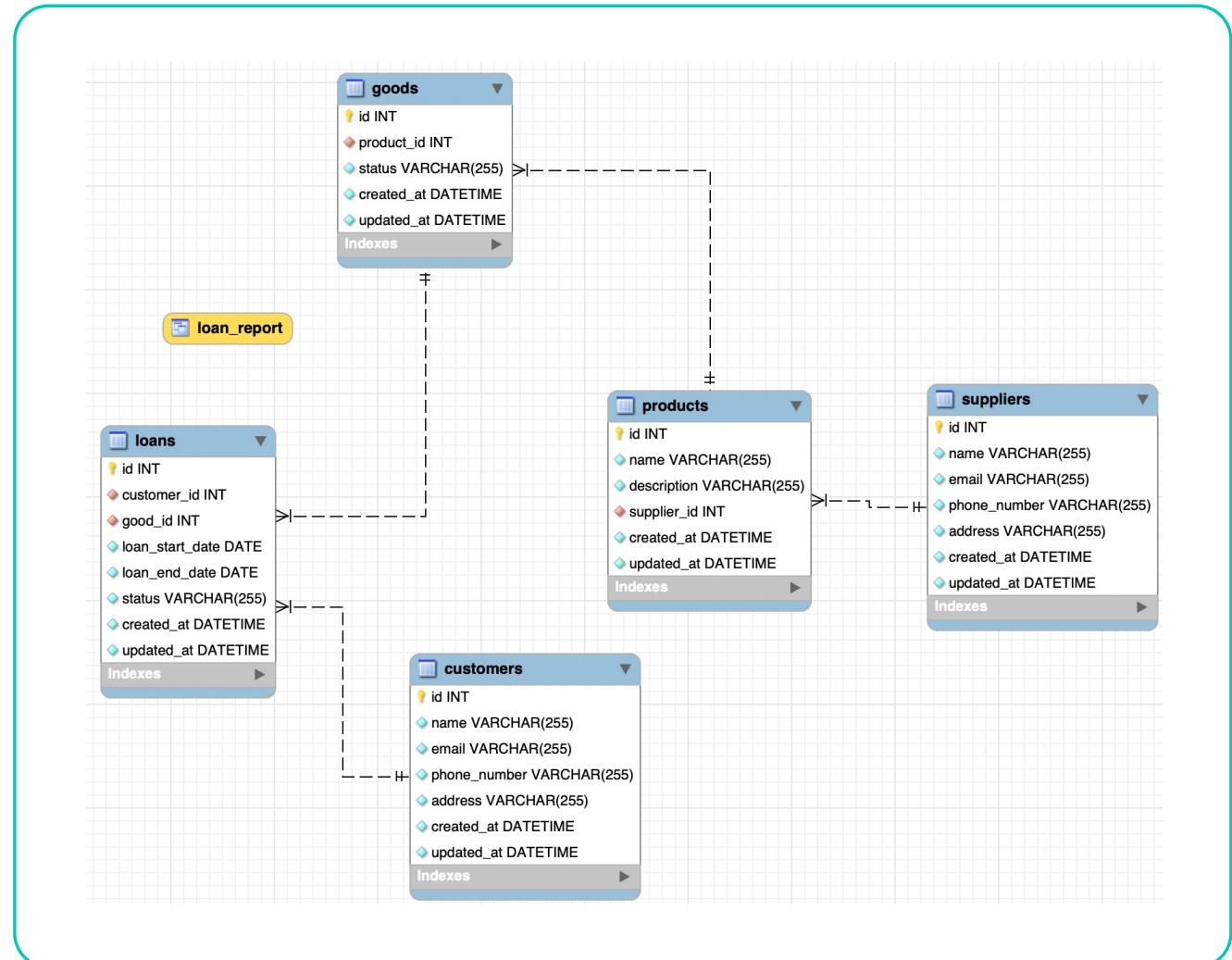
id	name	description	supplier_id	created_at	updated_at
1	Pressure washer	A high-powered water jet for cleaning driveways, patios, and other outdoor surfaces.	?	2023-10-17 18:36:02	2023-10-17 18:36:02
2	Hoover	A vacuum cleaner for cleaning floors and carpets.	1	2023-10-17 18:36:02	2023-10-17 18:36:02
3	Pram	A baby carriage for transporting a baby or toddler.	1	2023-10-17 18:36:02	2023-10-17 18:36:02
4	Toolkit	A set of tools for carrying out minor repairs and...	1	2023-10-17 18:36:02	2023-10-17 18:36:02
5	Lawnmower	A machine for cutting grass.	1	2023-10-17 18:36:02	2023-10-17 18:36:02
6	Hose	A flexible pipe for carrying water.	1	2023-10-17 18:36:02	2023-10-17 18:36:02
7	Hedge trimmer	A powered tool for trimming hedges and bushes.	1	2023-10-17 18:36:02	2023-10-17 18:36:02
8	Strimmers	A powered tool for trimming grass and other veg...	1	2023-10-17 18:36:02	2023-10-17 18:36:02
9	Tile cutter	A tool for cutting tiles.	1	2023-10-17 18:36:02	2023-10-17 18:36:02
10	Ladder	A portable structure that allows people to reach...	1	2023-10-17 18:36:02	2023-10-17 18:36:02
11	Paint sprayer	A tool for spraying paint onto surfaces.	1	2023-10-17 18:36:02	2023-10-17 18:36:02
12	Drills	A power tool for making holes in materials.	1	2023-10-17 18:36:02	2023-10-17 18:36:02
13	Saws	A tool for cutting materials.	1	2023-10-17 18:36:02	2023-10-17 18:36:02
14	Hammers	A tool for driving nails and other fasteners.	1	2023-10-17 18:36:02	2023-10-17 18:36:02
15	Screwdrivers	A tool for turning screws.	1	2023-10-17 18:36:02	2023-10-17 18:36:02
16	Tape measures	A tool for measuring distances.	1	2023-10-17 18:36:02	2023-10-17 18:36:02
17	Level	A tool for ensuring that surfaces are level.	1	2023-10-17 18:36:02	2023-10-17 18:36:02
NULL	NULL	NULL	NULL	NULL	NULL

