

Perguntas CUC SQL para contagens de mensagens e tamanhos da caixa postal

Índice

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Bancos de dados](#)

[Tabelas](#)

[Perguntas SQL](#)

[Aliste a contagem total das mensagens com conhecido aliás](#)

[Aliste usuários com contagem total das mensagens](#)

[Aliste usuários com mensagens totais contam baseado no primeiro caráter em aliás](#)

[Aliste usuários com mensagens da caixa de entrada totais](#)

[Aliste usuários com mensagens excluída totais](#)

[Aliste usuários com total, caixa de entrada, e mensagens excluída](#)

[Aliste o contagem de mensagens do usuário com tempo de chegada da mensagem a mais velha](#)

[Aliste o contagem de mensagens do usuário com tempo de chegada da mensagem a mais velha e tamanho da caixa postal/duração do total](#)

[Aliste a caixa de entrada do usuário e os mensagens excluída contam com tempo de chegada da mensagem a mais velha e tamanho da caixa postal/duração do total](#)

[Aliste o número total de mensagens para todas as caixas postais](#)

[Aliste um usuário o tamanho da caixa postal que com envia e recebe limites](#)

[Aliste todo o usuário os tamanhos da caixa postal que com enviam e recebe limites](#)

[Aliste o tamanho total de todas as caixas postais combinadas](#)

Introdução

Este documento descreve como obter o contagem de mensagens e o tamanho de uma caixa postal do usuário com perguntas da língua de consulta estruturada (SQL) através do CLI. Estes dados podem igualmente ser recuperados com a ferramenta da [descarga dos dados do usuário](#), da página das [ferramentas das comunicações unificadas de Cisco](#).

Pré-requisitos

Requisitos

Cisco recomenda que você tem o conhecimento do Cisco Unity Connection (CUC).

Componentes Utilizados

A informação neste documento é baseada em versões 8.X e mais recente CUC, mas esta informação pôde trabalhar para versões anterior também.

Bancos de dados

As perguntas SQL são formadas com os dados destes bases de dados:

- **UnityDirDB** - Este base de dados contém a informação USER-relacionada.
- **UnityMbxDB1** - Este base de dados contém a informação da caixa de correio do usuário.

Tabelas

As perguntas SQL são formadas com os dados nestas vistas. *Uma vista* é uma tabela que seja uma combinação de dois ou mais tabelas, ou os mesmos dados em uma única tabela.

Estas vistas são usadas no base de dados de **UnityDirDB**:

- **vw_mailbox** - Esta vista contém o mapeamento entre o usuário e a caixa postal.
- **vw_user** - Esta vista contém a informação USER-relacionada.

Estas vistas são usadas no base de dados **UnityMbxDB1**:

- **vw_message** - Esta vista contém um artigo da mensagem no sistema. Esta tabela é meramente um suporte das propriedades de mensagem.
- **vw_mailbox** - Esta vista contém uma caixa postal no sistema que guarda mensagens de entrada. Esta tabela contém a informação geral sobre o correio de correio individual, para incluir a maneira em que as mensagens são armazenadas na caixa postal.

Perguntas SQL

Esta seção descreve as várias perguntas SQL que você pode usar em CUC.

Aliste as mensagens totais contam com conhecido aliás

Incorpore este comando a fim obter uma lista das mensagens totais contam com conhecido aliás:

```
admin:run cuc dbquery unitymbxdb1 select count (*) as Messages from vw_message,  
unitydirdb:vw_mailbox, unitydirdb:vw_user where mailboxobjectid in (select
```

```
mailboxid from vw_mailbox where unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid and alias='Anirudh')
```

```
messages
```

```
-----
```

```
3
```

Esta pergunta é uma alta complexidade, a pergunta do dobro-base de dados que envolve tabelas múltiplas. Para server com um base de dados e um tamanho muito grandes da caixa postal, um período de tempo extendido pôde decorrer antes que uma saída apareça, mesmo sobre uma hora, que não fosse ideal. Em tais encenações, você pode usar esta pergunta pelo contrário:

```
admin:run cuc dbquery unitymbxdb1 select count (*) as Messages from vw_message where mailboxobjectid in (select mailboxobjectid from vw_mailbox where description='Anirudh')
```

```
messages
```

```
-----
```

```
3
```

A primeira pergunta retorna os dados quando o **pseudônimo** é mencionado, que é **original**. A segunda pergunta retorna os dados quando a **descrição** é mencionada, que não é **original**.

