



# **ADMINISTRACIÓN DE LA SEGURIDAD DE BASE DE DATOS**

Copias de seguridad y restauración de Base de Datos.

ING. JAIME LLANOS BARDALES

# Introducción

- Quiz



Universidad  
Nacional de  
Cajamarca  
*"Norte de la Universidad Peruana"*



# Poll Everywhere

# Agenda

*Copia Seguridad, Estándares de seguridad, Jobs, Índices*

Objetivo:

- Copias seguridad
- Estándares de Seguridad
- Jobs
- Índices



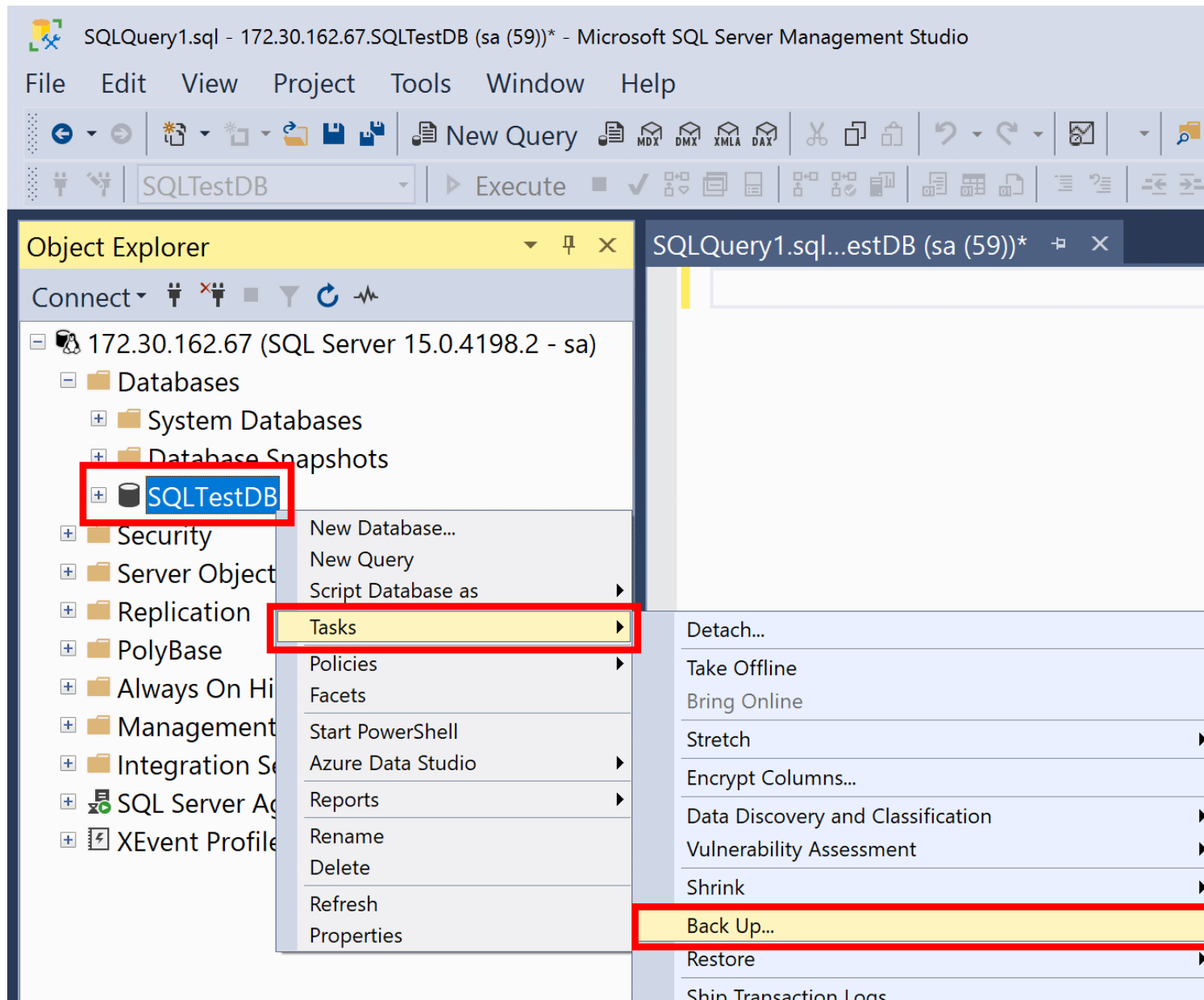
Universidad  
Nacional de  
Cajamarca

"Norte de la Universidad Peruana"



Universidad  
Nacional de  
Cajamarca  
*"Norte de la Universidad Peruana"*

# Seguridad en Bases de datos Copias de respaldo (backup).





## Copias de seguridad

Las copias de seguridad de SQL Server proveen una importante solución para proteger datos críticos que están almacenados en bases de datos SQL para minimizar el riesgo de pérdida de datos.

Es una buena práctica probar sus copias de seguridad restaurando archivos de copias de seguridad al azar a un ambiente de pruebas y verificar que los archivos no estén corruptos

# Tipos de copias de seguridad



Universidad  
Nacional de  
Cajamarca  
*"Norte de la Universidad Peruana"*

Copia de seguridad completa

Copia de seguridad diferencial

Copia de seguridad de registro de transacciones

## Copia de seguridad completa

Una copia de seguridad completa contiene todos los datos de una base de datos específica o un conjunto de grupos de archivos o archivos, y también el registro suficiente para permitir la recuperación de esos datos.

