



# ShelfSpace

"A digital shelf for every reader."



# Introduction



The Online Book Store Database Project by Om Vedant is a structured SQL-based system designed to manage book inventory, customer information, and order processing. The database consists of three core entities:

- Books: Stores details such as title, author, genre, publication year, price, and stock availability.
- Customers: Maintains customer records including name, email, phone, city, and country.
- Orders: Tracks purchase transactions with order dates, quantities, and total amounts, linked to both customers and books.

Through the use of SQL queries, joins, constraints, and aggregate functions, the project ensures efficient data storage, retrieval, and analysis. It provides insights into sales performance, customer behavior, and inventory management, simulating the core functionalities of a real-world online bookstore.



# Objectives of the Project

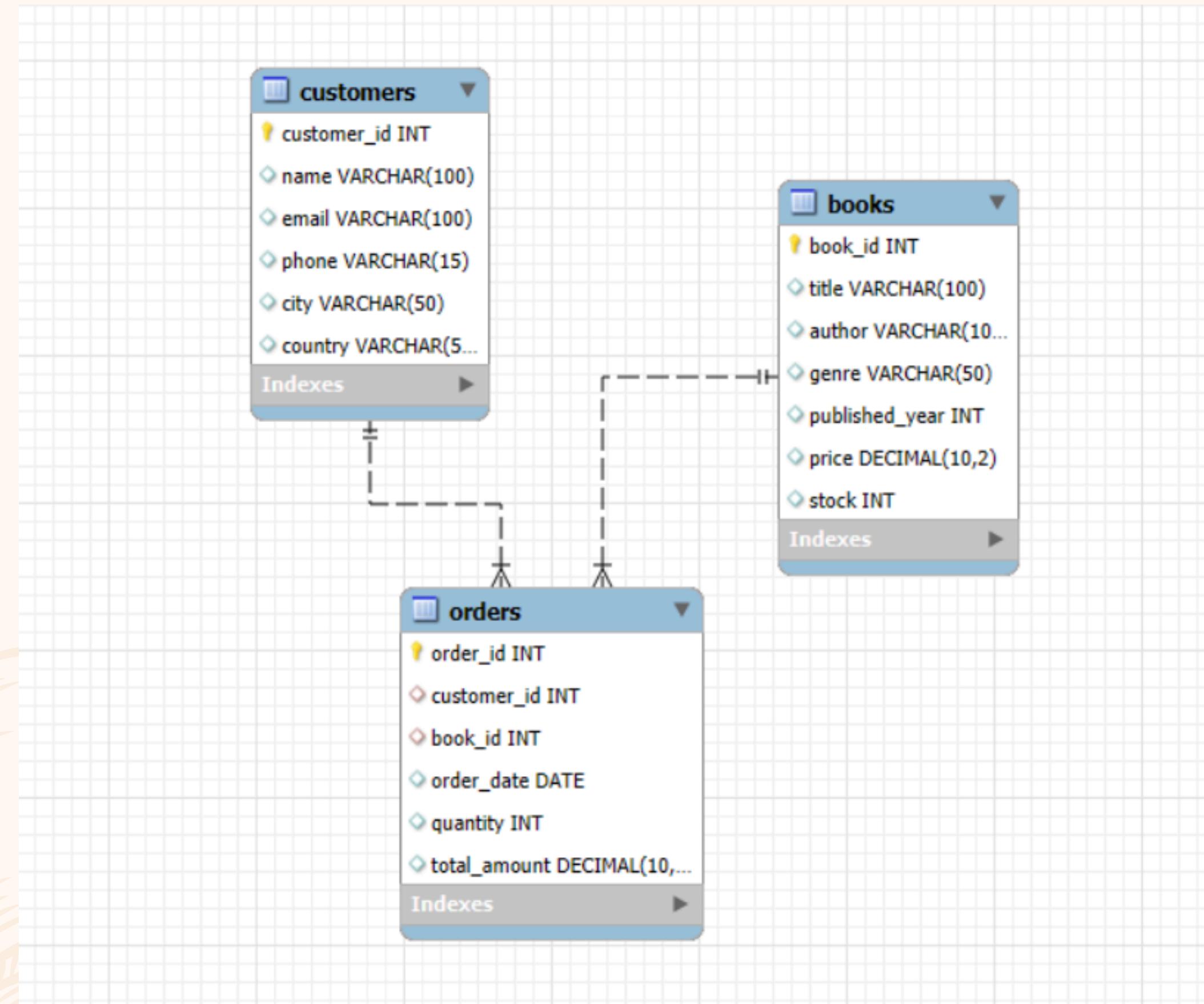


The main objectives of developing the Online Bookstore Database are:

1. Efficient Book Storage & Retrieval – To organize and manage book records systematically, ensuring quick and accurate access to information.
2. Manage Customer Data – To maintain customer details securely for smooth transactions and personalized services.
3. Handle Orders & Payments – To record, track, and process customer orders and payment details effectively.
4. Generate Reports – To create useful reports (such as sales, inventory, and customer activity) for better decision-making.



# ERD (Entity-Relationship Diagram)





# SQL Queries



```
1  create database online_bookstore;  
2  
3 • use online_bookstore;  
4  
5 • drop table if exists books;  
6 • create table books(  
7      book_id int auto_increment primary key,  
8      title varchar(100),  
9      author varchar(100),  
10     genre varchar(50),  
11     published_year int,  
12     price decimal(10,2),  
13     stock int  
14 );  
15
```





# SQL Queries



```
16 • drop table if exists customers;  
17 • create table customers(  
18     customer_id int auto_increment primary key,  
19     name varchar(100),  
20     email varchar(100),  
21     phone varchar(15),  
22     city varchar(50),  
23     country varchar(50)  
24 );  
25
```





# SQL Queries



```
26 • drop table if exists orders;  
27 •  create table orders(  
28     order_id int auto_increment primary key,  
29     customer_id int,  
30     book_id int,  
31     order_date date,  
32     quantity int,  
33     total_amount decimal(10,2),  
34     foreign key (customer_id) references customers(customer_id),  
35     foreign key (book_id) references books(book_id)  
36 );  
37  
38 • select * from books;  
39 • select * from customers;  
40 select * from orders;  
41
```



# Q1. Retrieve all books in the "Fictional" genre.

```
2
3 • select * from books
4 where genre = 'fiction';
5
6
```

Output :

book_id	title	author	genre	published_year	price	stock
4	Customizable 24hour product	Christopher Andrews	Fiction	2020	43.52	8
22	Multi-layered optimizing migration	Wesley Escobar	Fiction	1908	39.23	78
28	Expanded analyzing portal	Lisa Coffey	Fiction	1941	37.51	79
29	Quality-focused multi-tasking challenge	Katrina Underwood	Fiction	1905	31.12	100
31	Implemented encompassing conglomeration	Melissa Taylor	Fiction	2010	21.23	44
39	Optimized national process improvement	Megan Goodwin	Fiction	1978	10.99	42
40	Adaptive didactic interface	Natalie Gonzalez	Fiction	1923	25.97	94
47	Reverse-engineered directional conglomeration	John Christian	Fiction	2006	20.37	90
62	Re-contextualized real-time strategy	Nicole Lynch	Fiction	1953	26.34	23
63	Polarized heuristic database	Franklin Mack	Fiction	1989	22.38	56
100	Synchronized client-server service-desk	James Alvarado	Fiction	1906	49.89	29
116	Multi-tiered foreground contingency	Jamie Gates	Fiction	1938	41.82	50
125	Public-key analyzing Graphic Interface	Abigail Madden	Fiction	1990	32.41	16
130	Realigned context-sensitive pricing structure	Jason Rodriguez	Fiction	2004	6.64	90
134	Polarized bandwidth-monitored throughput	Linda Newman	Fiction	1955	35.72	49
142	Multi-tiered responsive parallelism	Amanda Wilson	Fiction	1940	48.96	11
143	Networked multimedia support	Nancy Goodman	Fiction	2012	43.65	50
144	Future-proofed scalable software	Matthew Wolf	Fiction	1945	19.27	90
156	Synergistic grid-enabled website	Brandon Black	Fiction	1953	31.68	34

