# **SQL-1 Assignment**

## Create Database e\_commerce

create database e\_commerce;

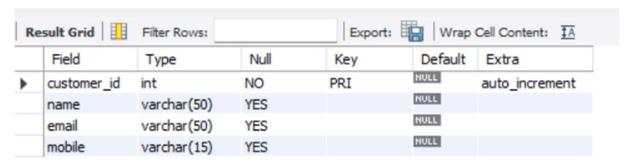


# 2. Create following Tables:

#### **Customers:**

- a. customer\_id int auto-increment primary key
- b. name varchar(50)
- c. email varchar(50)
- d. mobile varchar(15)

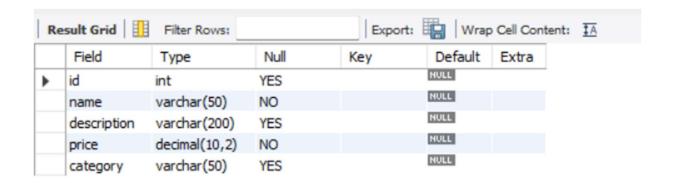
create table Customers(customer\_id int primary key auto\_increment, name varchar(50), email varchar(50), mobile varchar(15));



#### **Products:**

- a. id int
- b. name varchar(50) not null
- c. description varchar(200)
- d. price decimal(10, 2) not null
- e. category varchar(50)

create table Products(id int, name varchar(50) not null, description varchar(200), price decimal(10,2) not null, category varchar(50));



# 3. Modify Tables(using Alter keyword):

#### a. Add not null on name and email in the Customers table

Alter table customers

Modify column name varchar(50) not null, modify column email varchar(50) not null;

## b. Add unique key on email in the Customers table

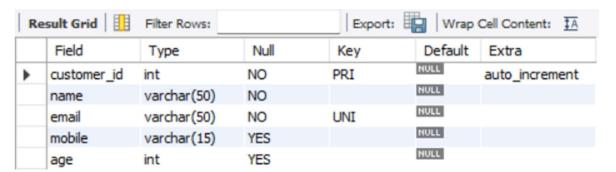
Alter table customers

Add constraint unique key(email);

## c. Add column age in the Customers table

**ALTER TABLE Customers** 

ADD Column age int;



# d. Change column name from id to product\_id in the Products table;

**ALTER TABLE Products** 

rename COLUMN id to product\_id;

## e. Add primary key and auto increment on product\_id in the Products table

**ALTER TABLE Products** 

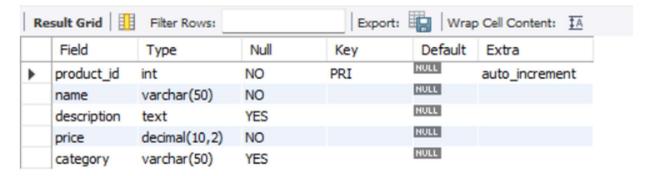
MODIFY COLUMN product\_id int auto\_increment,

ADD Constraint PRIMARY KEY(product\_id);

### f. Change datatype of description from varchar to text in the Products table

**ALTER TABLE Products** 

MODIFY COLUMN description text;



#### 4. Create table Order:

- a. order\_id int auto-increment primary key
- b. customer\_id int -foreign key
- c. product\_id int
- d. quantity int not null,
- e. order\_date date not null,
- f. status enum(Pending, Success, Cancel),
- g. payment\_method enum(Credit, Debit, UPI),
- h. total\_amount decimal(10, 2) not null

```
CREATE TABLE `Order`(

order_id int auto_increment primary key,

customer_id int,

product_id int,

quantity int not null,

order_date date not null,

status enum('Pending', 'Success', 'Cancel'),
```

```
payment_method enum('Credit', 'Debit', 'UPI'),
```

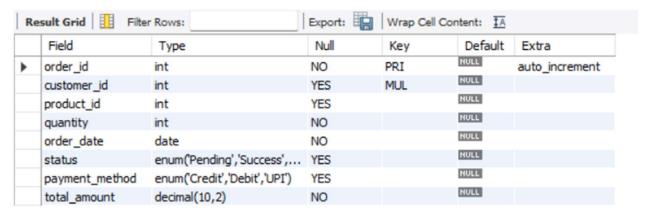
total\_amount decimal(10,2) not null,

foreign key(customer\_id) References Customers(customer\_id)

