EX1.

|  |  |  |
| --- | --- | --- |
| CPF | NOME | DATA |
| 20002000 | Ana | 2006/06/12 |
| 20002000 | Ana | 2006/06/19 |
| 2200020000 | Lucia | 2006/06/14 |
| 1100110000 | Carlos | 2006/06/14 |
| 1100110000 | Carlos | 2006/06/19 |
| 1100110000 | Carlos | 2006/06/20 |
| 1100110000 | Carlos | 2006/06/22 |

EX2.

|  |
| --- |
| NOME |
| JOÃO |
| CARLOS |

Aluno: Larissa Lorraine

Uc:2200367

ARQUIVO DO MYSQL WORKBECH

SELECT sum(amount) as soma,

count(distinct customer\_id) as qtd\_clientes, -- contar qnts distintos fizeram o pagamento

max(amount) , min(amount)

avg(amount) as media

FROM sakila.payment

where amount not between 5 and 10; -- aqui vai pegar o $ maior que 5 e menor que 10

where customer\_id in (10,12,14,16,18,20);

SELECT distinct customer\_id, -- crie uma consulta que retorne os 10 pgmt (payments) de maior valor

payment\_date

FROM sakila.payment

order by 2 desc,1 -- (ordernar por) ordenando uma lista de colunas ou uma projeção, que ira começar pela segunda coluna

limit 20; -- (desc) pela orderm descrescente

select actor\_id as id,

first\_name as nome,

last\_name as sobrenome, -- filtro

DATEDIFF(current\_date(),last\_update) as dias

from sakila.actor

where first\_name like 'A%A' -- aqui é para consultar se na tabela o primeiro nome começa com 'A' e tem 'a' no nome

and actor\_id between 20 and 30; -- teste

select count(\*)

from sakila.actor

where first\_name like '%A'

SELECT distinct first\_name, last\_update

from sakila.actor

where first\_name like '%A';

SELECT distinct customer\_id, sum(amount) as TotalGasto -- somar

FROM sakila.payment

group by customer\_id -- agrupar os clientes

order by 2 desc -- ordernar pela 2° coluna, decrescente

limit 10; -- e só os 10 clientes

SELECT distinct customer\_id, sum(amount) as Total -- ESTRUTURA TOTAL de um select

FROM sakila.payment

where payment\_date > '2005-05-25' -- where faz linha a linha

group by customer\_id

having sum(amount) > 150 -- having, sempre ter uma função de agregação

order by 2 desc;

SELECT distinct release\_year, title -- atividade

FROM sakila.film

order by title , release\_year;

SELECT release\_year, count(\*)

FROM sakila.film

group by release\_year

order by 1;

SELECT customer\_id sum(amount) as Cliente

FROM sakila.customer

where store\_id > 10

group by customer\_id

order by 1 desc

SELECT \*

FROM sakila.customer, -- PRODUTO CARTESIANO (ele faz uma combinação de duas tabelas)

sakila.payment; -- Join "implicito" um filtro do PD(lig~ção de chave prim. com chave estrang.

SELECT count(\*)

FROM sakila.customer,

sakila.payment;