Es la base de la copia de seguridad diferencial y de la copia de seguridad de los registros de transacciones.



## Copia de seguridad diferencial

Una copia de seguridad diferencial no es independiente y debe basarse en la última copia de seguridad completa de los datos. Eso significa que debería haber una copia de seguridad completa como base. Una copia de seguridad diferencial contiene sólo los datos que han cambiado desde la base diferencial.

Normalmente, las copias de seguridad diferenciales son más pequeñas y rápidas de crear que la base de una copia de seguridad completa y también requieren menos espacio en disco para almacenar imágenes de copia de seguridad

## Copia de seguridad del registro de transacciones

El registro de transacciones es un registro en serie de todas las transacciones que se han realizado contra la base de datos desde que se realizó la última copia de seguridad del registro de transacciones

Con las copias de seguridad del registro de transacciones, puede recuperar la base de datos hasta un momento específico (por ejemplo, antes de introducir datos no deseados) o hasta el punto de fallo.

# Copia de seguridad del registro de transacciones

Las copias de seguridad del registro de transacciones sólo son valiosas bajo el modelo de recuperación completa o el modelo de recuperación en bloque. Cada copia de seguridad de registro cubre la parte del registro de transacciones que estaba activa cuando se creó la copia de seguridad, e incluye todos los registros de registro que no se copiaron en una copia de seguridad de registro anterior.

Una secuencia ininterrumpida de copias de seguridad de registros contiene la cadena completa de registros de la base de datos, que se dice que no está rota. En el modelo de recuperación completa, y a veces en el modelo de recuperación de registros masivos, una cadena de registro ininterrumpida le permite restaurar la base de datos a cualquier punto en el tiempo



Universidad  
Nacional de  
Cajamarca  
*"Norte de la Universidad Peruana"*

# Restauración y recuperación de una base de datos



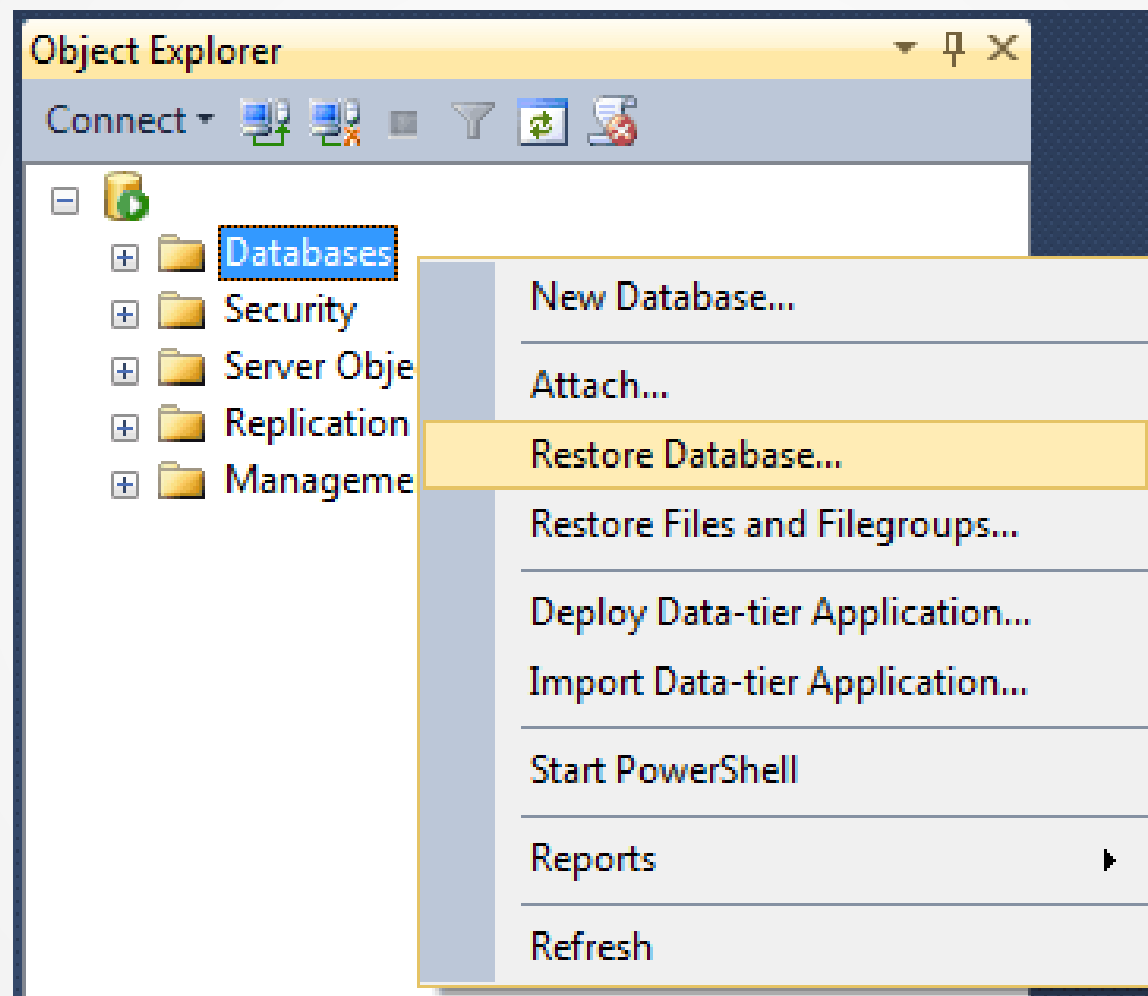


## Restaurar

Es cargar a una base de datos uno o varios objetos de una base de datos desde una copia de seguridad de esa base de datos o de esos objetos. La restauración sobrescribe cualquier información de la base de datos con la información de la copia de seguridad. Después de restaurar una base de datos, deberá recuperarla