from suppliers ;

Filter Rows:  Search Edit: Export/Import:

email	phone_number	address	created_at	updated_at
info@jonestools.co.uk	+44 123 456 789	123 Main Street, Cardiff, CF10 1AA	2023-10-17 18:34:18	2023-10-17 18:34:18
info@evanshire.co.uk	+44 789 456 123	456 Queen Street, Swansea, SA1 1AA	2023-10-17 18:34:18	2023-10-17 18:34:18
info@williamsrentals.co.uk	+44 123 456 789	890 High Street, Newport, NP10 1AA	2023-10-17 18:34:18	2023-10-17 18:34:18

# Schema Relational Diagram



# Some example queries

```
18 -- Get all loans for a specific customer
19 • SELECT * FROM loans WHERE customer_id = 3;
20
21
22
23
```

43:19

Result Grid Filter Rows: Search Edit: Export/Import:

id	customer_id	good_id	loan_start_da...	loan_end_date	status	created_at	updated_at
3	3	3	2023-10-19	2023-10-29	active	2023-10-17 18:51:52	2023-10-17 18:51:52
12	3	4	2023-11-19	2023-11-29	active	2023-10-17 19:54:45	2023-10-17 19:54:45
13	3	6	2023-11-20	2023-11-28	active	2023-10-17 19:54:45	2023-10-17 19:54:45
14	3	7	2023-11-21	2023-11-27	active	2023-10-17 19:54:45	2023-10-17 19:54:45
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

```
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
```