ON delete CASCADE

ON update Cascade

);



## 5. Modify Orders Table (using Alter keyword):

#### a. Change table name Order -> Orders

ALTER TABLE 'ORDER'

RENAME TO Orders;

## b. Set default value pending in status.

**ALTER TABLE ORDERS** 

MODIFY status enum('Pending', 'Success', 'Cancel') default 'Pending';

# c. Modify payment\_method ENUM to add one more value: 'COD'

**ALTER TABLE ORDERS** 

MODIFY payment\_method enum('CREDIT', 'Debit', 'UPI', 'COD');

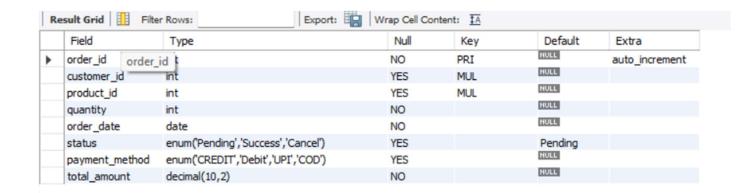
## d. Make product id as foreign key

**ALTER TABLE ORDERS** 

ADD CONSTRAINT FOREIGN KEY(product\_id) REFERENCES Products(product\_id)

ON DELETE CASCADE

on UPDATE CASCADE;

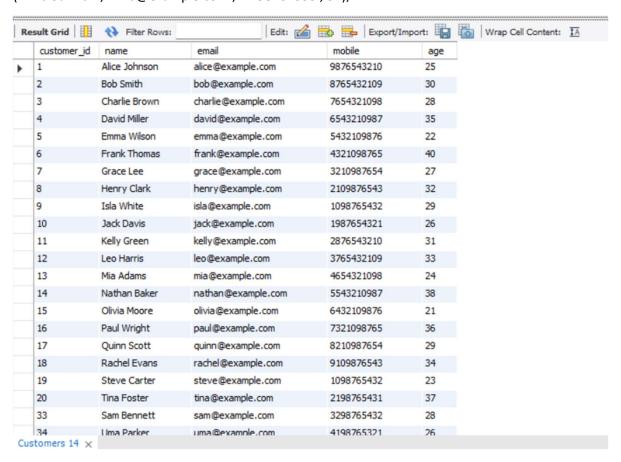


# 6. Insert 20 sample records in all the tables.

#### **Customers:**

INSERT INTO Customers (name, email, mobile, age) **VALUES** ('Alice Johnson', 'alice@example.com', '9876543210', 25), ('Bob Smith', 'bob@example.com', '8765432109', 30), ('Charlie Brown', 'charlie@example.com', '7654321098', 28), ('David Miller', 'david@example.com', '6543210987', 35), ('Emma Wilson', 'emma@example.com', '5432109876', 22), ('Frank Thomas', 'frank@example.com', '4321098765', 40), ('Grace Lee', 'grace@example.com', '3210987654', 27), ('Henry Clark', 'henry@example.com', '2109876543', 32), ('Isla White', 'isla@example.com', '1098765432', 29), ('Jack Davis', 'jack@example.com', '1987654321', 26), ('Kelly Green', 'kelly@example.com', '2876543210', 31), ('Leo Harris', 'leo@example.com', '3765432109', 33), ('Mia Adams', 'mia@example.com', '4654321098', 24), ('Nathan Baker', 'nathan@example.com', '5543210987', 38), ('Olivia Moore', 'olivia@example.com', '6432109876', 21), ('Paul Wright', 'paul@example.com', '7321098765', 36), ('Quinn Scott', 'quinn@example.com', '8210987654', 29), ('Rachel Evans', 'rachel@example.com', '9109876543', 34),

('Steve Carter', 'steve@example.com', '1098765432', 23),
('Tina Foster', 'tina@example.com', '2198765431', 37);
INSERT INTO Customers (name, email, mobile, age) VALUES
('Sam Bennett', 'sam@example.com', '3298765432', 28),
('Uma Parker', 'uma@example.com', '4198765321', 26),
('Victor Hughes', 'victor@example.com', '5098765210', 35),
('Wendy Morris', 'wendy@example.com', '6898765109', 30),
('Xavier Reed', 'xavier@example.com', '7798765008', 40),
('Yasmine Fisher', 'yasmine@example.com', '8698764997', 24),
('Zane Cooper', 'zane@example.com', '9598764886', 27),
('Eleanor Gray', 'eleanor@example.com', '0498764775', 22),
('Oscar Phillips', 'oscar@example.com', '1398764664', 32),
('Nina Sullivan', 'nina@example.com', '2298764553', 31);