## Recuperar

Es devolver una base de datos restaurada a un momento dado consistente anterior al momento en que se produjo el daño o fallo. Una vez que la base de datos se haya restaurado y recuperado correctamente, estará lista para su uso. Puede realizar recuperaciones tanto automáticas como manuales





# Ejercicio Base de datos de Ejemplo



[AdventureWorks2022](#)





Universidad  
Nacional de  
Cajamarca

*"Norte de la Universidad Peruana"*

# Estándares existentes para la seguridad de la información

# Definición



**Universidad  
Nacional de  
Cajamarca**  
*"Norte de la Universidad Peruana"*

**Seguridad de la  
Información**

Consiste en la preservación de su confidencialidad, integridad y disponibilidad, así como de los sistemas implicados en su tratamiento, dentro de una organización.



# POR QUE USAR ESTANDARES

Demuestra la garantía independiente de los controles internos y cumple los requisitos de gestión corporativa y de continuidad de la actividad comercial.



Demuestra independientemente que se respetan las leyes y normativas que sean de aplicación.



Proporciona una ventaja competitiva al cumplir los requisitos contractuales y demostrar a los clientes que la seguridad de su información es primordial.



Verifica independientemente que los riesgos de la organización estén correctamente identificados, evaluados y gestionados al tiempo que formaliza unos procesos, procedimientos y documentación de protección de la información.



Demuestra el compromiso de la cúpula directiva de su organización con la seguridad de la información.



El proceso de evaluaciones periódicas ayuda a supervisar continuamente el rendimiento y la mejora.

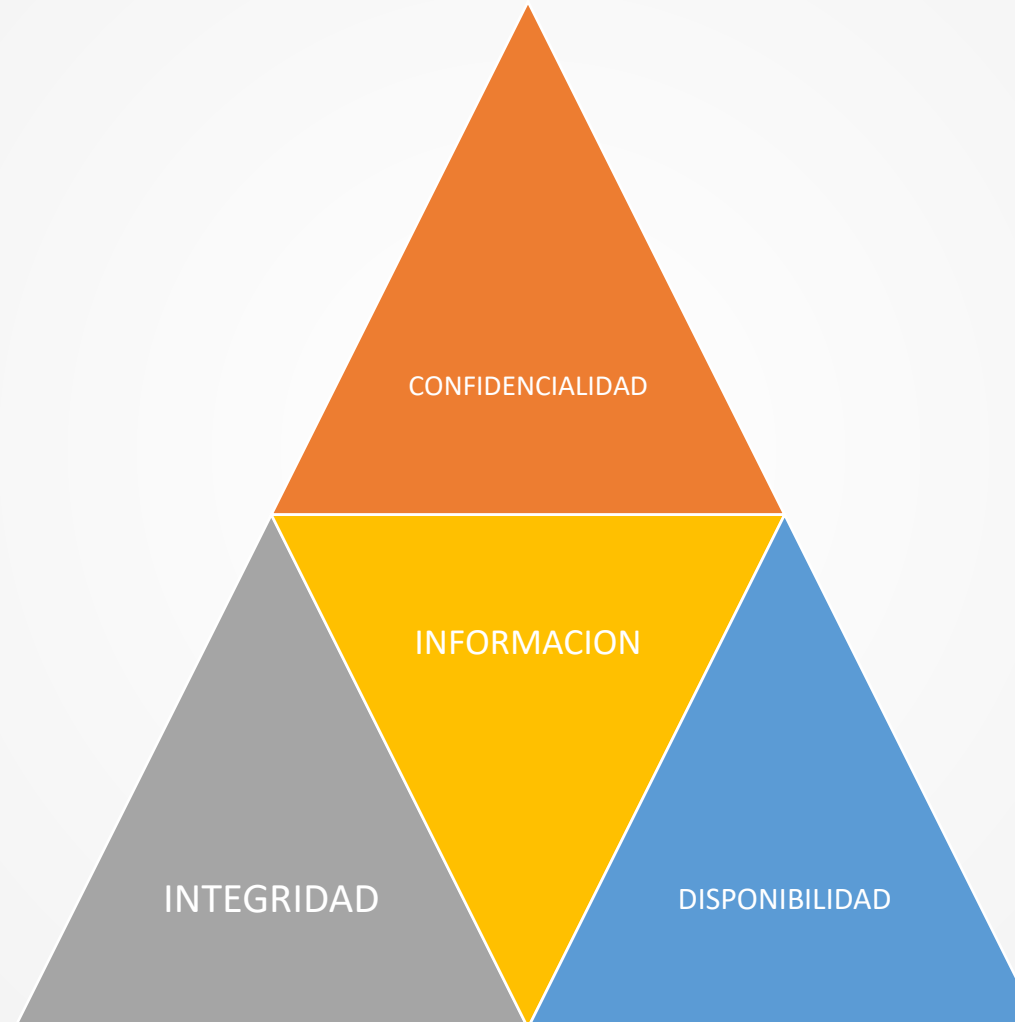
# BASES PARA LA SEGURIDAD DE LA INFORMACIÓN



Universidad  
Nacional de  
Cajamarca  
"Norte de la Universidad Peruana"

**FIABILIDAD:** La fiabilidad de la información es un sistema en el cual se compone de cómo se comporte y que haga lo que se espera de él.