6:19

Result Grid Filter Rows: Search Edit: Export/Import:

id	customer_id	good_id	loan_start_da...	loan_end_date	status	created_at	updated_at
1	1	1	2023-10-17	2023-10-31	active	2023-10-17 18:51:52	2023-10-17 18:51:52
18	9	1	2023-11-25	2023-11-26	active	2023-10-17 19:54:45	2023-10-17 19:54:45
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

```
36 -- Get the total number of loans made to date
37 • SELECT COUNT(*) AS total_loans FROM loans;
38
```

48:34

Result Grid Filter Rows: Search Export:

tot...
19

```
-- Stored function, finds average loan duration for different goods
use community_loan_scheme ;

DELIMITER //

CREATE FUNCTION get_average_loan_duration_by_product(product_id INT)

RETURNS DECIMAL(10,2)

DETERMINISTIC

BEGIN
    DECLARE AVG_LOAN_DURATION DECIMAL(10,2) ;

    SELECT AVG(loan_end_date - loan_start_date) INTO AVG_LOAN_DURATION FROM loans WHERE good_id = product_id;

    RETURN AVG_LOAN_DURATION;
END //

DELIMITER ;

SELECT get_average_loan_duration_by_product(1);
```

The screenshot shows a MySQL command-line interface window. At the top, there is a status bar with the number 61, a blue dot, and the text "SELECT get\_average\_loan\_duration\_by\_product(1);". Below the status bar, the text "62" is visible. The main area contains the SQL query "SELECT get\_average\_loan\_duration\_by\_product(1);". At the bottom, there is a results grid with one row. The first column is labeled "get\_average\_loan\_duration\_by\_produ..." and the second column contains the value "7.50".

get_average_loan_duration_by_produ...	7.50

# Create a Stored Function

-- Creates a view linking 3 tables, with a query showing all loans due to be returned in the next week

- `CREATE VIEW loan_report AS`  
`SELECT`  
`customers.name AS customer_name,`  
`products.name AS product_name,`  
`loans.loan_start_date AS loan_start_date,`  
`loans.loan_end_date AS loan_end_date,`  
`loans.status AS loan_status`  
`FROM loans`  
`JOIN customers ON loans.customer_id = customers.id`  
`JOIN products ON loans.good_id = products.id;`
- `SELECT * FROM loan_report WHERE loan_end_date BETWEEN CURRENT_DATE AND CURRENT_DATE + INTERVAL 7 DAY;`

# Create A Database View

Result Grid Filter Rows:  Search Export:

	customer_name	product_name	loan_start_da...	loan_end_date	loan_status	
	David Lewis	Hedge trimmer	2023-10-23	2023-10-25	active	
	Anne Hughes	Strimmers	2023-10-24	2023-10-24	active	
	Peter Griffiths	Tile cutter	2023-10-25	2023-10-23	active	

# Stored Procedure

```
oo
89      -- Stored procedure, which is used to input details to create a new loan
90
91      DELIMITER //
92 * ⊖ CREATE PROCEDURE create_new_loan(
93          IN customer_id INT,
94          IN good_id INT,
95          IN loan_start_date DATE,
96          IN loan_end_date DATE
97      )
98      ⊖ BEGIN
99          INSERT INTO loans (customer_id, good_id, loan_start_date, loan_end_date, status)
100         VALUES (customer_id, good_id, loan_start_date, loan_end_date, 'pending');
101     END //
102     DELIMITER ;
103
104 *   CALL create_new_loan(1, 2, '2023-12-17', '2023-12-24');
105
106
```

# Subquery Using GROUP BY and ORDER BY

```
-- Query with a subquery that also contains a 'group and order by'  
-- It returns the names of the 4 customers who have taken out the most loans, ordered by the number of loans they have taken out.
```

```
SELECT customers.name, COUNT(*) AS loan_count  
FROM loans  
JOIN customers ON loans.customer_id = customers.id  
GROUP BY customers.name  
ORDER BY loan_count DESC  
LIMIT 4;
```

Result Grid    Filter Rows:    Search

name	loan_count
Thomas Jones	5
David Williams	4
Susan Morgan	3
Peter Griffiths	2