book_id	title	author	genre	published_year	price	stock
173	Ergonomic foreground Graphic Interface	Patrick Williams	Fiction	1948	6.07	100
185	Networked object-oriented definition	Kelly Vazquez	Fiction	1940	21.98	10
187	Total well-modulated groupware	Kenneth Phillips	Fiction	1989	22.98	7
214	Fundamental methodical open architecture	Jeffery Green	Fiction	2005	18.16	47
220	Multi-lateral zero-defect matrix	Nicole Hughes	Fiction	1904	16.51	60
249	Fundamental national infrastructure	Nancy Austin	Fiction	1928	32.95	58
251	Seamless 5thgeneration challenge	Michelle Harris	Fiction	1906	8.87	98
252	Down-sized high-level encoding	Gabriela Russell MD	Fiction	1902	12.62	43
253	Horizontal didactic structure	Robert Rodriguez	Fiction	1961	11.66	21
260	Business-focused methodical application	Brian King	Fiction	1907	49.59	10
263	Right-sized bifurcated definition	Kayla Wilkinson	Fiction	1965	9.79	3
274	Automated systemic functionalities	Matthew Thomas	Fiction	1903	48.80	72
278	Exclusive asymmetric installation	Benjamin Flores	Fiction	1901	48.61	69
297	Advanced well-modulated architecture	Chad Craig	Fiction	2016	35.65	2
298	Assimilated composite archive	Mark Gibson	Fiction	1957	46.66	62
299	Centralized disintermediate array	Donald Wiggins	Fiction	1909	28.19	73
302	Ergonomic leading-edge portal	James Curtis	Fiction	1981	28.67	92
315	Virtual exuding Internet solution	Anne George	Fiction	2012	42.90	40
324	Diverse dynamic contingency	James Duran	Fiction	1993	46.08	23

book_id	title	author	genre	published_year	price	stock
327	Implemented zero-defect product	Kim Lee	Fiction	1984	21.40	29
329	Multi-tiered dedicated encryption	Jacob Fischer	Fiction	1973	38.49	27
331	Profit-focused systematic project	Dennis Hooper	Fiction	1936	15.36	89
342	Digitized regional monitoring	Kenneth Finley	Fiction	1969	17.11	84
362	Profit-focused multimedia throughput	Andrea Lucas	Fiction	1933	29.43	64
366	Re-contextualized national secured line	Robert Wilson	Fiction	1922	13.10	4
367	Universal analyzing support	Eric Taylor	Fiction	1940	29.32	35
372	Realigned mobile focus group	Michael Jackson	Fiction	1905	17.73	42
373	Exclusive dynamic approach	Michelle Mercado	Fiction	1911	39.09	49
403	Persevering incremental strategy	David Miller	Fiction	1928	15.01	82
406	De-engineered methodical capacity	James Gonzalez	Fiction	2006	35.79	50
408	Decentralized tertiary customer loyalty	William Roberts	Fiction	1946	10.07	2
414	Organized directional Internet solution	Amanda Carrillo	Fiction	1999	22.55	93
421	Exclusive zero administration superstructure	John Pearson	Fiction	1901	40.05	58
437	Customizable directional matrices	Alyssa Lopez	Fiction	1960	44.29	7
439	Monitored eco-centric framework	Kelly Rodriguez	Fiction	1970	16.00	94
458	Cross-group analyzing matrix	Nancy Griffith	Fiction	1914	37.28	53
472	Stand-alone clear-thinking extranet	Craig Dominguez	Fiction	1942	28.09	9
479	Balanced grid-enabled architecture	Christian Morales	Fiction	2009	38.27	60

## Q2. Find books published after the year 1950.

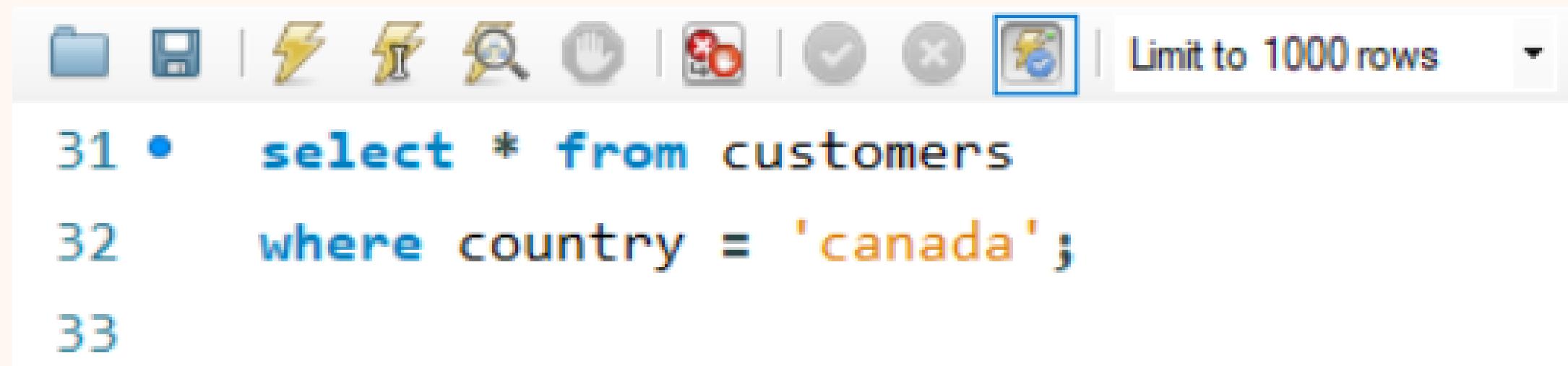
```
17 • select * from books
18 where published_year > 1950;
19
20
```

Output :

book_id	title	author	genre	published_year	price	stock
2	Persevering reciprocal knowledge user	Mario Moore	Fantasy	1971	35.80	19
4	Customizable 24hour product	Christopher Andrews	Fiction	2020	43.52	8
5	Adaptive 5thgeneration encoding	Juan Miller	Fantasy	1956	10.95	16
6	Advanced encompassing implementation	Bryan Morgan	Biography	1985	6.56	2
8	Persistent local encoding	Troy Cox	Science Fiction	2019	48.99	84
9	Optimized interactive challenge	Colin Buckley	Fantasy	1987	14.33	70
10	Ergonomic national hub	Samantha Ruiz	Mystery	2015	24.63	25
11	Secured zero tolerance time-frame	Denise Barnes	Fantasy	1998	35.95	10
12	Polarized optimal array	Destiny Scott	Non-Fiction	1989	27.43	63
15	User-friendly motivating strategy	Keith Smith	Non-Fiction	1997	23.83	58
17	Reduced secondary core	Benjamin Peters	Fantasy	1966	5.37	45
18	Adaptive 4thgeneration concept	Hector Palmer	Non-Fiction	2021	39.47	32
19	Progressive asymmetric Internet solution	Sean Miller	Science Fiction	1990	11.31	1
20	Face-to-face systematic throughput	Teresa Brennan	Non-Fiction	1978	48.13	64
23	Reverse-engineered context-sensitive ...	Christina Hernandez	Mystery	1967	38.55	70
25	Devolved mobile conglomeration	Alexander Bailey	Biography	1984	8.55	79
26	Multi-channelled multi-tasking capability	Patricia Buck	Science Fiction	1964	21.05	41
30	Multi-layered global open system	Jose Meyer	Biography	2012	30.58	37
31	Implemented encompassing conglomera...	Melissa Taylor	Fiction	2010	21.23	44