Este en un sistema será seguro si y fiable si se compone de estas características:





# BASES PARA LA SEGURIDAD DE LA INFORMACIÓN

- **Confidencialidad:** acceso a la información solo mediante autorización y de forma controlada. El objetivo de la confidencialidad es, entonces, prevenir la divulgación no autorizada de la información.
- **Integridad:** modificación de la información solo mediante autorización. El objetivo de la integridad es, entonces, prevenir modificaciones no autorizadas de la información.
- **Disponibilidad:** la información del sistema debe permanecer accesible mediante autorización. El objetivo de la disponibilidad es, entonces, prevenir interrupciones no autorizadas/controladas de los recursos informáticos.



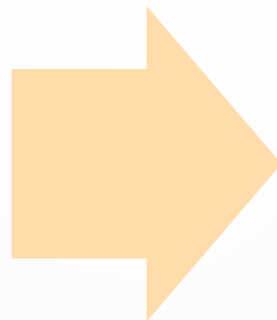


Universidad  
Nacional de  
Cajamarca  
"Norte de la Universidad Peruana"



# Definición SGSI

**Sistema de Gestión de la  
Seguridad de la  
Información (SGSI)**



Es un conjunto de políticas y procedimientos que sirven para estandarizar la gestión de la Seguridad de la Información. A continuación les dejamos algunos detalles de cada uno de los estándares que están incluidos en la familia de ISO 27000.

# ISO 27000



Universidad  
Nacional de  
Cajamarca  
"Norte de la Universidad Peruana"

## Familia de Normas ISO 27000

### Vocabulario Común

ISO 27000

[MAS INFO](#)

### Requisitos

ISO 27001

ISO 27006

ISO 27009

*SGSI*

### Guías de aplicación

ISO 27002

ISO 27003

ISO 27004

ISO 27005

ISO 27007

TR 27008

ISO 27013

ISO 27014

TR 27016

ISO 27021

### Requisitos específicos

ISO 27010

ISO 27011

ISO 27017

ISO 27018

ISO 27019



Universidad  
Nacional de  
Cajamarca  
*"Norte de la Universidad Peruana"*

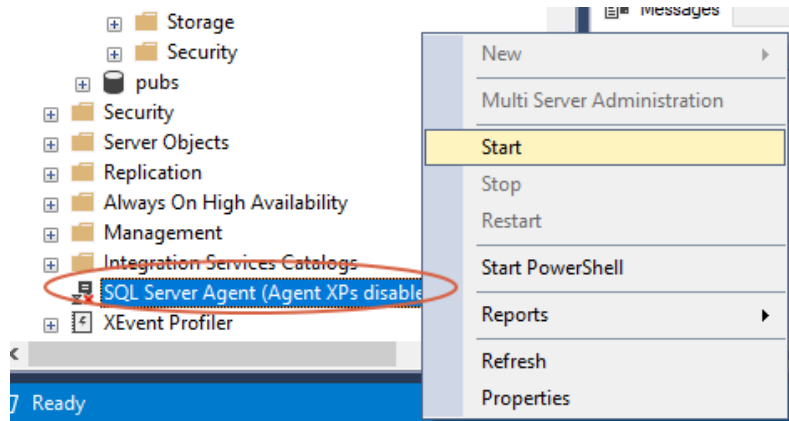
# SQL Server Agent

# Job SQL Server

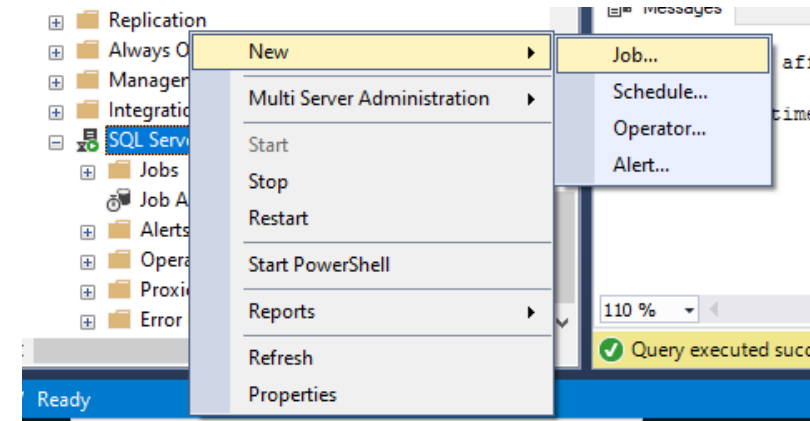


Universidad  
Nacional de  
Cajamarca  
"Norte de la Universidad Peruana"

