



PC STORE

РОБЕРТ БОРИСОВ 62529

КАЛОЯН БОЖИЛСКИ 62527

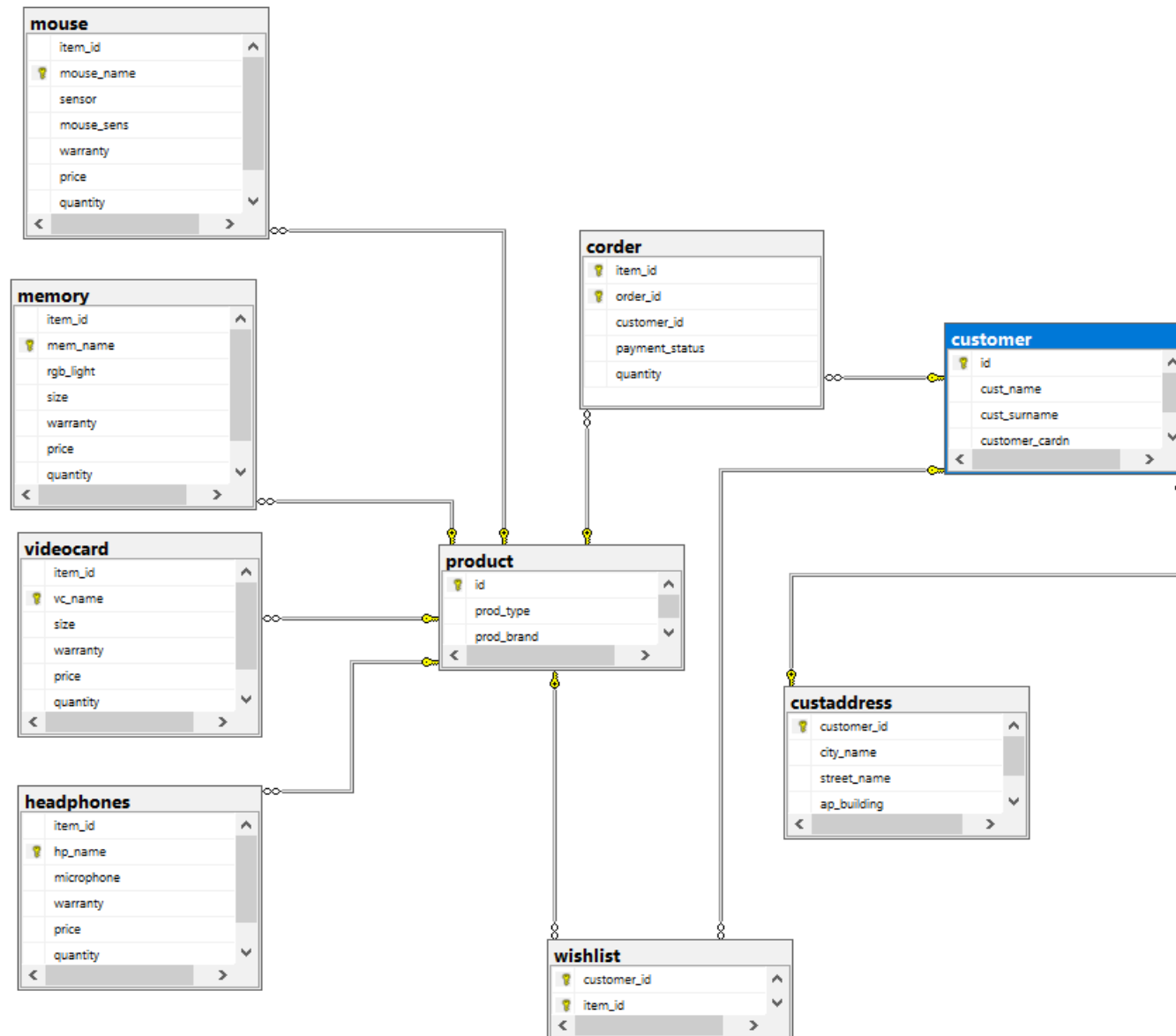
ВАСИЛ ЛЯЛЯКОВ 62371

Описание на темата

- Описание на заданието
- Начин на реализация



Cxema



Пример с прости заявка

Напишете заявка, която извежда всички слушалки с микрофон, както и гаранцията им.

```
select hp_name, warranty  
from headphones  
where microphone = 'Y'
```

	hp_name	warranty
1	HYPERX STINGER	12
2	LOGITECH G332	22
3	N60NC	10
4	WF-1000XM4	3