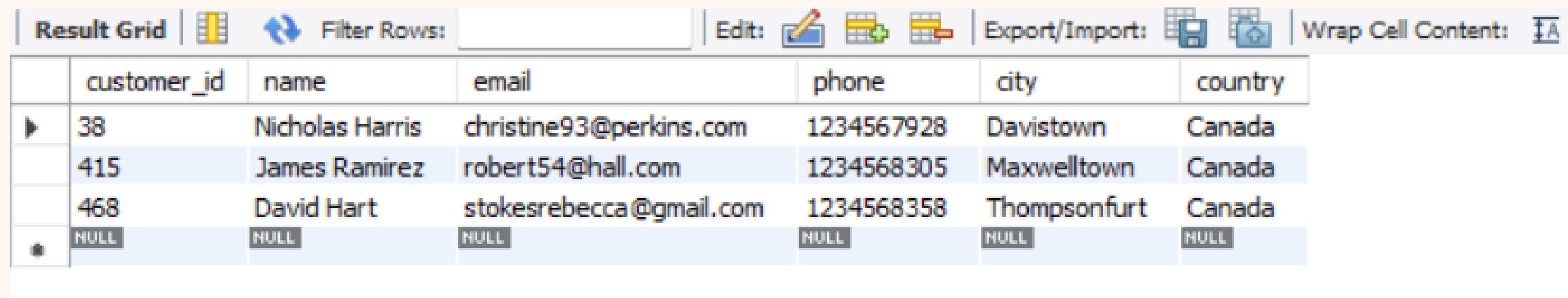
book_id	title	author	genre	published_year	price	stock
32	Synergistic dedicated concept	Lisa Bailey	Mystery	2021	21.56	100
33	Team-oriented empowering synergy	Sandra Williams	Non-Fiction	2023	35.06	49
35	Grass-roots solution-oriented adapter	William Turner	Non-Fiction	2018	31.81	85
37	Up-sized tertiary archive	Todd Kennedy	Fantasy	1955	13.08	3
38	Pre-emptive asynchronous leverage	Courtney Nichols	Fantasy	2022	16.47	72
39	Optimized national process improvement	Megan Goodwin	Fiction	1978	10.99	42
43	Function-based zero-defect initiative	Daniel Nunez	Romance	1952	47.39	61
44	Networked systemic implementation	Ryan Frank	Science Fiction	1965	13.55	0
45	Centralized cohesive website	Luis Rogers	Mystery	1984	25.49	59
47	Reverse-engineered directional congl...	John Christian	Fiction	2006	20.37	90
48	Multi-channelled 5thgeneration Internet...	Jennifer Powell	Biography	1963	24.70	94
49	Robust attitude-oriented attitude	Zachary Hayes	Biography	1955	49.50	15
50	Face-to-face responsive secured line	Lindsey Rodriguez	Fantasy	1970	37.00	63
51	Visionary optimizing project	Douglas Bush	Biography	2022	46.30	95
52	Triple-buffered neural adapter	Kristi Phillips	Science Fiction	2003	16.53	85
53	Triple-buffered multi-tasking help-desk	Brian Johnson	Biography	1969	29.28	60
54	Progressive bi-directional methodology	Adrian Taylor	Fantasy	2015	44.78	27
56	Integrated intermediate Internet solution	Stacy Cabrera	Mystery	1976	7.87	67
57	Team-oriented 3rdgeneration encryption	Sabrina Hernandez ...	Romance	1962	23.59	27

Q3. List all the customers from Canada.



```
31 • select * from customers;
32 where country = 'canada';
33
```

Output :



	customer_id	name	email	phone	city	country
▶	38	Nicholas Harris	christine93@perkins.com	1234567928	Davistown	Canada
	415	James Ramirez	robert54@hall.com	1234568305	Maxwelltown	Canada
	468	David Hart	stakesrebecca@gmail.com	1234568358	Thompsonfurt	Canada
●	NULL	NULL	NULL	NULL	NULL	NULL

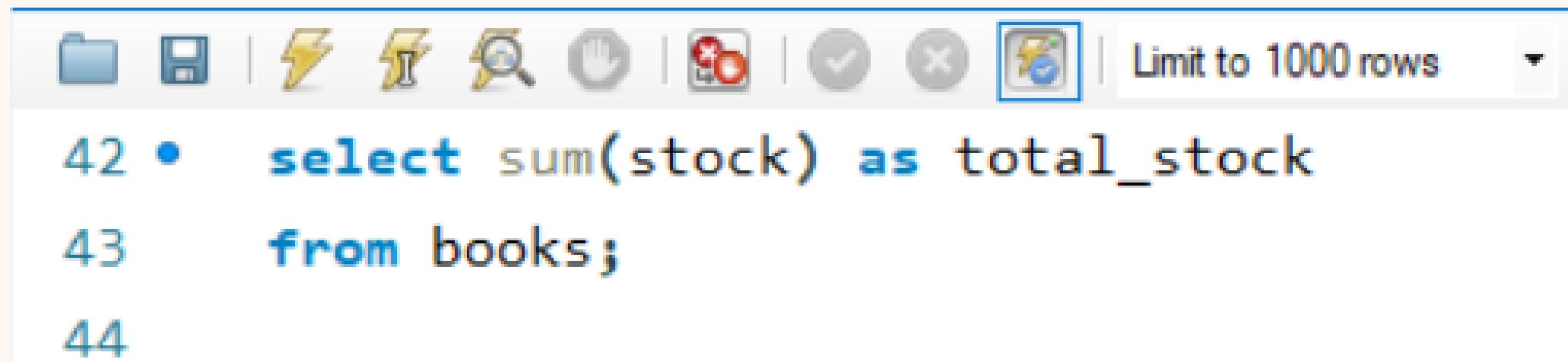
## Q4. Show orders placed in November 2023.

```
36 • select * from orders
37 where order_date between '2023-11-01' and '2023-11-30';
38
39
```

Output :

	order_id	customer_id	book_id	order_date	quantity	total_amount
▶	4	433	343	2023-11-25	7	301.21
	19	496	60	2023-11-17	9	316.26
	75	291	375	2023-11-30	5	170.75
	132	469	333	2023-11-22	7	194.32
	137	474	471	2023-11-25	8	363.04
	163	207	384	2023-11-23	3	101.76
	182	129	293	2023-11-01	7	125.51
	200	313	303	2023-11-23	1	6.57
	213	325	447	2023-11-17	7	253.75
	231	22	384	2023-11-11	1	33.92
	245	386	97	2023-11-01	9	411.66
	252	405	387	2023-11-15	5	237.10
	257	123	403	2023-11-06	1	15.01
	288	6	128	2023-11-13	1	24.04
	307	368	133	2023-11-17	1	20.96
	322	270	112	2023-11-08	2	16.04
	344	385	218	2023-11-25	5	26.80
	389	485	391	2023-11-18	2	66.84
	414	23	234	2023-11-10	1	7.15