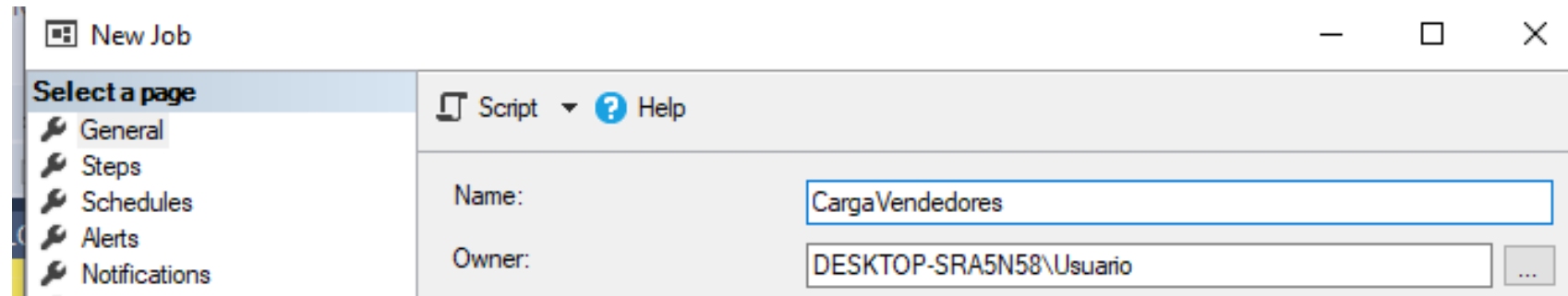
Habilitar el  
Servicio



Crear un  
Job

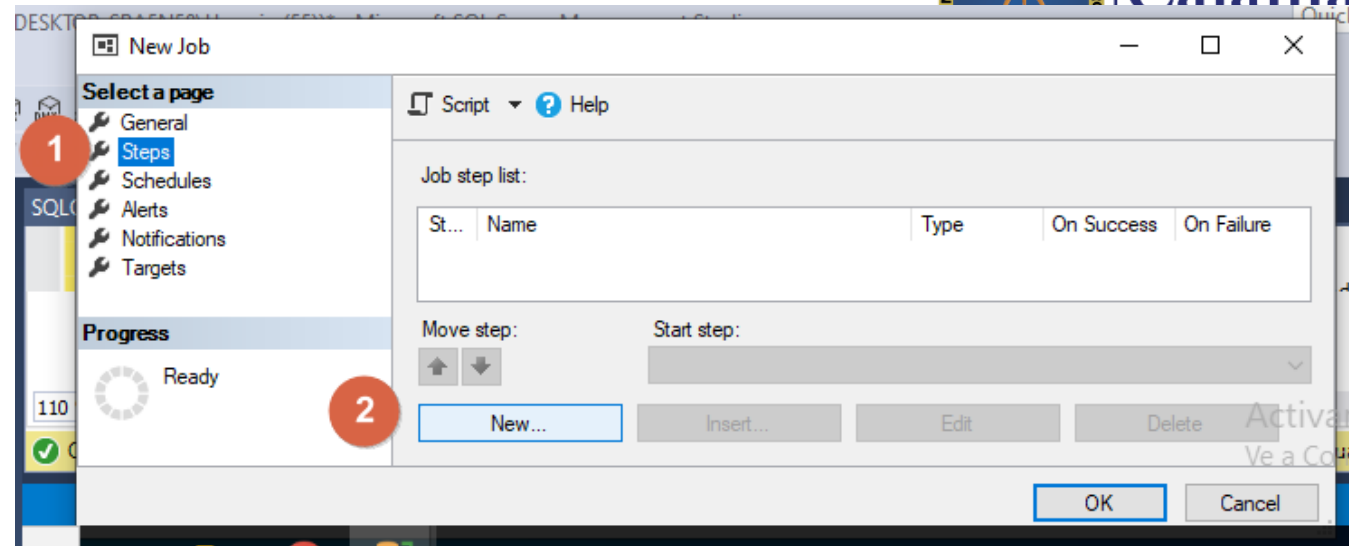


Asignarle  
un nombre



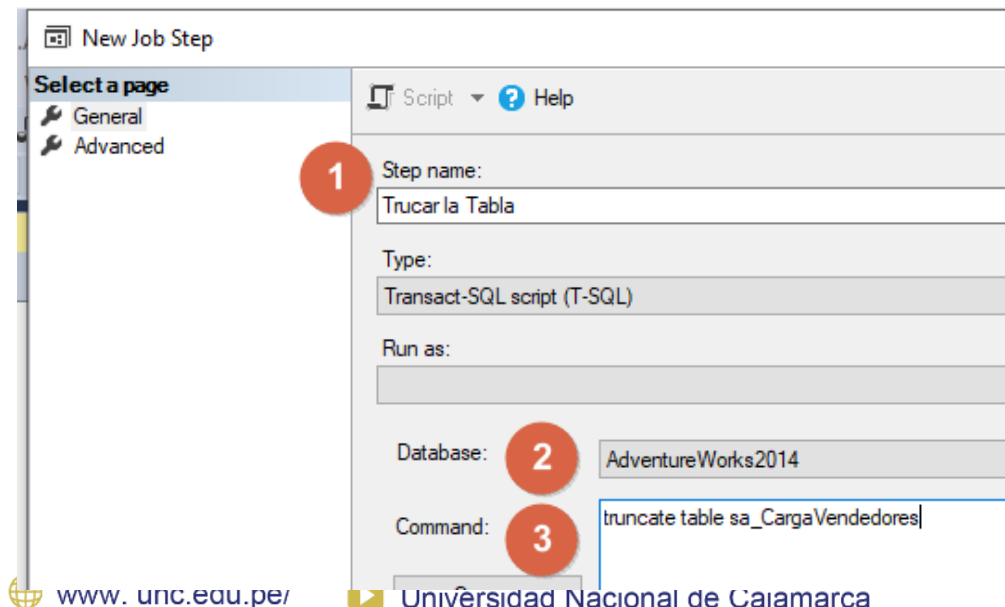
# Job SQL Server

Cree un  
nuevo Paso

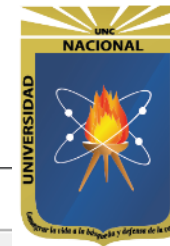


En el paso asigne:

- Nombre
- Seleccione la BD
- Script a ejecutar







# Job SQL Server

Cree el nuevo paso para  
insertar los datos

Job Step Properties - CargarTabla

Select a page

- General
- Advanced

Script Help

Step name: CargarTabla

Type: Transact-SQL script (T-SQL)

Run as:

Database: AdventureWorks2014

Command:

```
insert into sa_CargaVendedores  
select (date, OrderDate) OrderDate,  
from [Sales].[SalesOrderHeader] so  