Nota: Quando a caixa postal é criada, a descrição é a mesma que o pseudônimo; contudo, quando o pseudônimo é atualizado, a descrição não é atualizada. Para bases de dados pequenos, a primeira pergunta é ideal. A fim demonstrar que a descrição não está mudada depois que o pseudônimo é alterado, **test3** é alterado a **Atest3** e usado para as próximas seções.

Usuários da lista com contagem total das mensagens

Incorpore este comando a fim obter uma lista de usuários com a contagem total das mensagens:

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as messages from vw_message, unitydirdb:vw_mailbox, unitydirdb:vw_user where mailboxobjectid in (select mailboxid from vw_mailbox where unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid) group by alias order by messages desc
```

```
userid messages
```

```
-----
```

```
Anirudh 3
```

```
Atest3 2
```

```
undeliverablemessagesmailbox 1
```

Para as mesmas razões que são mencionadas na [lista que as mensagens totais contam com uma seção de pseudônimo conhecida](#), esta pergunta pode igualmente ser usada:

```
admin:run cuc dbquery unitymbxdb1 select description, count (*) as Messages from vw_message, vw_mailbox where vw_mailbox.mailboxobjectid = vw_message.mailboxobjectid group by description order by messages desc
```

```
description messages
```

```
-----
```

```
Anirudh 3
```

```
test3 2
```

Nota: Na segunda pergunta, a descrição não muda de **test3** a **Atest3** depois que o pseudônimo é mudado.

Os usuários da lista com mensagens totais contam baseado no primeiro caráter em aliás

Incorpore este comando a fim obter uma lista de usuários com as mensagens totais contam baseado no primeiro caráter de um pseudônimo:

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as messages from
vw_message, unitydirdb:vw_mailbox, unitydirdb:vw_user where deleted='0' and
mailboxobjectid in (select mailboxid from vw_mailbox where unitydirdb:
vw_user.objectid = unitydirdb:vw_mailbox.userobjectid) and alias like 'A%' group by
alias order by messages
```

```
userid messages
-----
Atest3 2
Anirudh 3
```

Para as mesmas razões que são mencionadas na [lista que as mensagens totais contam com uma seção de pseudônimo conhecida](#), esta pergunta pode igualmente ser usada:

```
admin:run cuc dbquery unitymbxdb1 select description, count (*) as Messages from
vw_message, vw_mailbox where vw_mailbox.mailboxobjectid =
vw_message.mailboxobjectid and description like 'A%' group by description order
by messages
```

Estão aqui algumas observações importantes sobre esta pergunta:

- O **A%** ajusta a pergunta para bater os pseudônimos que começam com a letra **A**.
- O formato é **onde o columnname** gosta da “circunstância”. Aqui, o nome de coluna é **aliás** para a primeira pergunta e **descrição** para a segunda pergunta.

Estão aqui algumas condições do exemplo:

- **_n%** - A primeira letra pode ser todo o caráter (um convite), seguido pela letra **n** e por qualquer número de caracteres.
- **%s** - Isto ajusta a pergunta para bater os pseudônimos que terminam com a letra **S**.

As perguntas que são mencionadas até aqui são usadas a fim obter as mensagens totais (caixa de entrada e artigos suprimido). A próxima seção descreve as perguntas que são usadas a fim obter o número total de mensagens na caixa de entrada e nos artigos suprimido.