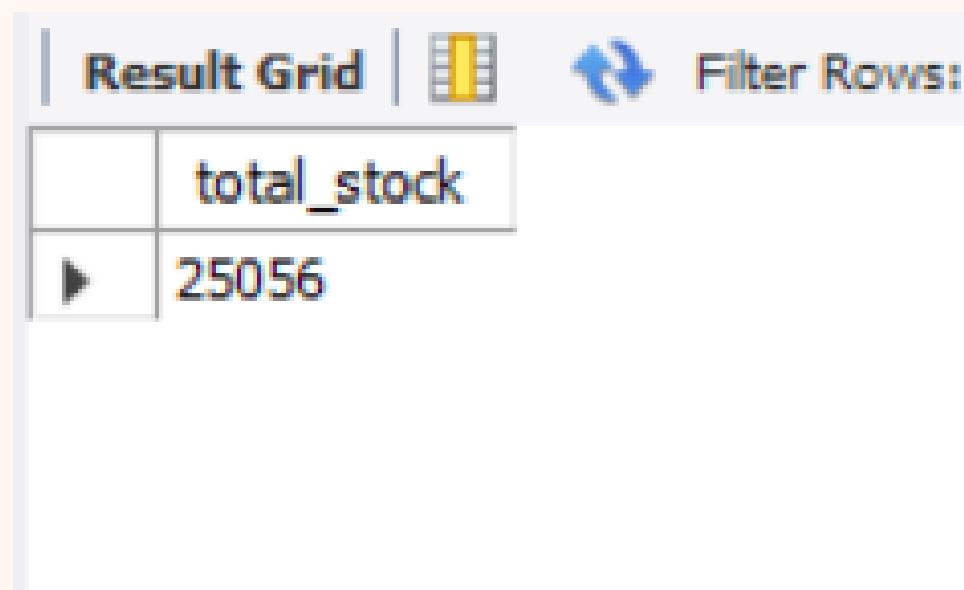
Q5. Retrieve the total stock of books available.



The screenshot shows a MySQL query editor interface. At the top, there is a toolbar with various icons: a folder, a table, a lightning bolt, a refresh, a magnifying glass, a hand, a checkmark, a cross, and a refresh with a gear. To the right of these icons is a dropdown menu labeled "Limit to 1000 rows". Below the toolbar, the query text is displayed in three numbered lines:

```
42 • select sum(stock) as total_stock  
43 from books;  
44
```

Output :



The screenshot shows the results of the query execution. At the top, there are two tabs: "Result Grid" and "Filter Rows:". Below the tabs is a table with one row. The table has two columns: the first column is empty, and the second column is labeled "total\_stock" with the value "25056".

	total_stock
▶	25056

**Q6. Find the details of the most expensive book.**

48 • `select * from books`  
49       `order by price desc limit 1;`  
50  
51

# Output :

Q7. Show all the customers who ordered more than 1 quantity of books.

Query 1 SQL File 3\* ×

57 • `select * from orders`

58     `where quantity > 1;`

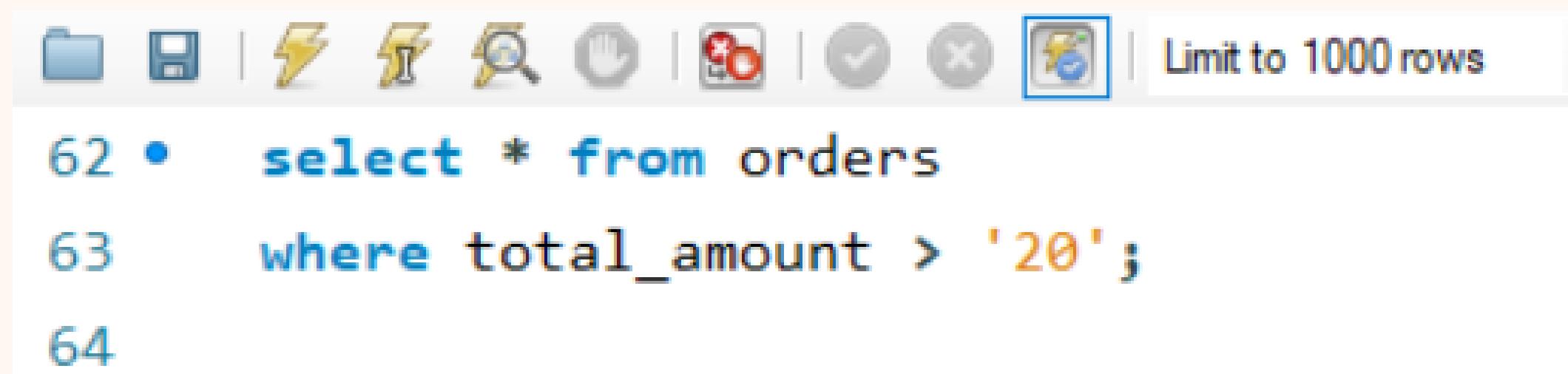
59

Output :

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

	order_id	customer_id	book_id	order_date	quantity	total_amount
▶	1	84	169	2023-05-26	8	188.56
	2	137	301	2023-01-23	10	216.60
	3	216	261	2024-05-27	6	85.50
	4	433	343	2023-11-25	7	301.21
	5	14	431	2023-07-26	7	136.36
	6	439	119	2024-10-11	5	249.40
	7	195	467	2023-10-23	6	82.92
	8	32	159	2024-05-07	4	144.84
	9	109	407	2024-01-04	9	379.71
	10	94	122	2024-07-09	4	123.00
	12	454	3	2024-06-17	2	31.50
	13	420	180	2023-06-08	5	125.45
	14	454	319	2023-08-24	2	85.22
	15	127	479	2023-01-10	6	229.62
	16	412	196	2023-10-06	8	53.52
	17	462	481	2023-03-20	5	52.75
	18	377	101	2024-08-07	4	193.96
	19	496	60	2023-11-17	9	316.26
	21	356	287	2024-12-05	3	120.57

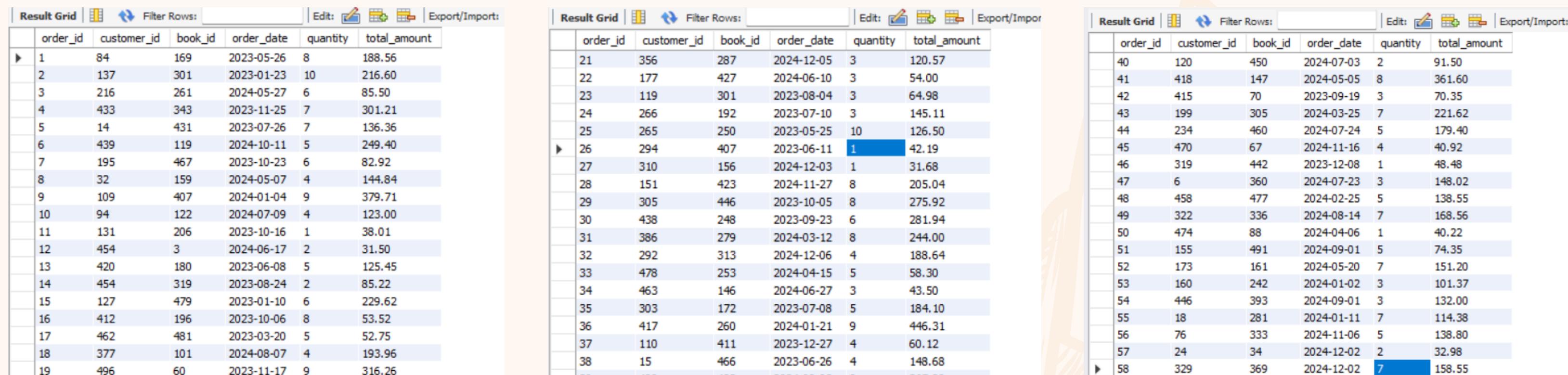
## Q8. Retrieve all orders where the total amount exceeds \$20.