join [Sales].[SalesOrderDetail] od on od.SalesOrderID = so.SalesOrderID  
join [Sales].[SalesTerritory] t on t.TerritoryID = so.TerritoryID  
join [Sales].[SalesPerson] v on v.BusinessEntityID = so.SalesPersonID  
join [Production].[Product] p on p.ProductID = so.ProductID
```

Open... Select All

El Job tiene ahora los dos  
pasos a ejecutar.

Job Properties - CargaVendedores

Select a page

- General
- Steps
- Schedules
- Alerts
- Notifications
- Targets

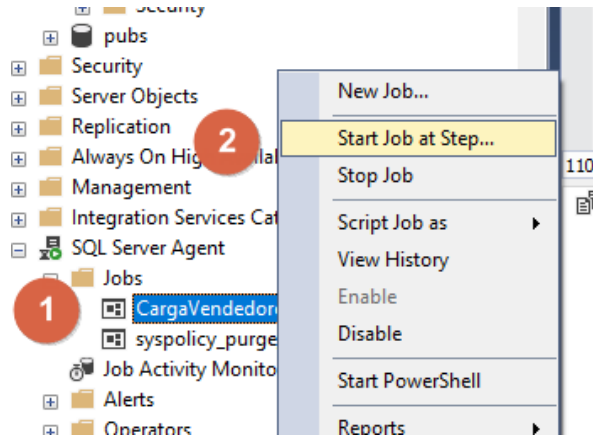
Script Help

Job step list:

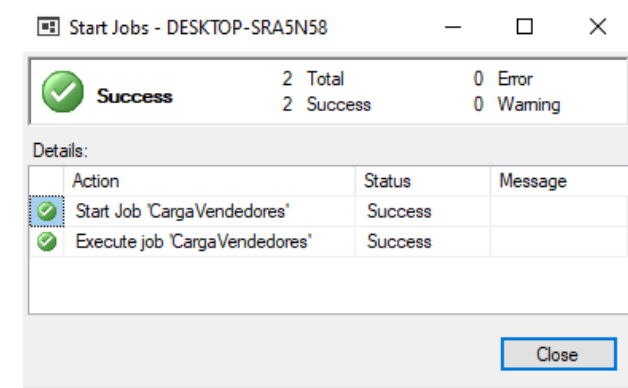
St...	Name	Type	On Success	On Failure
1	Trucar la Tabla	Transact-...	Quit the jo...	Quit the job...
2	CargarTabla	Transact-...	Go to the ...	Quit the job...