## **Products:**

```
INSERT INTO Products (name, description, price, category) VALUES
('Laptop', 'High-performance laptop with 16GB RAM and 512GB SSD.', 1200.00, 'Electronics'),
('Smartphone', 'Latest model smartphone with 5G support.', 800.00, 'Electronics'),
('Headphones', 'Noise-canceling over-ear headphones.', 150.00, 'Accessories'),
('Smartwatch', 'Fitness smartwatch with heart rate monitoring.', 250.00, 'Wearables'),
('Bluetooth Speaker', 'Portable Bluetooth speaker with deep bass.', 100.00, 'Accessories'),
('Gaming Console', 'Next-gen gaming console with 4K support.', 500.00, 'Gaming'),
('Wireless Mouse', 'Ergonomic wireless mouse with fast response.', 40.00, 'Accessories'),
('Mechanical Keyboard', 'RGB mechanical keyboard with blue switches.', 80.00, 'Accessories'),
('External Hard Drive', '2TB external hard drive for backup storage.', 90.00, 'Storage'),
('Smart TV', '55-inch 4K Ultra HD Smart TV.', 700.00, 'Electronics'),
('Tablet', '10-inch tablet with stylus support.', 450.00, 'Electronics'),
('Fitness Band', 'Waterproof fitness band with step tracking.', 70.00, 'Wearables'),
('Drone', 'Quadcopter drone with 4K camera.', 600.00, 'Electronics'),
('Camera', 'DSLR camera with 24MP sensor.', 1000.00, 'Photography'),
('Tripod', 'Adjustable tripod stand for cameras.', 50.00, 'Photography'),
('Microwave Oven', '800W microwave with grill function.', 200.00, 'Home Appliances'),
('Refrigerator', 'Double-door refrigerator with inverter technology.', 900.00, 'Home Appliances'),
('Air Conditioner', 'Split AC with fast cooling technology.', 1100.00, 'Home Appliances'),
('Washing Machine', 'Front-load washing machine with 6kg capacity.', 850.00, 'Home Appliances'),
('Electric Kettle', '1.5L electric kettle with auto shut-off.', 30.00, 'Home Appliances');
INSERT INTO Products (name, description, price, category) VALUES
('Smart Door Lock', 'Fingerprint and password-enabled smart lock.', 180.00, 'Home Security'),
('Gaming Chair', 'Ergonomic gaming chair with lumbar support.', 250.00, 'Furniture'),
('Power Bank', '20000mAh power bank with fast charging support.', 60.00, 'Accessories'),
('Monitor', '27-inch 144Hz gaming monitor.', 350.00, 'Electronics'),
('Graphic Tablet', 'Drawing tablet with stylus and pressure sensitivity.', 280.00, 'Electronics'),
('Projector', 'Full HD home theater projector.', 500.00, 'Electronics'),
```

('Coffee Maker', 'Automatic coffee maker with milk frother.', 150.00, 'Home Appliances'),

('Treadmill', 'Motorized treadmill with incline settings.', 750.00, 'Fitness'),

('VR Headset', 'Virtual reality headset with motion controllers.', 400.00, 'Gaming'),

('Electric Scooter', 'Foldable electric scooter with 25km range.', 700.00, 'Transport');

INSERT INTO Products (name, description, price, category)

#### **VALUES**

('Smart Lamp', 'WiFi-enabled smart lamp with voice control.', 60.00, 'Home Appliances'),

('VR Headset', 'Virtual reality headset for immersive gaming.', 300.00, 'Gaming'),

('Graphic Tablet', 'Digital drawing tablet with stylus.', 150.00, 'Accessories'),

('Security Camera', 'Wireless security camera with night vision.', 120.00, 'Electronics'),