The screenshot shows a MySQL query editor interface. At the top, there are several icons: a folder, a document, a lightning bolt, a refresh, a magnifying glass, a hand, a red circle with a minus sign, and a blue square with a checkmark. To the right of these is a dropdown menu labeled "Limit to 1000 rows". Below the toolbar, the query is displayed in two parts:

```
62 • select * from orders
63 where total_amount > '20';
64
```

Output :



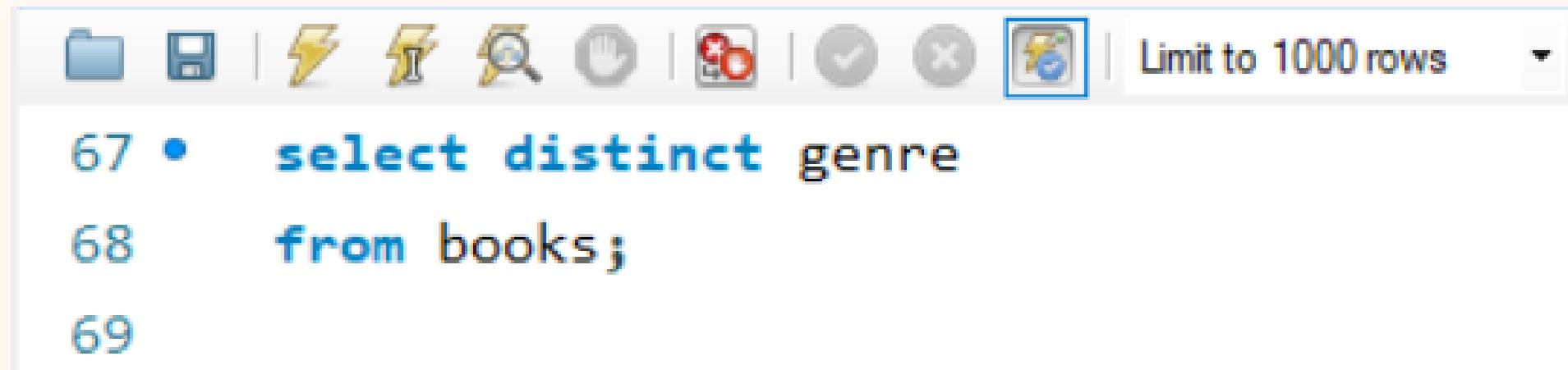
The image displays three separate screenshots of the MySQL query results grid, each showing a table with columns: order\_id, customer\_id, book\_id, order\_date, quantity, and total\_amount. The data in each grid is as follows:

	order_id	customer_id	book_id	order_date	quantity	total_amount
1	84	169	2023-05-26	8	188.56	
2	137	301	2023-01-23	10	216.60	
3	216	261	2024-05-27	6	85.50	
4	433	343	2023-11-25	7	301.21	
5	14	431	2023-07-26	7	136.36	
6	439	119	2024-10-11	5	249.40	
7	195	467	2023-10-23	6	82.92	
8	32	159	2024-05-07	4	144.84	
9	109	407	2024-01-04	9	379.71	
10	94	122	2024-07-09	4	123.00	
11	131	206	2023-10-16	1	38.01	
12	454	3	2024-06-17	2	31.50	
13	420	180	2023-06-08	5	125.45	
14	454	319	2023-08-24	2	85.22	
15	127	479	2023-01-10	6	229.62	
16	412	196	2023-10-06	8	53.52	
17	462	481	2023-03-20	5	52.75	
18	377	101	2024-08-07	4	193.96	
19	496	60	2023-11-17	9	316.26	

	order_id	customer_id	book_id	order_date	quantity	total_amount
21	356	287	2024-12-05	3	120.57	
22	177	427	2024-06-10	3	54.00	
23	119	301	2023-08-04	3	64.98	
24	266	192	2023-07-10	3	145.11	
25	265	250	2023-05-25	10	126.50	
26	294	407	2023-06-11	1	42.19	
27	310	156	2024-12-03	1	31.68	
28	151	423	2024-11-27	8	205.04	
29	305	446	2023-10-05	8	275.92	
30	438	248	2023-09-23	6	281.94	
31	386	279	2024-03-12	8	244.00	
32	292	313	2024-12-06	4	188.64	
33	478	253	2024-04-15	5	58.30	
34	463	146	2024-06-27	3	43.50	
35	303	172	2023-07-08	5	184.10	
36	417	260	2024-01-21	9	446.31	
37	110	411	2023-12-27	4	60.12	
38	15	466	2023-06-26	4	148.68	

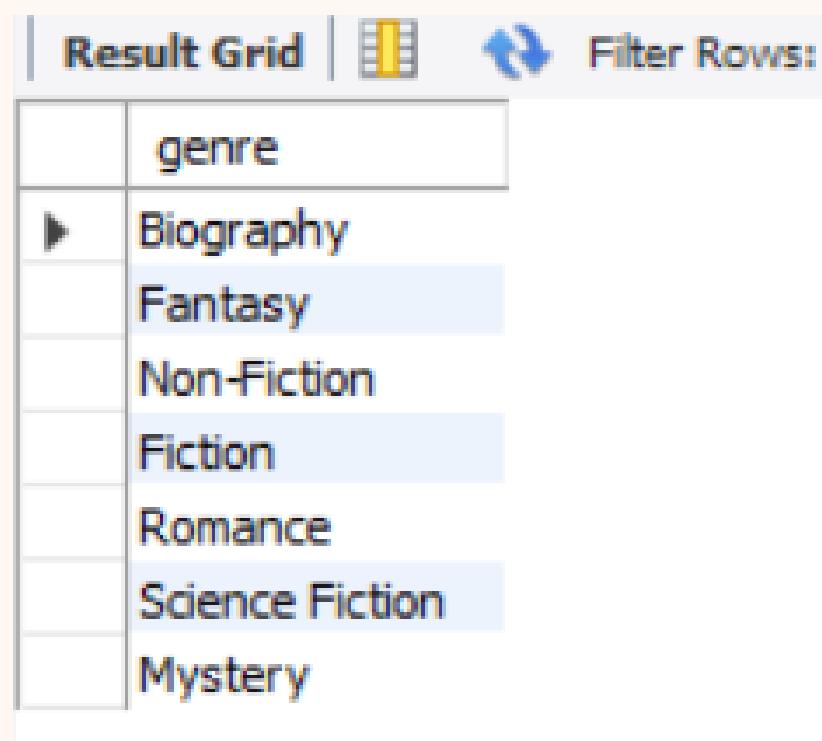
	order_id	customer_id	book_id	order_date	quantity	total_amount
40	120	450	2024-07-03	2	91.50	
41	418	147	2024-05-05	8	361.60	
42	415	70	2023-09-19	3	70.35	
43	199	305	2024-03-25	7	221.62	
44	234	460	2024-07-24	5	179.40	
45	470	67	2024-11-16	4	40.92	
46	319	442	2023-12-08	1	48.48	
47	6	360	2024-07-23	3	148.02	
48	458	477	2024-02-25	5	138.55	
49	322	336	2024-08-14	7	168.56	
50	474	88	2024-04-06	1	40.22	
51	155	491	2024-09-01	5	74.35	
52	173	161	2024-05-20	7	151.20	
53	160	242	2024-01-02	3	101.37	
54	446	393	2024-09-01	3	132.00	
55	18	281	2024-01-11	7	114.38	
56	76	333	2024-11-06	5	138.80	
57	24	34	2024-12-02	2	32.98	
58	329	369	2024-12-02	7	158.55	

