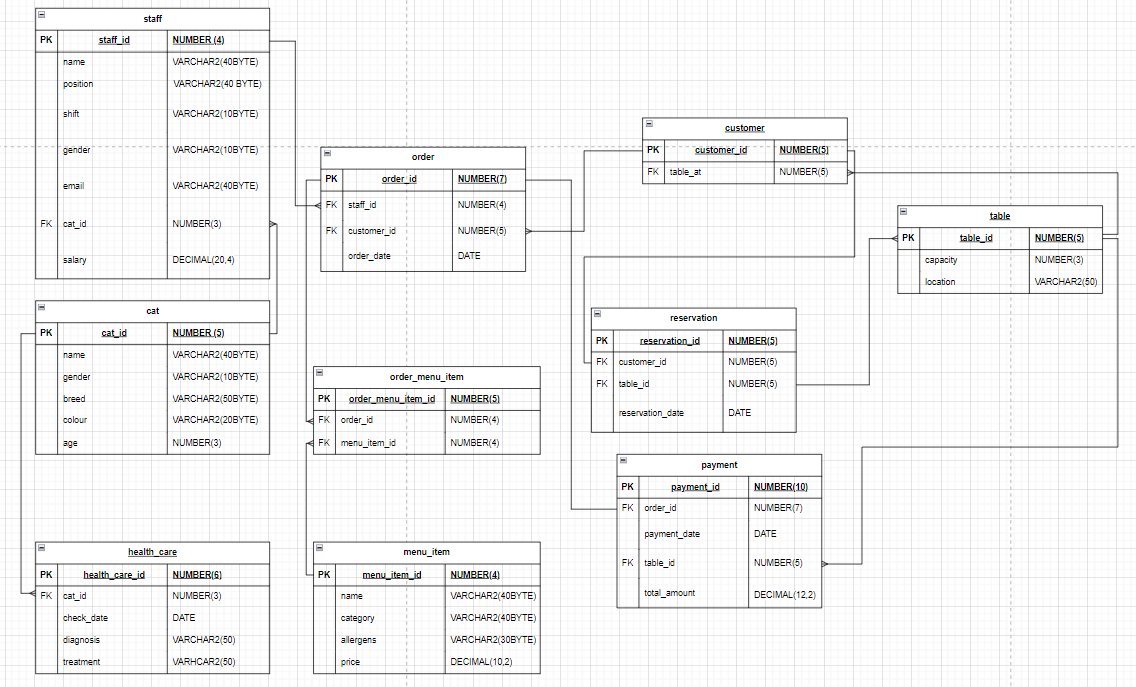
Popis úlohy:

Zameriaval som sa na vytvorenie rôznych pohľadov a triggerov. Nižšie je zhrnutie každej úlohy spolu s SQL kódom a demonštráciou funkčnosti.

Relačná schéma:



**SQL kódy a popis:**

1. **Pohľad OlderCats:**
   * Tento pohľad vyberá všetky záznamy z tabuľky Cat, kde vek mačiek je väčší ako 3 roky.

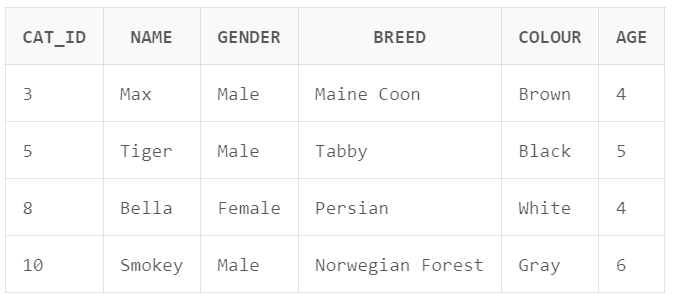
-- s netriviálnym selektom

CREATE VIEW OlderCats AS

SELECT \*

FROM Cat

WHERE AGE > 3;



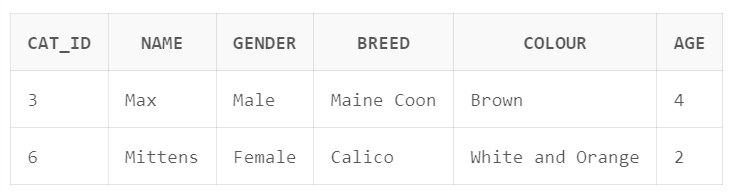
1. **Pohľad CatsWithMNames:**
   * Vyberá mačky, ktorých mená začínajú na písmeno 'M'.