('Coffee Maker', 'Automatic coffee machine with grinder.', 250.00, 'Home Appliances');

product_id	name	description	price	category	
1	Laptop	High-performance laptop with 16GB RAM and 512GB SSD.	1200.00	Electronics	
2	Smartphone	Latest model smartphone with 5G support.	800.00	Electronics	
3	Headphones	Noise-canceling over-ear headphones.	150.00	Accessories	
4	Smartwatch	Fitness smartwatch with heart rate monitoring.	250.00	Wearables	
5	Bluetooth Speaker	Portable Bluetooth speaker with deep bass.	100.00	Accessories	
6	Gaming Console	Next-gen gaming console with 4K support.	500.00	Gaming	
7	Wireless Mouse	Ergonomic wireless mouse with fast response.	40.00	Accessories	
8	Mechanical Keyboard	RGB mechanical keyboard with blue switches.	80.00	Accessories	
9	External Hard Drive	2TB external hard drive for backup storage.	90.00	Storage	
10	Smart TV	55-inch 4K Ultra HD Smart TV.	700.00	Electronics	
11	Tablet	10-inch tablet with stylus support.	450.00	Electronics	
12	Fitness Band	Waterproof fitness band with step tracking.	70.00	Wearables	
13	Drone	Quadcopter drone with 4K camera.	600.00	Electronics	
14	Camera	DSLR camera with 24MP sensor.	1000.00	Photography	
15	Tripod	Adjustable tripod stand for cameras.	50.00	Photography	
16	Microwave Oven	800W microwave with grill function.	200.00	Home Appliances	
17	Refrigerator	Double-door refrigerator with inverter technology.	900.00	Home Appliances	
18	Air Conditioner	Split AC with fast cooling technology.	1100.00	Home Appliances	
19	Washing Machine	Front-load washing machine with 6kg capacity.	850.00	Home Appliances	
20	Electric Kettle	1.5L electric kettle with auto shut-off.	30.00	Home Appliances	
41	Smart Door Lock	Fingerprint and password-enabled smart lock.	180.00	Home Security	
42	Gaming Chair	Ergonomic gaming chair with lumbar support.	250.00	Furniture	
43	Power Bank	20000mAh power bank with fast charging support.	60.00	Accessories	
44	Monitor	27-inch 144Hz gaming monitor.	350.00	Electronics	
45	Graphic Tablet	Drawing tablet with stylus and pressure sensitivity.	280.00	Electronics	

#### Orders:

```
INSERT INTO Orders (customer_id, product_id, quantity, order_date, status, payment_method, total_amount) VALUES
```

- (1, 1, 2, '2024-02-01', 'Pending', 'Credit', 2400.00),
- (2, 3, 1, '2024-02-02', 'Success', 'Debit', 150.00),
- (3, 5, 3, '2024-02-03', 'Cancel', 'UPI', 300.00),
- (4, 7, 2, '2024-02-04', 'Success', 'CREDIT', 80.00),
- (5, 9, 5, '2024-02-05', 'Pending', 'Debit', 450.00),
- (6, 11, 1, '2024-02-06', 'Success', 'UPI', 450.00),
- (7, 13, 4, '2024-02-07', 'Cancel', 'CREDIT', 2400.00),
- (8, 15, 2, '2024-02-08', 'Pending', 'Debit', 100.00),
- (9, 17, 3, '2024-02-09', 'Success', 'UPI', 3300.00),
- (10, 19, 1, '2024-02-10', 'Cancel', 'CREDIT', 850.00),
- (11, 2, 2, '2024-02-11', 'Pending', 'Debit', 1600.00),
- (12, 4, 5, '2024-02-12', 'Success', 'UPI', 250.00),
- (13, 6, 1, '2024-02-13', 'Cancel', 'CREDIT', 500.00),
- (14, 8, 3, '2024-02-14', 'Pending', 'Debit', 270.00),
- (15, 10, 4, '2024-02-15', 'Success', 'UPI', 2800.00),
- (16, 12, 2, '2024-02-16', 'Cancel', 'CREDIT', 140.00),
- (17, 14, 3, '2024-02-17', 'Pending', 'Debit', 3000.00),
- (18, 16, 5, '2024-02-18', 'Success', 'UPI', 1000.00),
- (19, 18, 1, '2024-02-19', 'Cancel', 'CREDIT', 1100.00),
- (20, 20, 2, '2024-02-20', 'Pending', 'Debit', 60.00);

INSERT INTO Orders (customer\_id, product\_id, quantity, order\_date, status, payment\_method, total\_amount) VALUES

- (1, 41, 1, '2024-02-21', 'Success', 'UPI', 180.00),
- (2, 42, 2, '2024-02-22', 'Pending', 'CREDIT', 500.00),
- (3, 43, 3, '2024-02-23', 'Cancel', 'Debit', 180.00),
- (4, 44, 1, '2024-02-24', 'Success', 'UPI', 350.00),