Usuários da lista com mensagens da caixa de entrada totais

Incorpore este comando a fim obter uma lista de usuários com os mensagens da caixa de entrada totais:

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as
inboxmessages from vw_message, unitydirdb:vw_mailbox, unitydirdb:vw_user
where deleted='0' and mailboxobjectid in (select mailboxid from vw_mailbox
where unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid) and
alias like 'A%' group by alias order by inboxmessages
```

```
userid inboxmessages
```

```
-----
Atest3 2
Anirudh 3
```

Para as mesmas razões que são mencionadas na [lista que as mensagens totais contam com uma seção de pseudônimo conhecida](#), esta pergunta pode igualmente ser usada:

```
admin:run cuc dbquery unitymbxdb1 select description, count (*) as InboxMessages
from vw_message, vw_mailbox where vw_mailbox.mailboxobjectid =
vw_message.mailboxobjectid and deleted = '0' and description like 'A%' group by
description order by InboxMessages
```

Estão aqui algumas observações importantes sobre esta pergunta:

- O **A%** ajusta a pergunta para bater os pseudônimos que começam com a letra **A**.
- O formato é **onde o columnname gosta da “circunstância”**. Aqui, o nome de coluna é **aliás** para a primeira pergunta e **descrição** para a segunda pergunta.

Estão aqui algumas condições do exemplo:

- **_n%** - A primeira letra pode ser todo o caráter (um convite), seguido pela letra **n** e por qualquer número de caracteres.
- **%s** - Isto ajusta a pergunta para bater os pseudônimos que terminam com a letra **S**.

Nota: Neste exemplo, uma circunstância é usada a fim limitar usuários com um pseudônimo/descrição que comece com a letra **A**.

Estas são algumas variações desta pergunta:

- A fim alistar todos os usuários, remova **e aliás como “A%”** na primeira pergunta, ou **e na descrição como “A%”** na segunda pergunta.
- A fim alistar um usuário particular (aliste a contagem para a caixa postal de Anirudh, por exemplo), substitui-lo **e aliás como “A%”** com **e alias='Anirudh** na primeira pergunta, ou substitui-lo **e na descrição como “A%”** com **e descrição = ' Anirudh** na segunda pergunta. Assegure-se de que a substituição esteja na posição exata, ou a pergunta falha.

Aliste usuários com mensagens excluída totais

Incorpore este comando a fim obter uma lista de usuários com os mensagens excluída totais:

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as
deletedmessages from vw_message, unitydirdb:vw_mailbox, unitydirdb:vw_user
where deleted='1' and mailboxobjectid in (select mailboxid from vw_mailbox
```

```
where unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid) and
alias like 'A%' group by alias order by deletedmessages
```

No records found

Para as mesmas razões que são mencionadas na [lista que as mensagens totais contam com uma](#) seção de [pseudônimo conhecida](#), esta pergunta pode igualmente ser usada:

```
admin:run cuc dbquery unitymbxdb1 select description, count (*) as
deletedmessages from vw_message, vw_mailbox where vw_mailbox.mailboxobjectid=
vw_message.mailboxobjectid and deleted = '1' and description like 'A%' group
by description order by deletedmessages
```

Nota: Neste exemplo não há nenhuma mensagem excluída, assim que a saída aparece porque **nenhum registro encontrou**.

Estão aqui algumas observações importantes sobre esta pergunta:

- O **A%** ajusta a pergunta para bater os pseudônimos que começam com a letra **A**.
- O formato é **onde o columnname gosta da “circunstância”**. Aqui, o nome de coluna é **aliás** para a primeira pergunta e **descrição** para a segunda pergunta.

Estão aqui algumas condições do exemplo:

- **_n%** - A primeira letra pode ser todo o caráter (um convite), seguido pela letra **n** e por qualquer número de caracteres.
- **%s** - Isto ajusta a pergunta para bater os pseudônimos que terminam com a letra **S**.