Q9. List all genres available in the books table.



```
67 • select distinct genre
68 from books;
69
```

Output :

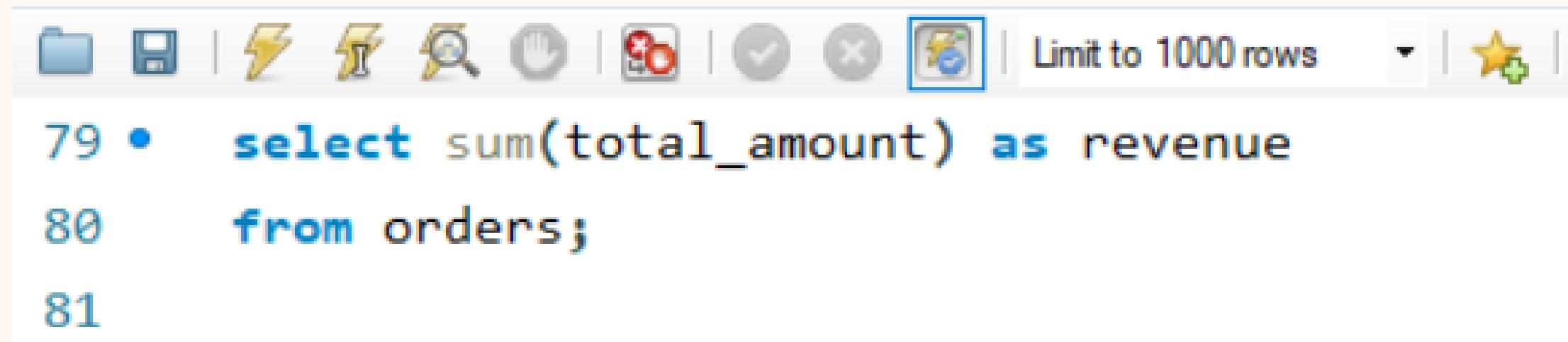


Result Grid | Filter Rows:

	genre
▶	Biography
	Fantasy
	Non-Fiction
	Fiction
	Romance
	Science Fiction
	Mystery



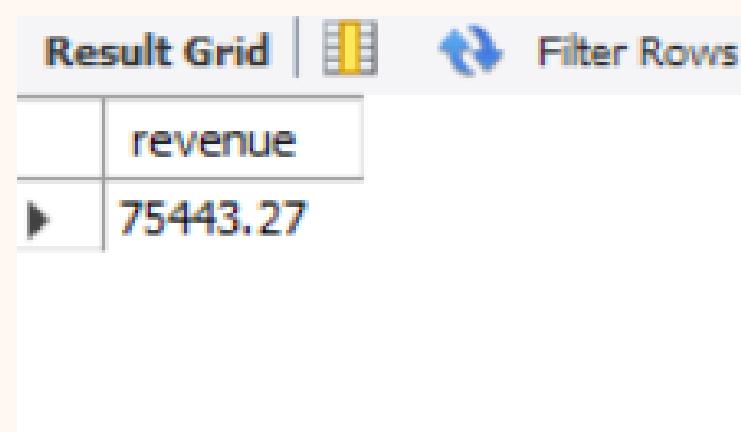
Q11. Calculate the total revenue generated from all orders.



The screenshot shows a MySQL query editor interface. At the top, there is a toolbar with various icons: folder, table, lightning bolt, refresh, magnifying glass, circular arrow, error, checkmark, close, and a highlighted gear icon. To the right of the toolbar are buttons for "Limit to 1000 rows" and a star with a plus sign. Below the toolbar, the SQL code is displayed:

```
79 • select sum(total_amount) as revenue  
80   from orders;  
81
```

Output :



The screenshot shows the results of the query execution. At the top, it says "Result Grid" and "Filter Rows:". Below is a table with one row:

revenue
75443.27

Q12. Retrieve the total number of books sold for each genre.

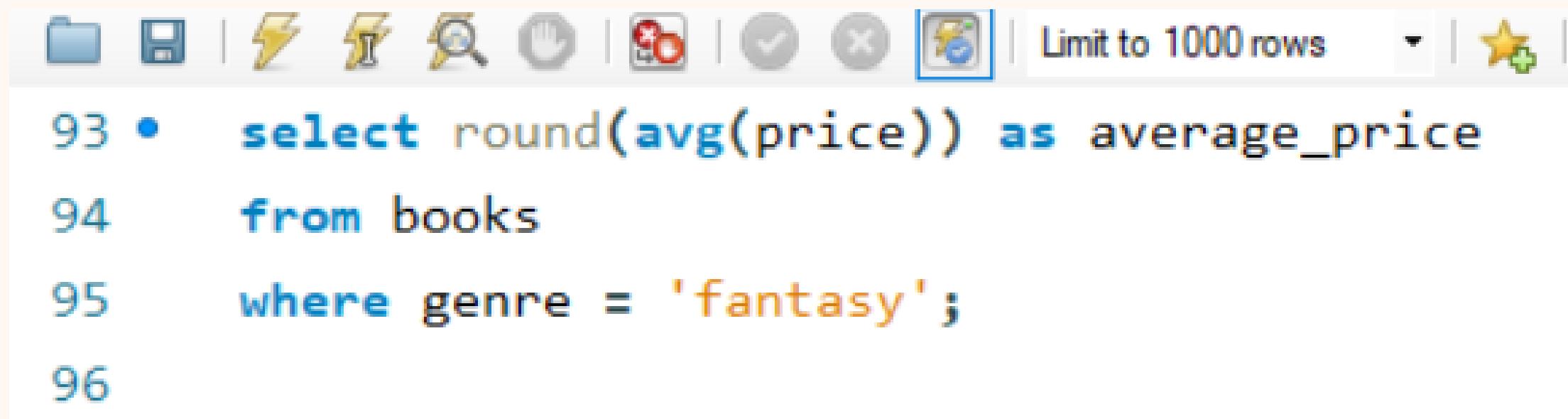
```
| File | New | Open | Save | Print | Find | Replace | Delete | Run | Limit to 1000 rows |  
85 • select b.genre,  
86     sum(o.quantity) as total_books_sold  
87     from books b join orders o  
88     on b.book_id = o.book_id  
89     group by b.genre;  
90
```

Output :

Result Grid | Filter Rows:

	genre	total_books_sold
▶	Biography	284
	Fantasy	446
	Science Fiction	447
	Mystery	504
	Romance	433
	Non-Fiction	351
	Fiction	225