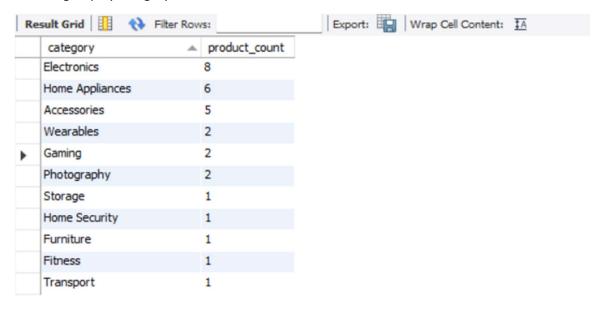
- (5, 45, 2, '2024-02-25', 'Pending', 'CREDIT', 560.00),
- (6, 46, 1, '2024-02-26', 'Success', 'Debit', 500.00),
- (7, 47, 3, '2024-02-27', 'Cancel', 'UPI', 450.00),
- (8, 48, 2, '2024-02-28', 'Pending', 'CREDIT', 300.00),
- (9, 49, 1, '2024-02-29', 'Success', 'Debit', 700.00),
- (10, 50, 4, '2024-03-01', 'Pending', 'UPI', 2800.00);

Re	esult Grid	- Name of the Filter	Rows:		Edit:	■ ■	Export/Import:	Wrap Cell Co
	order_id	customer_id	product_id	quantity	order_date	status	payment_method	total_amount
	1	1	1	2	2024-02-01	Pending	CREDIT	2400.00
	2	2	3	1	2024-02-02	Success	Debit	150.00
	3	3	5	3	2024-02-03	Cancel	UPI	300.00
	4	4	7	2	2024-02-04	Success	CREDIT	80.00
	5	5	9	5	2024-02-05	Pending	Debit	450.00
	6	6	11	1	2024-02-06	Success	UPI	450.00
	7	7	13	4	2024-02-07	Cancel	CREDIT	2400.00
	8	8	15	2	2024-02-08	Pending	Debit	100.00
	9	9	17	3	2024-02-09	Success	UPI	3300.00
	10	10	19	1	2024-02-10	Cancel	CREDIT	850.00
	11	11	2	2	2024-02-11	Pending	Debit	1600.00
	12	12	4	5	2024-02-12	Success	UPI	250.00
	13	13	6	1	2024-02-13	Cancel	CREDIT	500.00
	14	14	8	3	2024-02-14	Pending	Debit	270.00
	15	15	10	4	2024-02-15	Success	UPI	2800.00
	16	16	12	2	2024-02-16	Cancel	CREDIT	140.00
	17	17	14	3	2024-02-17	Pending	Debit	3000.00
	18	18	16	5	2024-02-18	Success	UPI	1000.00
	19	19	18	1	2024-02-19	Cancel	CREDIT	1100.00
	20	20	20	2	2024-02-20	Pending	Debit	60.00
	31	1	41	1	2024-02-21	Success	UPI	180.00
	32	2	42	2	2024-02-22	Pending	CREDIT	500.00
	33	3	43	3	2024-02-23	Cancel	Debit	180.00
	34	4	44	1	2024-02-24	Success	UPI	350.00
	35	5	45	2	2024-02-25	Pending	CREDIT	560.00
						-	- 1	

# 7. Perform following queries:

a. Count the number of products as product\_count in each category.

SELECT category, count(name) AS product\_count FROM Products group by category;



b. Retrieve all products that belong to the 'Electronics' category, have a price between \$50 and \$500, and whose name contains the letter 'a'.

SELECT \* from Products

where category='Electronics' AND

price between 5 and 500 AND

name like '%a%';



#### c. Get the top 5 most expensive products in the 'Electronics' category, skipping the first 2.

select name, price

from products

where category='Electronics'

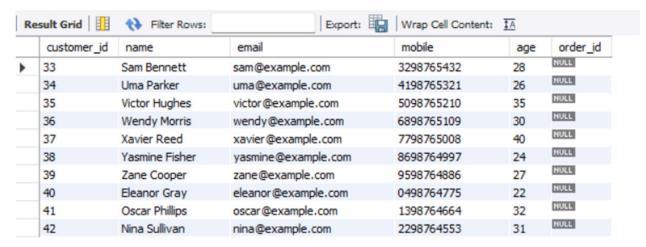
order by price DESC

limit 5 offset 2;