-- s netriviálnym selektom

CREATE VIEW CatsWithMNames AS

SELECT \*

FROM Cat

WHERE SUBSTR(name, 1, 1) = 'M'; 

1. **Pohľad CatAreCarredBy:**
   * Zobrazuje informácie o zamestnancoch, ktorí sa starajú o mačky, vrátane diagnozy starostlivosti o zdravie.

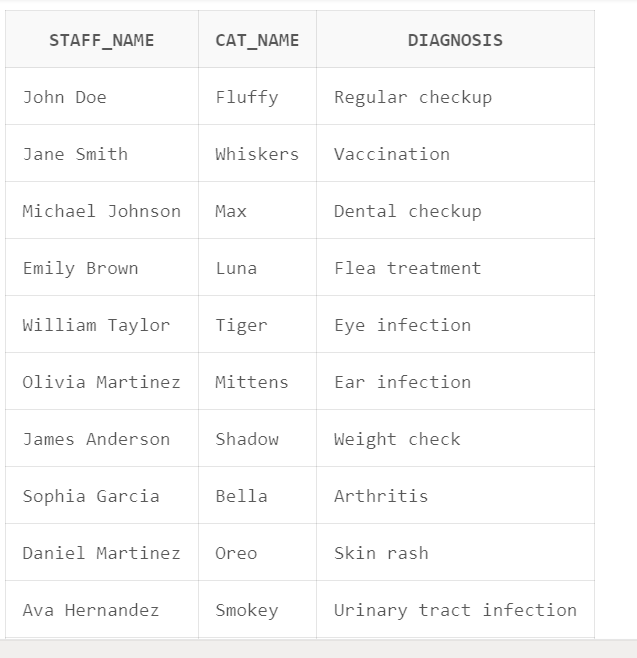
-- 1x spojenie 3 tabuliek

CREATE VIEW CatAreCarredBy AS

SELECT Staff.name AS staff\_name, Cat.name AS cat\_name, health\_care.diagnosis

FROM Staff

JOIN Cat ON Staff.cat\_id = Cat.cat\_id

JOIN health\_care ON Cat.cat\_id = health\_care.cat\_id; 

1. **Pohľad TableOrder:**
   * Zoskupuje údaje o objednávkach z tabuliek "order", menu\_item a zákazník.

-- 1x outer join

CREATE VIEW TableOrder AS

SELECT "order".order\_id, menu\_item.name AS menu\_item\_name, customer.table\_at

FROM "order"

FULL OUTER JOIN menu\_item ON "order".menu\_item\_id = menu\_item.menu\_item\_id

FULL OUTER JOIN customer ON "order".customer\_id = customer.customer\_id; 

1. **Pohľad PaidOrder:**
   * Zobrazuje informácie o platbách k objednávkam.

-- 1x spojenie aspoň 2 tabuliek

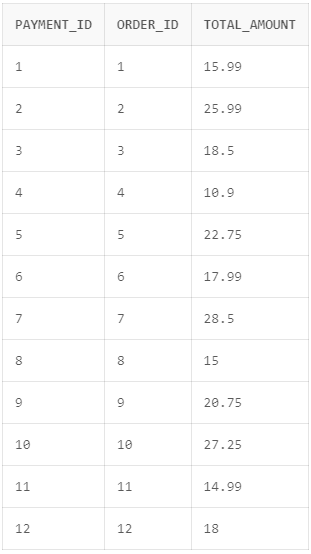
CREATE VIEW PaidOrder AS

SELECT p.payment\_id, o.order\_id, p.total\_amount

FROM payment p

RIGHT JOIN "order" o ON p.order\_id = o.order\_id;

SELECT \* FROM PaidOrder;



1. **Pohľad CustomerOrderCount:**
   * Zobrazuje počet objednávok pre každého zákazníka.

-- s použitím agregačných funkcií

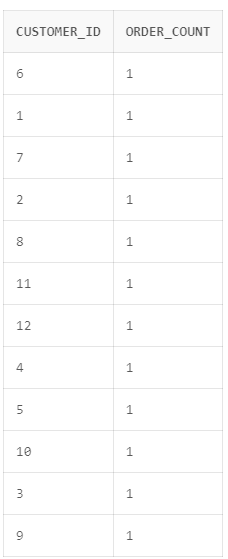
CREATE VIEW CustomerOrderCount AS

SELECT customer\_id, COUNT(\*) AS order\_count

FROM "order"

GROUP BY customer\_id;

SELECT \* FROM CustomerOrderCount;



1. **Pohľad CustomerTotalPayment:**
   * Zobrazuje celkovú sumu zaplatenú každým zákazníkom.