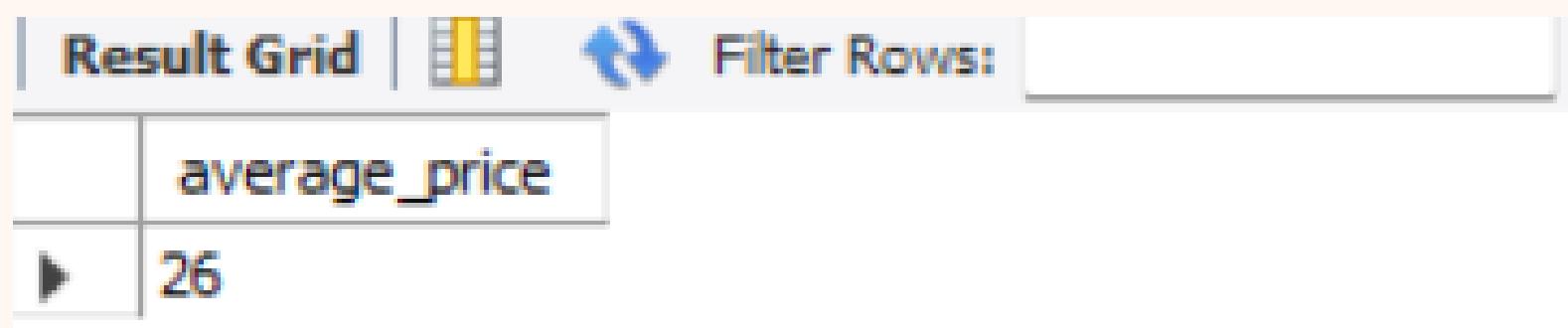
Q13. Find the average price of books in the "Fantasy" genre.



The screenshot shows a MySQL query editor interface. At the top, there are several icons: a folder, a file, a lightning bolt, a refresh, a magnifying glass, a hand, a red circle with a minus sign, a checkmark, a close button, and a gear icon. To the right of these is a dropdown menu labeled "Limit to 1000 rows". Below the toolbar, the query code is displayed:

```
93 • select round(avg(price)) as average_price  
94 from books  
95 where genre = 'fantasy';  
96
```

Output :



The screenshot shows the results of the query execution. At the top, there are buttons for "Result Grid" (highlighted in blue) and "Filter Rows:" followed by a search bar. The result grid itself has a header row with the column name "average\_price" and a data row containing the value "26".

average_price
26

## Q14. List the customers who have placed atleast 2 orders.

```
99 • select customer_id,  
100    count(order_id) as order_count  
101   from orders  
102   group by customer_id  
103   having count(order_id) >= 2;  
104
```

Output :

Result Grid		
	customer_id	order_count
▶	2	2
	6	2
	8	2
	13	2
	14	2
	15	2
	16	3
	21	2
	22	3
	23	2
	24	2
	37	2
	47	3
	54	2
	57	2
	59	3
	75	2
	81	2
	82	2

Result Grid		
	customer_id	order_count
▶	84	2
	94	3
	96	2
	98	3
	102	2
	103	2
	107	4
	109	2
	115	2
	117	2
	119	3
	120	2
	121	2
	125	2
	131	2
	134	2
	136	2
	137	2
▶	140	2

Result Grid		
	customer_id	order_count
▶	145	2
	149	3
	153	2
	154	2
	155	2
	158	2
	159	2
	160	2
	162	2
	163	2
	165	2
	166	3
	167	3
	173	2
	174	4
	177	2
	178	2
	181	2
▶	183	2

Result Grid		
	customer_id	order_count
▶	184	2
	189	2
	194	3
	195	3
	198	2
	199	2
	207	2
	214	3
	216	2
	225	2
	226	2
	232	2
	239	2
	260	2
	265	3
	272	3
	277	2
	279	2
▶	280	2

## Q15. Find the most frequently ordered book.

```
107 • select orders.book_id, books.title,  
108     count(orders.order_id) as order_count  
109     from orders join books  
110     on orders.book_id = books.book_id  
111     group by orders.book_id, books.title  
112     order by order_count desc limit 1;  
113
```

Output :

	book_id	title	order_count
▶	31	Implemented encompassing conglomeration	4

Q16. Show the top most expensive books of 'fantasy' genre.

```
116 • select * from books  
117 where genre = 'fantasy'  
118 order by price desc limit 3;  
119
```

# Output :

## Q17. Retrieve the total quantity of books sold by each author.

```
122 •    select b.author,  
123        sum(o.quantity) as total_books_sold  
124        from orders o join books b  
125        on o.book_id = b.book_id  
126        group by b.author;  
127
```

Output :

author	total_books_sold
Margaret Moore	8
John Davidson	13
Christopher Fuentes	6
Marissa Smith	16
Christopher Dixon	15
Tonya Saunders	21
Larry Hunt	6
Brandon Foster	4
Michelle Bell	11
Mary French	14
Lisa Lopez	1
Derrick Howard	5
David Rodriguez	5
Michael Mckenzie	12
Christian Morales	12
Rachel Gibbs	22
Nicole Smith	5
Amanda Knight	10
Brian Haney	9

author	total_books_sold
Michelle Taylor	5
Cynthia Diaz	3
Denise McLaughlin	9
Nicholas Roth	3
Jessica Lopez	13
Brandon Black	6
Daisy Johnson	9
Carolyn Schmidt	8
Cathy Knight	6
Elizabeth Wilson	13
Nicole Olson	7
Robert Rodriguez	7
Ethan Kennedy	17
Dennis Smith	5
Brian King	9
Aaron Hall Jr.	4
Michelle Hanson	18
Joyce Patton	19
Miguel Smith	9

## Q18. List the cities where customers who spend over \$30 are located.

```
131 • select distinct c.city, country, total_amount  
132   from orders o join customers c  
133   on c.customer_id = o.customer_id  
134   where o.total_amount > 30;  
135
```

Output :