Nota: Neste exemplo, uma circunstância é usada a fim limitar usuários com um pseudônimo/descrição que comece com a letra **A**.

Estas são algumas variações desta pergunta:

- A fim alistar todos os usuários, remova **e aliás como “A%”** na primeira pergunta, ou **e na descrição como “A%”** na segunda pergunta.
- A fim alistar um usuário particular (aliste a contagem para a caixa postal de Anirudh, por exemplo), substitui-lo **e aliás como “A%”** com **e alias='Anirudh** na primeira pergunta, ou substitui-lo **e na descrição como “A%”** com **e descrição = ' Anirudh** na segunda pergunta. Assegure-se de que a substituição esteja na posição exata, ou a pergunta falha.

Aliste usuários com total, caixa de entrada, e mensagens excluída

Incorpore este comando a fim obter uma lista de usuários com o total, a caixa de entrada, e os mensagens excluída:

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as messages,
sum(case when deleted='0' then 1 else 0 end) as Inboxmessages, sum(case when
deleted='1' then 1 else 0 end) as Deletedmessages from vw_message, unitydirdb:
vw_mailbox, unitydirdb:vw_user where mailboxobjectid in (select mailboxid from
```

```
vw_mailbox where unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid)
group by alias order by messages desc
```

```
userid messages inboxmessages deletedmessages
```

```
-----
```

```
Anirudh 3 3 0
```

```
Atest3 2 2 0
```

```
undeliverablemessagesmailbox 1 1 0
```

Para as mesmas razões que são mencionadas na [lista que as mensagens totais contam com uma seção de pseudônimo conhecida](#), esta pergunta pode igualmente ser usada:

```
admin:run cuc dbquery unitymbxdb1 select description as UserID, count (*) as
messages, sum(case when deleted='0' then 1 else 0 end) as Inboxmessages, sum
(case when deleted='1' then 1 else 0 end) as Deletedmessages from vw_mailbox
join vw_message on vw_message.mailboxobjectid=vw_mailbox.mailboxobjectid
group by description order by messages desc
```

Estão aqui algumas observações importantes sobre esta pergunta:

- O **A%** ajusta a pergunta para bater os pseudônimos que começam com a letra **A**.
- O formato é **onde o columnname gosta da “circunstância”**. Aqui, o nome de coluna é **aliás** para a primeira pergunta e **descrição** para a segunda pergunta.

Estão aqui algumas condições do exemplo:

- **_n%** - A primeira letra pode ser todo o caráter (um convite), seguido pela letra **n** e por qualquer número de caracteres.
- **%s** - Isto ajusta a pergunta para bater os pseudônimos que terminam com a letra **S**.

Nota: Neste exemplo, uma circunstância é usada a fim limitar usuários com um pseudônimo/descrição que comece com a letra **A**.

Estas são algumas variações desta pergunta:

- A fim alistar todos os usuários, remova **e aliás como “A%”** na primeira pergunta, ou **e na descrição como “A%”** na segunda pergunta.
- A fim alistar um usuário particular (aliste a contagem para a caixa postal de Anirudh, por exemplo), substitui-lo **e aliás como “A%”** com **e alias='Anirudh** na primeira pergunta, ou substitui-lo **e na descrição como “A%”** com **e descrição = ' Anirudh** na segunda pergunta. Assegure-se de que a substituição esteja na posição exata, ou a pergunta falha.

Aliste o contagem de mensagens do usuário com tempo de chegada da mensagem a mais velha

Esta pergunta pode ser usada a fim determinar se os trabalhos programados para a limpeza do mailboxe tomam o efeito:

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as messages,
min(arrivaltime) as OldestMessageTime from vw_message, unitydirdb:vw_mailbox,
unitydirdb:vw_user where mailboxobjectid in (select mailboxid from vw_mailbox
```

```
where unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid) group
by alias order by messages desc
```

```
userid messages oldestmessagetime
```

```
-----
```

```
Anirudh 3 2013-03-19 14:38:14.459
```

```
Atest3 2 2013-01-18 05:49:45.355
```

```
undeliverablemessagesmailbox 1 2012-07-05 01:10:19.961
```

