08 October 2024

Natalia Palej

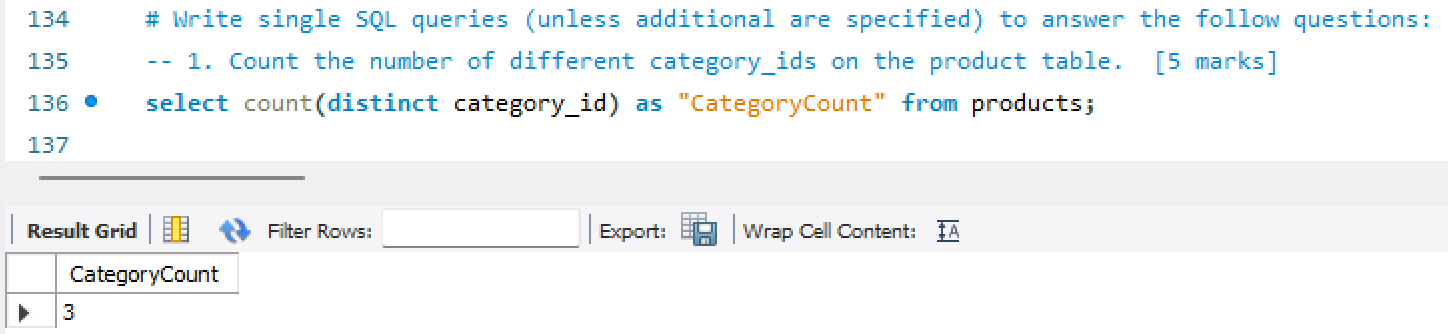
Student ID: A00279259

Software Design with AI for Cloud Computing

SQL Assessment

Q1

select count(distinct category\_id) as “CategoryCount” from products;



Q2

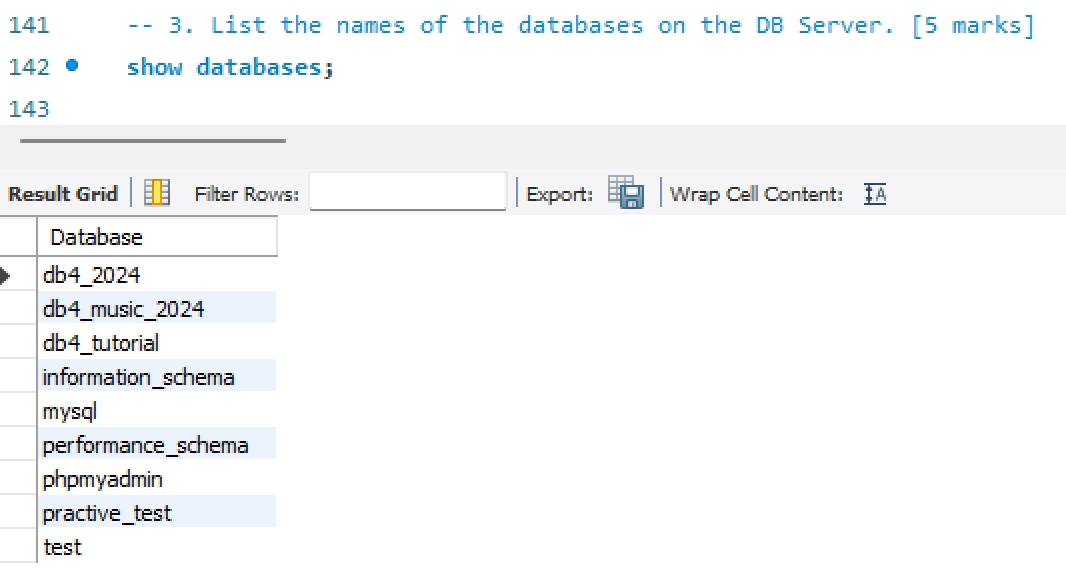
select first\_name, last\_name from customers where length(last\_name) >= 6;

A screenshot of a computer

Description automatically generated

Q3

show databases;



Q4

select sum(list\_price) as "Total List Price", avg(list\_price) as "Average List Price", avg(discount\_percent) as "Average Discount Percent", max(list\_price) as "H", min(list\_price) as "L" from products;

A screenshot of a computer

Description automatically generated

Q5

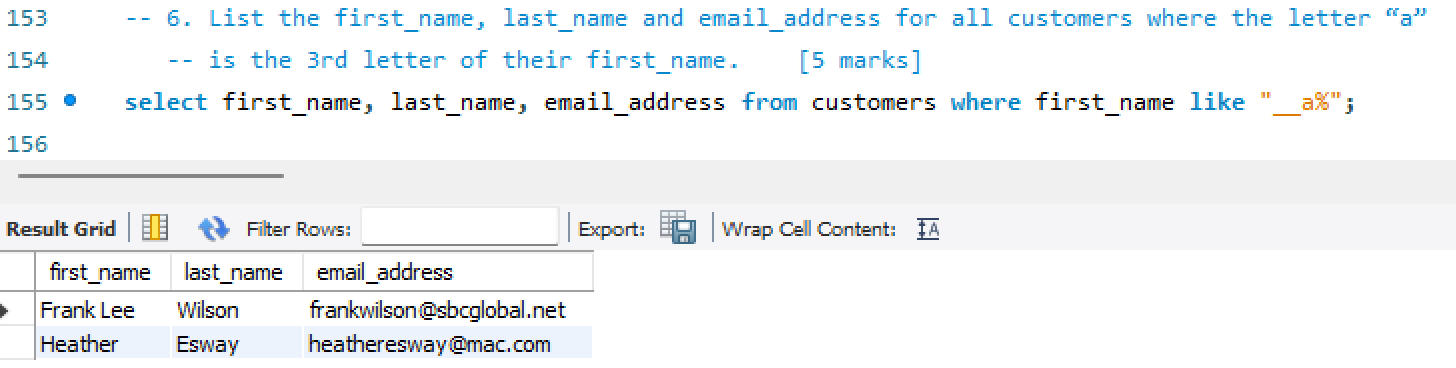
select product\_code, product\_name from products where description like "%Cymbals%";