# d. Retrieve customers who have not placed any orders.

SELECT c.\*, o.order\_id from customers c LEFT JOIN orders o ON c.customer\_id=o.customer\_id where o.order\_id is NULL;



#### e. Find the average total amount spent by each customer.

SELECT o.customer\_id, c.name AS customer\_name, AVG(o.total\_amount) AS average\_spent FROM Orders o

JOIN Customers c ON o.customer\_id = c.customer\_id

GROUP BY o.customer\_id, c.name

ORDER BY average\_spent DESC;



f. Get the products that have a price less than the average price of all products.

SELECT name from products where price < (SELECT avg(price) from products);

	name					
•	Headphones					
	Smartwatch					
	Bluetooth Speaker					
	Wireless Mouse					
	Mechanical Keyboard					
	External Hard Drive					
	Fitness Band					
	Tripod					
	Microwave Oven					
	Electric Kettle					
	Smart Door Lock					
	Gaming Chair					
	Power Bank					
	Monitor					
	Graphic Tablet					
	Coffee Maker					
	VR Headset					

## g. Calculate the total quantity of products ordered by each customer:

 ${\tt SELECT\ o.customer\_id,\ c.name\ AS\ customer\_name,\ SUM(o.quantity)\ AS\ total\_products\_ordered}$ 

FROM Orders o

JOIN Customers c ON o.customer\_id = c.customer\_id

GROUP BY o.customer\_id, c.name

ORDER BY total\_products\_ordered DESC;



## h. List all orders along with customer name and product name.

Select p.name, p.price

from orders o

RIGHT join Products p

ON o.product\_id=p.product\_id

# where o.order\_id is NULL;

order_id	customer_name	product_name	order_date	status	payment_method	total_amount
1	Alice Johnson	Laptop	2024-02-01	Pending	CREDIT	2400.00
2	Bob Smith	Headphones	2024-02-02	Success	Debit	150.00
3	Charlie Brown	Bluetooth Speaker	2024-02-03	Cancel	UPI	300.00
4	David Miller	Wireless Mouse	2024-02-04	Success	CREDIT	80.00
5	Emma Wilson	External Hard Drive	2024-02-05	Pending	Debit	450.00
6	Frank Thomas	Tablet	2024-02-06	Success	UPI	450.00
7	Grace Lee	Drone	2024-02-07	Cancel	CREDIT	2400.00
8	Henry Clark	Tripod	2024-02-08	Pending	Debit	100.00
9	Isla White	Refrigerator	2024-02-09	Success	UPI	3300.00
10	Jack Davis	Washing Machine	2024-02-10	Cancel	CREDIT	850.00
11	Kelly Green	Smartphone	2024-02-11	Pending	Debit	1600.00
12	Leo Harris	Smartwatch	2024-02-12	Success	UPI	250.00
13	Mia Adams	Gaming Console	2024-02-13	Cancel	CREDIT	500.00
14	Nathan Baker	Mechanical Keyboard	2024-02-14	Pending	Debit	270.00
15	Olivia Moore	Smart TV	2024-02-15	Success	UPI	2800.00
16	Paul Wright	Fitness Band	2024-02-16	Cancel	CREDIT	140.00
17	Quinn Scott	Camera	2024-02-17	Pending	Debit	3000.00
18	Rachel Evans	Microwave Oven	2024-02-18	Success	UPI	1000.00
19	Steve Carter	Air Conditioner	2024-02-19	Cancel	CREDIT	1100.00
20	Tina Foster	Electric Kettle	2024-02-20	Pending	Debit	60.00
31	Alice Johnson	Smart Door Lock	2024-02-21	Success	UPI	180.00
32	Bob Smith	Gaming Chair	2024-02-22	Pending	CREDIT	500.00
33	Charlie Brown	Power Bank	2024-02-23	Cancel	Debit	180.00
34	David Miller	Monitor	2024-02-24	Success	UPI	350.00
35	Emma Wilson	Graphic Tablet	2024-02-25	Pending	CREDIT	560.00
20	r Th	Danie stan	2024 02 20	C	n-Lis	F00 00

# i. Find products that have never been ordered.

Select p.name, p.price

from orders o

RIGHT join Products p

ON o.product\_id=p.product\_id

where o.order\_id is NULL;