Para as mesmas razões que são mencionadas na [lista que as mensagens totais contam com uma seção de pseudônimo conhecida](#), esta pergunta pode igualmente ser usada:

```
admin:run cuc dbquery unitymbxdb1 select description, count (*) as Messages,
min(arrivaltime) as OldestMessageTime from vw_message, vw_mailbox where
vw_mailbox.mailboxobjectid = vw_message.mailboxobjectid group by description
order by messages desc
```

Nota: A fim obter o último tempo de chegada para as perguntas nas seções anterior, adicionar o `min(arrivaltime)` como `OldestMessageTime` imediatamente depois da **contagem (*)** como **mensagens**.

Estas são algumas variações desta pergunta:

- A fim alistar todos os usuários com pseudônimos que começam com a letra A, adicionar **e aliás como "A%"** na primeira pergunta imediatamente antes do **grupo pela** condição do **pseudônimo**, ou **e da descrição como "A%"** na segunda pergunta imediatamente antes do **grupo pela** condição da **descrição**. Assegure-se de que a substituição esteja na posição exata, ou a pergunta falha.
- A fim alistar um usuário particular (aliste a contagem para a caixa postal de Anirudh, por exemplo), adicionar **e aliás='Anirudh** na primeira pergunta imediatamente antes do **grupo pela** condição do **pseudônimo**, ou **e pela descrição = ' Anirudh** na segunda pergunta imediatamente antes do **grupo pela** condição da **descrição**. Assegure-se de que a substituição esteja na posição exata, ou a pergunta falha.

Aliste o contagem de mensagens do usuário com tempo de chegada da mensagem a mais velha e tamanho da caixa postal/duração do total

Incorpore este comando a fim obter uma lista do contagem de mensagens do usuário com o tempo de chegada de mensagem a mais velha e o tamanho da caixa postal (sem duração total):

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as messages,
min(arrivaltime) as OldestMessageTime, vw_mailbox.bytesize from vw_message,
vw_mailbox, unitydirdb:vw_mailbox, unitydirdb:vw_user where
vw_message.mailboxobjectid=vw_mailbox.mailboxobjectid and
vw_mailbox.mailboxobjectid in (select mailboxid from vw_mailbox where
unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid) group by
alias, vw_mailbox.bytesize order by messages desc
```

```
userid messages oldestmessagetime bytesize
```

```
-----
```

```
Anirudh 3 2013-03-19 14:38:14.459 93319
```

```
Atest3 2 2013-01-18 05:49:45.355 59890
```

```
undeliverablemessagesmailbox 1 2012-07-05 01:10:19.961 317003
```


Nota: Para obter a duração total das mensagens: adicionar “, **sum(duration/1000) como TotalDuration_In_sec**” imediatamente antes de “do **vw_message**”. Não esqueça a vírgula antes da soma. Isto pode igualmente ser usado para as perguntas nas seções anterior.

Para as mesmas razões que são mencionadas na [lista que as mensagens totais contam com uma seção de pseudônimo conhecida](#), esta pergunta pode igualmente ser usada:

```
admin:run cuc dbquery unitymbxdb1 select description, count (*) as Messages,
min(arrivaltime) as OldestMessageTime, vw_mailbox.bytesize from vw_message,
vw_mailbox where vw_mailbox.mailboxobjectid = vw_message.mailboxobjectid
group by description, vw_mailbox.bytesize order by messages desc
```

Nota: Em ordem para obter a duração total das mensagens: adicionar “, **sum(duration/1000) como TotalDuration_In_sec**” imediatamente antes de “do **vw_message**”. Não esqueça a vírgula antes da soma. Isto pode igualmente ser usado para as perguntas nas seções anterior.

Estas são algumas variações desta pergunta:

- A fim alistar todos os usuários com pseudônimos que começam com a letra A, adicionar **e aliás como “A%”** na primeira pergunta imediatamente antes do **grupo pela** condição do **pseudônimo**, ou **e da descrição como “A%”** na segunda pergunta imediatamente antes do **grupo pela** condição da **descrição**. Assegure-se de que a substituição esteja na posição exata, ou a pergunta falha.
- A fim alistar um usuário particular (aliste a contagem para a caixa postal de Anirudh, por exemplo), adicionar **e alias='Anirudh'** na primeira pergunta imediatamente antes do **grupo pela** condição do **pseudônimo**, ou **e pela descrição = ' Anirudh'** na segunda pergunta imediatamente antes do **grupo pela** condição da **descrição**. Assegure-se de que a substituição esteja na posição exata, ou a pergunta falha.

Aliste a caixa de entrada do usuário e os mensagens excluída contam com tempo de chegada da mensagem a mais velha e tamanho da caixa postal/duração do total

Incorpore este comando a fim obter uma lista da caixa de entrada do usuário e os mensagens excluída contam com o tempo de chegada de mensagem a mais velha e o tamanho da caixa postal (sem duração total):

```
admin:run cuc dbquery unitymbxdb1 select alias as UserID, count (*) as
TotalMessages, sum(case when deleted='0' then 1 else 0 end) as Inbox,
sum(case when deleted='1' then 1 else 0 end) as Deleted, min
(arrivaltime) as OldestMessageTime, vw_mailbox.bytesize from vw_message,
vw_mailbox, unitydirdb:vw_mailbox, unitydirdb:vw_user where
vw_message.mailboxobjectid=vw_mailbox.mailboxobjectid and
vw_mailbox.mailboxobjectid in (select mailboxid from vw_mailbox where
unitydirdb:vw_user.objectid = unitydirdb:vw_mailbox.userobjectid) group
by alias, vw_mailbox.bytesize order by TotalMessages desc
```

```
userid total inbox deleted oldestmessagetime byte
messages size
```

Anirudh 3 3 0 2013-03-19 14:38:14.459 93319
Atest3 2 2 0 2013-01-18 05:49:45.355 59890
undeliverable 1 1 0 2012-07-05 01:10:19.961 317003
messagesmailbox

Nota: A fim obter a duração total das mensagens: adicionar “, **sum(duration/1000) como TotalDuration_In_sec**” imediatamente antes de “do **vw_message**”. Não esqueça a vírgula antes da soma. Isto pode igualmente ser usado para as perguntas nas seções anterior.

Para as mesmas razões que são mencionadas na [lista que as mensagens totais contam com uma](#) seção de [pseudônimo conhecida](#), esta pergunta pode igualmente ser usada:

```
admin:run cuc dbquery unitymbxdb1 select description, count (*) as
TotalMessages, sum(case when deleted='0' then 1 else 0 end) as Inbox,
sum(case when deleted='1' then 1 else 0 end) as Deleted, min(arrivaltime)
as OldestMessageTime, vw_mailbox.bytesize from vw_message, vw_mailbox
where vw_mailbox.mailboxobjectid = vw_message.mailboxobjectid group by
description, vw_mailbox.bytesize order by TotalMessages desc
```

Nota: Para obter a duração total das mensagens: adicionar “, **sum(duration/1000) como TotalDuration_In_sec**” imediatamente antes de “do **vw_message**”. Não esqueça a vírgula antes da soma. Isto pode igualmente ser usado para as perguntas nas seções anterior.

Estas são algumas variações desta pergunta:

- A fim alistar todos os usuários com pseudônimos que começam com a letra A, adicionar **e aliás como “A%”** na primeira pergunta imediatamente antes do **grupo pela** condição do **pseudônimo**, ou **e da descrição como “A%”** na segunda pergunta imediatamente antes do **grupo pela** condição da **descrição**. Assegure-se de que a substituição esteja na posição exata, ou a pergunta falha.
- A fim alistar um usuário particular (aliste a contagem para a caixa postal de Anirudh, por exemplo), adicionar **e alias='Anirudh** na primeira pergunta imediatamente antes do **grupo pela** condição do **pseudônimo**, ou **e pela descrição = ' Anirudh** na segunda pergunta imediatamente antes do **grupo pela** condição da **descrição**. Assegure-se de que a substituição esteja na posição exata, ou a pergunta falha.

Aliste o número total de mensagens para todas as caixas postais

Incorpore este comando a fim obter o número total de mensagens para todas as caixas postais combinadas:

```
admin:run cuc dbquery unitymbxdb1 select count(*) as messages
from vw_message
```

messages

6

Aliste um usuário o tamanho da caixa postal que com envia e recebe limites

Incorpore este comando a fim obter o usuário o tamanho da caixa postal que com envia e receba limites:

```
admin:run cuc dbquery unitydirdb select alias as UserID,bytesize,send,receive,
warning from vw_user,unitymbxdb1:vw_mailbox where vw_user.objectid in (select
userobjectid from vw_usermailboxmap where
vw_usermailboxmap.mailboxid=unitymbxdb1:vw_mailbox.mailboxobjectid and
alias='Anirudh')
```

```
userid bytesize send receive warning
-----
Anirudh 93319 13000000 14745600 12000000
```

Para as mesmas razões que são mencionadas na [lista que as mensagens totais contam com uma seção de pseudônimo conhecida](#), esta pergunta pode igualmente ser usada:

```
admin:run cuc dbquery unitymbxdb1 select description, bytesize,send,receive,
warning from vw_mailbox where description ='Anirudh'
```

Aliste todo o usuário os tamanhos da caixa postal que com enviam e receba limites

Incorpore este comando a fim obter todo o o usuário os tamanhos da caixa postal que com enviam e receba limites:

```
admin:run cuc dbquery unitydirdb select alias as UserID,bytesize,send,receive,
warning from vw_user,unitymbxdb1:vw_mailbox where vw_user.objectid in (select
userobjectid from vw_usermailboxmap where
vw_usermailboxmap.mailboxid=unitymbxdb1:vw_mailbox.mailboxobjectid) order by
bytesize desc
```

```
userid bytesize send receive warning
-----
undeliverablemessagesmailbox 317003 13000000 14745600 12000000
Anirudh 93319 13000000 14745600 12000000
Atest3 59890 13000000 14745600 12000000
Solomon 0 13000000 14745600 12000000
UnityConnection 0 50000000 50000000 45000000
Suvir 0 13000000 14745600 12000000
dsas 0 13000000 14745600 12000000
test1 0 13000000 14745600 12000000
Atest2 0 13000000 14745600 12000000
operator 0 13000000 14745600 12000000
```

Para as mesmas razões que são mencionadas na [lista que as mensagens totais contam com uma seção de pseudônimo conhecida](#), esta pergunta pode igualmente ser usada:

```
admin:run cuc dbquery unitymbxdb1 select description, bytesize,send,receive,
warning from vw_mailbox order by bytesize desc
```

Como uma variação desta pergunta a fim alistar todos os usuários com pseudônimos que começam com a letra A, adicionar e aliás como "A%" na primeira pergunta imediatamente depois do vw_usermailboxmap.mailboxid=unitymbxdb1:vw_mailbox.mailboxobjectid e antes) ordem pela circunstância, ou você pode adicionar onde descrição como "A%" na segunda pergunta imediatamente antes da ordem pela circunstância. Assegure-se de que isto esteja

adicionado na posição correta, ou a pergunta falha.

Aliste o tamanho total de todas as caixas postais combinadas

Incorpore este comando a fim obter o tamanho total de todas as caixas postais combinadas:

```
admin:run cuc dbquery unitymbxdb1 select sum (bytesize) from vw_mailbox
```

```
(sum)
```

```
-----
```

```
2683210
```

```
admin:
```