Result Grid			
city	country	total_amount	Export:
Lake Paul	Armenia	188.56	
North Keith	Papua New Guinea	216.60	
Kelseyfort	Micronesia	85.50	
East David	Zambia	301.21	
Richardsonville	New Caledonia	136.36	
Ramosstad	Liechtenstein	249.40	
Rogersborough	Djibouti	82.92	
New Carlosbury	Malta	144.84	
Ravenberg	Macao	379.71	
West Anthony	New Zealand	123.00	
North Carolyn	Jersey	38.01	
Micheleborough	Guyana	31.50	
North Joseph	Western Sahara	125.45	
Micheleborough	Guyana	85.22	
Lake Mary	Reunion	229.62	
Lake Karen	Hungary	53.52	
Chandlerberg	Mongolia	52.75	
Austinfort	Montenegro	193.96	
Davidview	Netherlands Antilles	316.26	
Result Grid			
Evanshaven	Bahamas	120.57	
Kaylatown	Guyana	54.00	
Lake Victormouth	Serbia	64.98	
Ortizfurt	Falkland Islands (Malvinas)	145.11	
Port Erinberg	Iraq	126.50	
Angelaside	Mali	42.19	
North Emily	Libyan Arab Jamahiriya	31.68	
Lake Anthony	Belgium	205.04	
Perrytown	New Caledonia	275.92	
North Stephen...	Tajikistan	281.94	
East Richardbu...	Yemen	244.00	
Lambertfort	Armenia	188.64	
Joshuabury	Grenada	58.30	
East Lindsey	Saint Kitts and Nevis	43.50	
East Erikatown	Iran	184.10	
Erikaberg	Bahamas	446.31	
Chelseachester	Serbia	60.12	
Rodriguezmouth	Wallis and Futuna	148.68	
North Brian	Heard Island and McDonald Islands	367.28	
Result Grid			
Lake Eric	Zimbabwe	91.50	
Brandimouth	Belarus	361.60	
Maxwelltown	Canada	70.35	
East Sandra	Micronesia	221.62	
Nicoleshire	Djibouti	179.40	
Lake Melissa	Turkey	40.92	
Port Justin	Bahrain	48.48	
Hamiltonstad	Rwanda	148.02	
West Amanda	Cuba	138.55	
Sanchezport	Mexico	168.56	
East Chelsea	Cook Islands	40.22	
South Mariaside	Turks and Caicos Islands	74.35	
Annhaven	French Polynesia	151.20	
Lake Sarahside	Finland	101.37	
East Robert	Zimbabwe	132.00	
New Joseptown	Equatorial Guinea	114.38	
Janicemouth	Turkey	138.80	
Bridgetown	Trinidad and Tobago	32.98	
Marshallbury	Estonia	158.55	
Result Grid			
Port Amandah...	Jersey	140.98	
Lake Tyler	Korea	445.50	
Freemanland	Vanuatu	198.75	
Kennethland	Hungary	230.40	
East Michaelfurt	Saint Helena	209.25	
Gentryfort	Mozambique	344.80	
Brendafurt	Central African Republic	201.42	
New Roberto...	Niger	286.64	
West Maria	Swaziland	366.66	
Angelastad	Dominican Republic	426.10	
West Justin	Slovakia (Slovak Republic)	34.16	
Jonathanberg	Heard Island and McDonald Islands	217.50	
Port Angela	Moldova	267.12	
Lake Aaronstad	Micronesia	44.19	
Port Davidhaven	Denmark	204.60	
Alanton	Russian Federation	170.75	
Lake Melissa	Turkey	371.70	
West Billymouth	Côte d'Ivoire	236.53	
Conniefort	Lesotho	33.45	

## Q19. Find the customers who spend the most on orders.

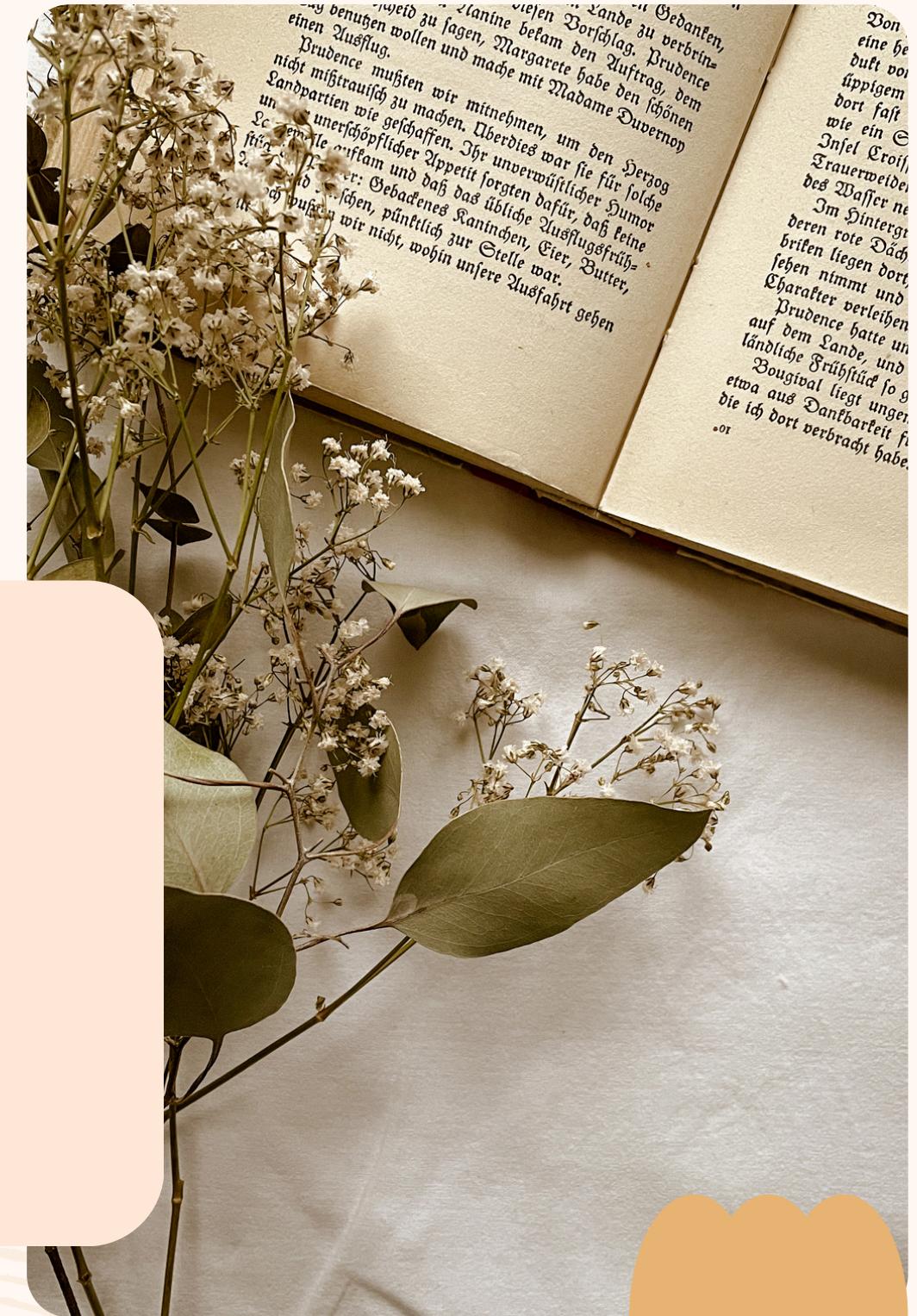
```
139 • select c.customer_id, c.name,  
140     sum(o.total_amount) as most_spend  
141     from customers c join orders o  
142     on c.customer_id = o.customer_id  
143     group by c.customer_id, c.name  
144     order by most_spend desc limit 3;  
145
```

Output :

	customer_id	name	most_spend
▶	457	Kim Turner	1398.90
	174	Jonathon Strickland	1080.95
	364	Carrie Perez	1052.27

# Advantages / Features

- Enables fast and efficient retrieval of book information
- Ensures secure management of customer data
- Provides easy tracking and monitoring of orders
- Designed to be scalable for handling large bookstores





- Integration with a web or mobile application for real-time access
- Implementation of a recommendation system for personalized book suggestions
- Addition of a secure payment gateway for online transactions
- Automation of sales, inventory, and customer activity reports
- Potential use of cloud-based databases for higher availability and performance

## Future Scope





# Conclusion

- The Online Bookstore Database project successfully demonstrates how SQL can be used to design and manage a structured system for handling books, customers, orders, and payments.
- It provides a reliable and efficient solution for bookstore management while laying the foundation for future enhancements such as integration with web applications, recommendation systems, and advanced reporting.





Thank you for exploring our digital stacks. We hope this Online Bookstore project, built on a robust SQL architecture for efficient data management, inspires your next great read. Your time and interest are greatly appreciated.

- Om Vedant