Пример с заявка върху две или повече релации

Заявка, която извежда имената на клиентите, чийто адрес се състои от точно 2 думи.

```
select c.cust_name as 'name' , c.cust_surname as surname
from customer c,
custaddress ca
where ca.customer_id = c.id
and ca.street_name like '% %'
and ca.street_name not like '% % %' |
```

	name	sumame
1	DIMITUR	MARTINOV
2	NENCHO	TOSHEV
3	PETUR	DIMITROV

Пример с подзаявка

Напишете заявка, която извежда името, марката и цената на продукта (слушалки, мишки, видеокарти и memory) с най-висока цена, като използвате псевдонимите съответно name за име и brand за марка

```
select hw.name, p.prod_brand as brand, hw.price
from product p, (select hp_name as 'name', item_id, price from headphones
union
select mouse_name, item_id, price from mouse
union
select mem_name, item_id, price from memory
union
select vc_name, item_id, price from videocard) hw
where hw.item_id = p.id
and hw.price >= ALL(select price from headphones
union
select price from mouse
union
select price from videocard
union
select price from memory)
```

	name	brand	price
1	NVIDIA 3080	NVIDIA	2000

Пример със съединение

Напишете заявка, която извежда името и цената на всеки продукт, който никога не е бил поръчван

```
select hw.name, hw.price
from product p1 JOIN (select mouse_name as 'name', item_id, price from mouse
                      union
                      select hp_name as 'name', item_id, price from headphones
                      union
                      select vc_name as 'name', item_id, price from videocard
                      union
                      select mem_name as 'name', item_id, price from memory) hw
ON p1.id = hw.item_id
LEFT JOIN corder c1 ON p1.id = c1.item_id
where c1.item_id is null
```

	name	price
1	AMD RX 6900 XT	799
2	CORSAIR MASTER	200
3	CORSAIR VENGEANCE	300
4	G PRO X	100
5	G502 HERO	200
6	Geforce RTX 3080	799
7	Geforce RTX 3090	499
8	HYPERX FURY	200
9	Trident X RGB	1000

Пример а агрегатна функция и групиране

Напишете заявка, която извежда броя на паметите с размер над 4GB, за тези марки, които имат поне 2 памети в базата данни

```
SELECT p.prod_brand, AVG(m.size) as "avg_size"
FROM product p
JOIN memory m
ON p.id = m.item_id
WHERE p.prod_brand in (SELECT p2.prod_brand
                        FROM product p2
                        JOIN videocard v
                        ON v.item_id = p2.id)
GROUP BY p.prod_brand
```

	prod_brand	num_mems
1	ADATA	2
2	CORSAIR	2
3	G.Skill	2

Примери с ограничения

```
ALTER TABLE videocard ADD CONSTRAINT PK_videocards PRIMARY KEY(vc_name)
```

```
ALTER TABLE wishlist ADD CONSTRAINT FK_wishlist_customer FOREIGN KEY(customer_id) REFERENCES customer(id)
```

```
ALTER TABLE wishlist ADD CONSTRAINT FK_wishlist_product FOREIGN KEY(item_id) REFERENCES product(id)
```

```
ALTER TABLE headphones ADD CONSTRAINT CK_headphones  
CHECK(warranty > 0 and price > 0 and microphone in ('Y','N'));
```

Примери с индекси и изгледи

```
Create View CustomersBigWishList
as
SELECT w.customer_id, COUNT(w.item_id) as "num_products"
FROM WATCHLIST w
GROUP BY (w.customer_id)
HAVING COUNT(w.item_id) > 10
```

Пример за изглед на тези клиенти от таблицата wishlist с повече от 10 любими артикула

```
CREATE INDEX idx_headphones_price
ON headphones(price)
CREATE INDEX idx_customer_id
ON corder(customer_id)
```

Създаване на индекси за цена на слушалките и клиентско id при поръчките, за по-бързо търсене

Пример с тригър

```
CREATE TRIGGER delete_ifmore_all on corder
AFTER INSERT
AS
begin
SET NOCOUNT ON

DECLARE @Quan int
SELECT @Quan = INSERTED.quantity from INSERTED
DECLARE @ID int
SELECT @ID = inserted.item_id from inserted
DECLARE @Type varchar(15)
SELECT @Type = (select prod_type from product where id = @ID)
if exists(select m1.item_id from (select item_id,quantity from headphones
                                union
                                select item_id,quantity from mouse
                                union
                                select item_id,quantity from videocard
                                union
                                select item_id,quantity from memory) m1 where item_id = @ID and m1.quantity < @Quan)

begin
rollback transaction
end
else
if exists(select 1 from headphones where item_id = @ID)
begin
UPDATE headphones
SET quantity = quantity - @Quan from headphones where item_id = @ID
end
else if exists (select 1 from mouse where item_id = @ID)
begin
UPDATE mouse
SET quantity = quantity - @Quan from mouse where item_id = @ID
end
else if exists(select 1 from videocard where item_id = @ID)
begin
UPDATE videocard
SET quantity = quantity - @Quan from videocard where item_id = @ID
end
else if exists(select 1 from memory where item_id = @ID)
begin
UPDATE memory
SET quantity = quantity - @Quan from memory where item_id = @ID
end
end
```

Заключение

- Оценка за реализацията
- Препоръки за възможни подобрения



Благодарим за вниманието!