# Job SQL Server

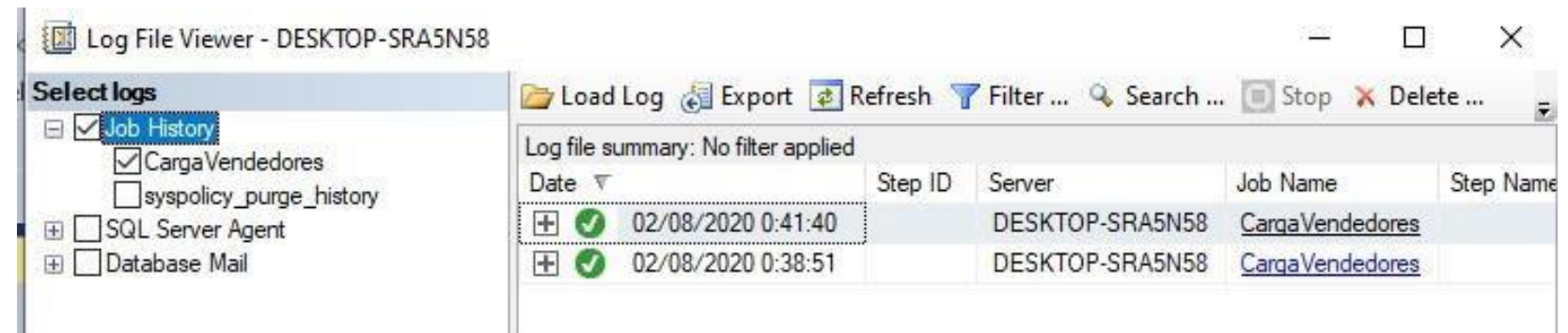
Elija el Job e inicie su ejecución



El Job indica que ejecuto los pasos correctamente



El Job guarda el historial de ejecuciones



# Job SQL Server



Creamos una  
tarea de  
ejecución

Job Schedule Properties - EjecucionXMinuto

Name:  Jobs in

Schedule type:  ☒ Enabled

One-time occurrence

Date:  Time:

Frequency

Occurs:

Recurs every:  day(s)

Daily frequency

☐ Occurs once at:

☒ Occurs every:  minute(s)

Starting at:

Ending at:

Duration

Start date:

☐ End date:

☒ No end date:

Log file summary: No filter applied

Date	Step ID	Server	Job Name
<input checked="" type="checkbox"/> 02/08/2020 0:51:00		DESKTOP-SRA5N58	<a href="#">CargaVendedores</a>
<input checked="" type="checkbox"/> 02/08/2020 0:50:00		DESKTOP-SRA5N58	<a href="#">CargaVendedores</a>
<input checked="" type="checkbox"/> 02/08/2020 0:49:00		DESKTOP-SRA5N58	<a href="#">CargaVendedores</a>
<input checked="" type="checkbox"/> 02/08/2020 0:41:40		DESKTOP-SRA5N58	<a href="#">CargaVendedores</a>
<input checked="" type="checkbox"/> 02/08/2020 0:38:51		DESKTOP-SRA5N58	<a href="#">CargaVendedores</a>

*Vemos que el historial tiene  
nuevas ejecuciones*

# Referencias

Create a Transact-SQL Job Step

<https://docs.microsoft.com/en-us/sql/ssms/agent/create-a-transact-sql-job-step?view=sql-server-ver15>