A screenshot of a computer

Description automatically generated

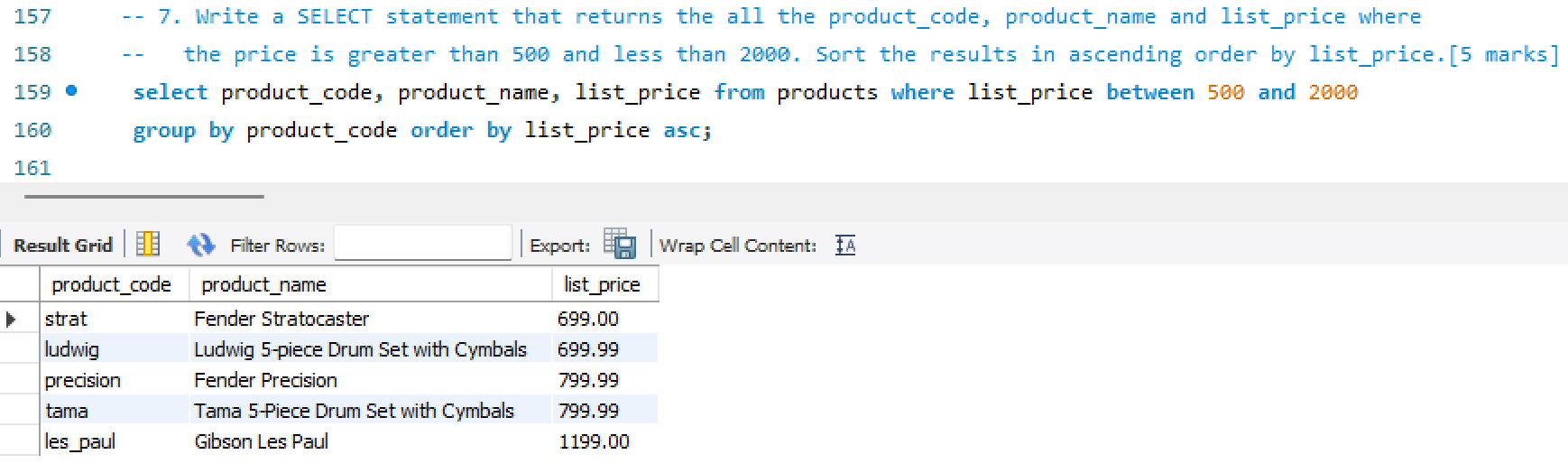
Q6

select first\_name, last\_name, email\_address from customers where first\_name like "\_\_a%";



Q7

select product\_code, product\_name, list\_price from products where list\_price between 500 and 2000 group by product\_code order by list\_price asc;



Q8

select c.category\_name, p.product\_name, p.list\_price from products p join categories c order by c.category\_name desc, p.product\_name asc;

A screenshot of a computer

Description automatically generated

Q9

drop view if exists vwPriceAndDiscount;

create view vwPriceAndDiscount as select product\_name as “Product”, list\_price as "Price", discount\_percent as "Discount", round((list\_price\*discount\_percent)/100, 2) as "Discount Amount", list\_price-round((list\_price\*discount\_percent)/100, 2) as "Discounted Price" from products;

select \* from vwPriceAndDiscount order by Product desc;

A screenshot of a computer

Description automatically generated

Q10

select product\_name as "Product", sum(list\_price) as "TotalPrice", sum(round((list\_price\*discount\_percent)/100, 2)) as "TotalDiscount" from products group by product\_name order by product\_name desc;

A screenshot of a computer

Description automatically generated

Q11

select c.email\_address, count(o.order\_id) as "TotalOrders" from customers c left join orders o on c.customer\_id=o.customer\_id group by c.customer\_id;

A screenshot of a computer

Description automatically generated

Q12

insert into categories (category\_name) values ("Kazoos");

**NOTE: I have added originally as “Kazooos” and then noticed its supposed to be “Kazoos”, that’s why theres two entries 😊**

A screenshot of a computer

Description automatically generated

describe categories;

A screenshot of a computer

Description automatically generated

Q13

select \* from categories;  
update categories set category\_name="Misc" where category\_name like "Kazoos";

A screenshot of a computer

Description automatically generated

Q14

drop procedure if exists sp\_UpdateDiscountPercent;

delimiter //

create procedure sp\_UpdateDiscountPercent(

productID int

)

begin

start transaction;

update products set discount\_percent="99" where product\_id=ProductID;

commit;

end //   
delimiter ;

select \* from products where product\_id=1;

call sp\_UpdateDiscountPercent(1);

select \* from products where product\_id=1;

**Before calling** sp\_UpdateDiscountPercent

A screenshot of a computer

Description automatically generated

**After calling** sp\_UpdateDiscountPercent

A screenshot of a computer

Description automatically generated

Q15

drop function if exists fnGetDiscountedPrice;

delimiter //

create function fnGetDiscountedPrice(

productID int

)

returns decimal (9, 2)

begin

declare DiscountedPrice decimal(9,2);

select (list\_price\*discount\_percent)/100 into DiscountedPrice from products where product\_id=ProductID;

return DiscountedPrice;

end //

delimiter ;

select fnGetDiscountedPrice(1);

A screenshot of a computer

Description automatically generated