-- s použitím agregačných funkcií alebo zoskupenia

CREATE VIEW CustomerTotalPayment AS

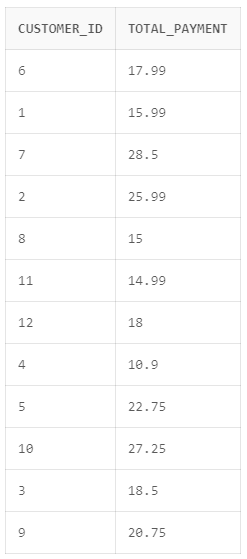
SELECT "order".customer\_id, SUM(payment.total\_amount) AS total\_payment

FROM "order"

JOIN payment ON "order".order\_id = payment.order\_id

GROUP BY "order".customer\_id;

SELECT \* FROM CustomerTotalPayment;



1. **Pohľad StaffAndCatIDs:**
   * Spája ID zamestnancov a mačiek do jednej tabuľky.

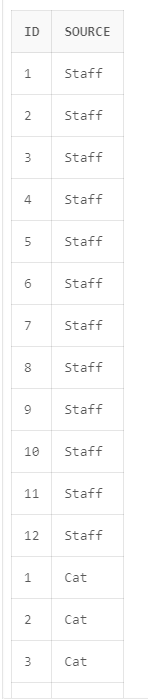
CREATE VIEW StaffAndCatIDs AS

SELECT Staff\_id AS ID, 'Staff' AS Source FROM Staff

UNION ALL

SELECT cat\_id AS ID, 'Cat' AS Source FROM cat;

SELECT \* FROM StaffAndCatIDs;



1. **Pohľad CustomersWithOrders:**
   * Zobrazuje zákazníkov, ktorí majú aspoň jednu objednávku.

-- s použitím netriviálnych vnorených selektov

CREATE VIEW CustomersWithOrders AS

SELECT customer.customer\_id, customer.table\_at

FROM customer

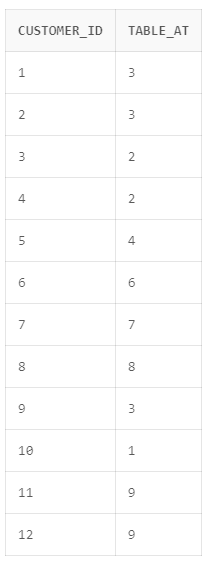
WHERE EXISTS (

SELECT 1

FROM "order"

WHERE "order".customer\_id = customer.customer\_id

); SELECT \* FROM CustomersWithOrders;



1. **Pohľad CustomerOrderPaymentInfo:**
   * Poskytuje informácie o počte objednávok a priemernú sumu platby pre každého zákazníka.

-- s použitím netriviálnych vnorených selektov

CREATE VIEW CustomerOrderPaymentInfo AS

SELECT customer.customer\_id,

(

SELECT COUNT(\*)

FROM "order"

WHERE "order".customer\_id = customer.customer\_id

) AS order\_count,

(

SELECT AVG(payment.total\_amount)

FROM payment

WHERE payment.order\_id IN (

SELECT "order".order\_id

FROM "order"

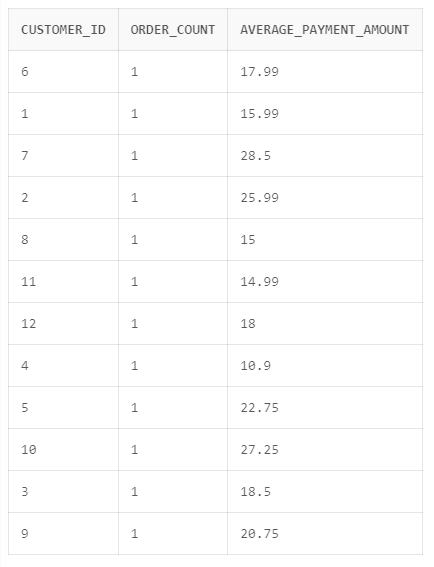
WHERE "order".customer\_id = customer.customer\_id

)

) AS average\_payment\_amount

FROM customer;

SELECT \* FROM CustomerOrderPaymentInfo;



**Sekvencia a trigger:**

* Sekvencia **cat\_primary\_key\_sequence** je nastavená na začiatok s hodnotou 13 a inkrementuje sa o 1 pri každom nasledujúcom použití.
* Trigger **insert\_cat\_primary\_key** sa stará o automatické priradenie hodnôt do stĺpca **cat\_id** v tabuľke Cat pri vkladaní nových záznamov.

CREATE SEQUENCE cat\_primary\_key\_sequence

START WITH 13

INCREMENT BY 1

NOCACHE

NOCYCLE;

-- trigger, ktorý bude vkladať hodnoty do príslušných tabuliek (Cat)

CREATE OR REPLACE TRIGGER insert\_cat\_primary\_key

BEFORE INSERT ON Cat

FOR EACH ROW

BEGIN

IF :NEW.cat\_id IS NULL THEN

:NEW.cat\_id := cat\_primary\_key\_sequence.NEXTVAL;

END IF;

END;

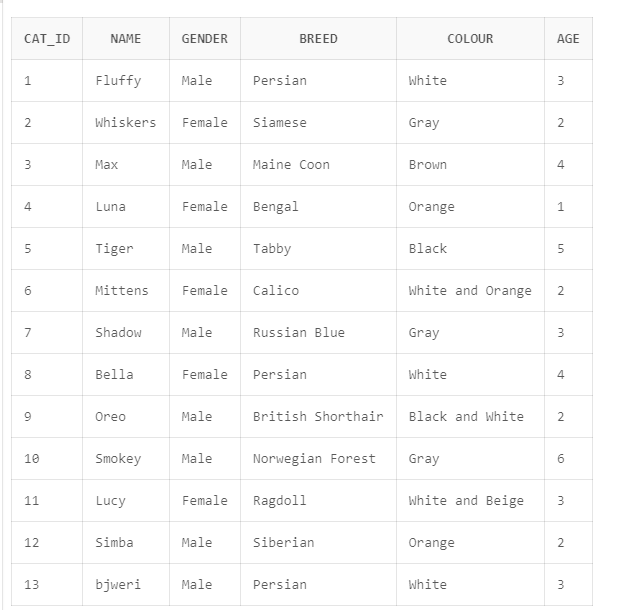
/

-- Vloženie záznamu bez zadania hodnoty cat\_id

INSERT INTO Cat (name, gender, breed, colour, age)

VALUES ('bjweri', 'Male', 'Persian', 'White', 3);

SELECT \* FROM Cat;



**Trigger staff\_changes\_trigger:**

* Tento trigger reaguje na zmeny v tabuľke Staff po vykonaní operácií INSERT, UPDATE alebo DELETE.
* Vypíše správu o detekcii zmeny v tabuľke Staff.

-- Tento trigger reaguje na zmeny v tabuľke Staff po vykonaní operácií INSERT, UPDATE alebo DELETE.

-- Podľa toho, či bola vykonaná operácia INSERT, UPDATE alebo DELETE, sa priradí odpovedajúca operácia.

-- Potom sa vypíše správa o detekcii zmeny v tabuľke Staff.

CREATE OR REPLACE TRIGGER staff\_changes\_trigger

AFTER INSERT OR UPDATE OR DELETE ON Staff

DECLARE

operation VARCHAR2(10);

BEGIN

IF INSERTING THEN

operation := 'INSERT';

ELSIF UPDATING THEN

operation := 'UPDATE';

ELSIF DELETING THEN

operation := 'DELETE';

END IF;

DBMS\_OUTPUT.PUT\_LINE('Change detected in Staff table: ' || operation || ' operation performed.');

END;

/

-- Vloženie nového záznamu do tabuľky Staff

INSERT INTO Staff (Staff\_id, name, position, shift, gender, email, cat\_id)

VALUES (13, 'Test Staff', 'Tester', 'Night', 'Male', 'test@example.com', null);

-- Aktualizácia existujúceho záznamu v tabuľke Staff

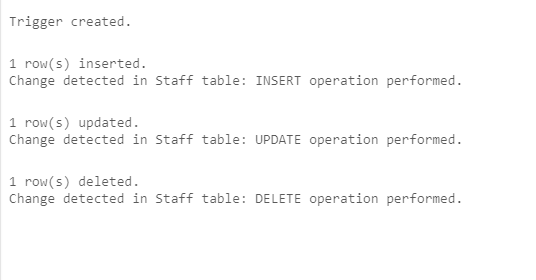
UPDATE Staff

SET position = 'Senior Tester'

WHERE Staff\_id = 13;

-- Odstránenie záznamu z tabuľky Staff

DELETE FROM Staff WHERE Staff\_id = 13;



Zaver

|  |  |
| --- | --- |
| Netrivialne selekty | 1 |
| Spájanie tabuliek | 3 |
| Pouzitie agregacnych funkcií alebo zoskupenia | 2 |
| Mnozinová operácia | 1 |
| Vnorerené selekty | 2 |
| Sekvencia + trigger | 1 |
| Trigger | 1 |
| Celkovo | 11 |