**Universidad  
Nacional de  
Cajamarca**  
*"Norte de la Universidad Peruana"*



**Universidad  
Nacional de  
Cajamarca**  
*"Norte de la Universidad Peruana"*

# INDICES

# Tipos de Indices

Clustered  
Indexes

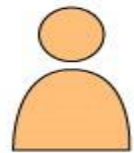
Nonclustered  
Indexes

Columnstore  
Indexes

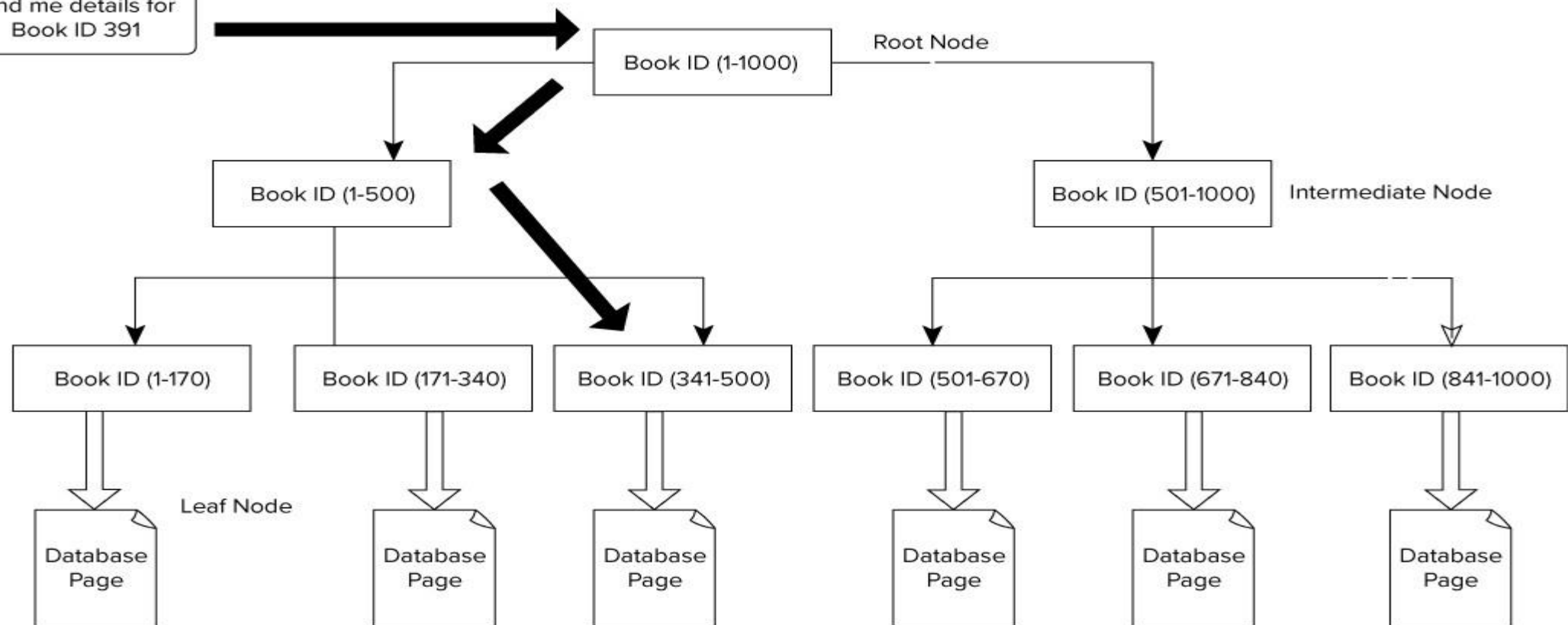
Rowstore  
Indexes



# ¿Por qué construir índices?



Find me details for  
Book ID 391



# Diseño de índices



Universidad  
Nacional de  
Cajamarca  
*"Voz de la Universidad Peruana"*



Los valores de clave de clúster deben ser secuenciales (una clave de número entero creciente es un buen ejemplo aquí)



Los índices agrupados deben ser tan estrechos (columnas mínimas) como sea posible



En claves de índice no agrupadas, la columna más selectiva debe ser la columna más a la izquierda en la clave



Use columnas incluidas en sus índices no agrupados cuando sea posible, para mantener las claves más estrechas



No debe crear índices redundantes y evitar crear índices no utilizados en sus tablas

# Clustered indexes



Universidad  
Nacional de  
Cajamarca

"Voz de la Universidad Peruana"



Un índice agrupado ordena y almacena los datos en una tabla en función de los valores clave



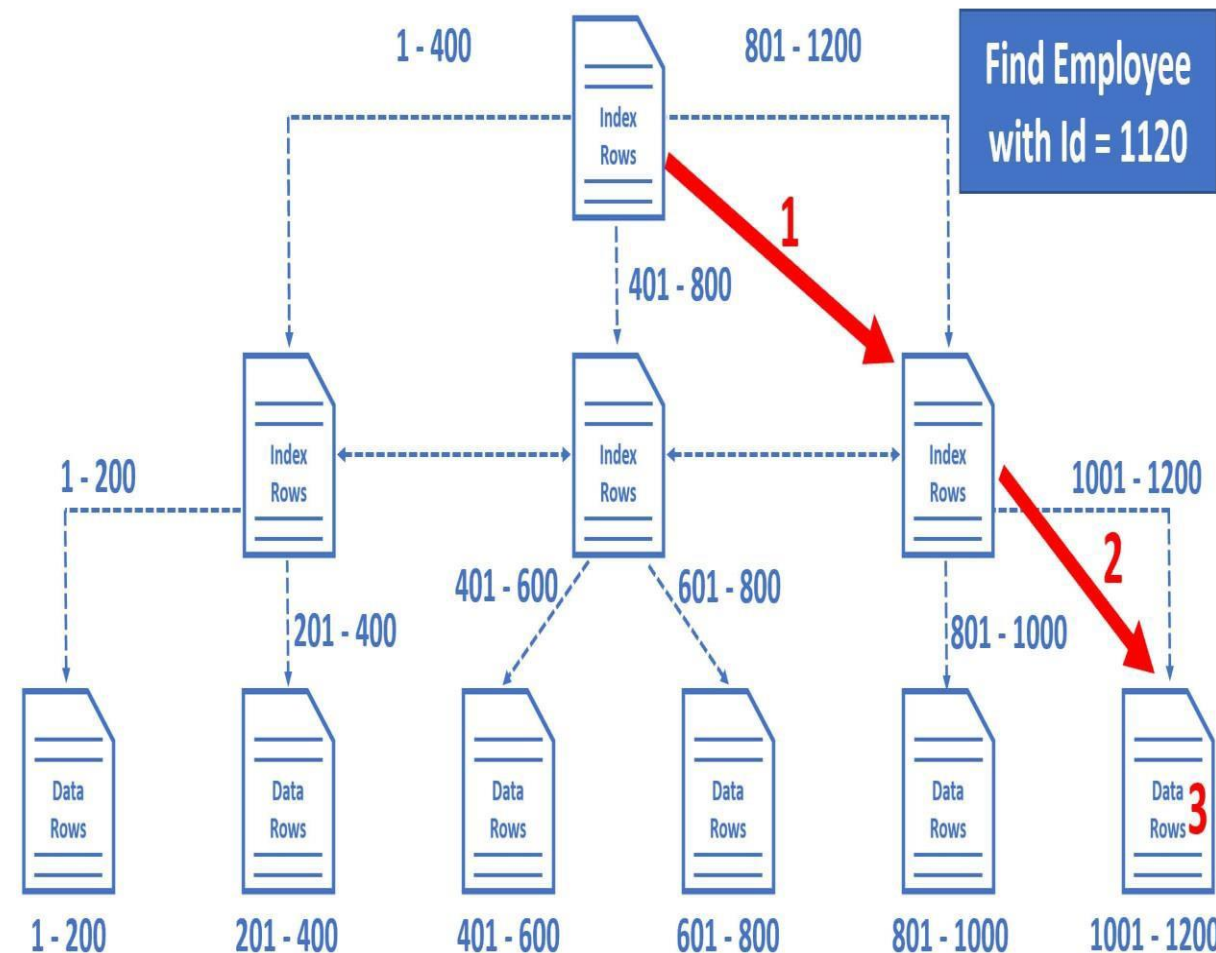
Solo puede haber un índice agrupado por tabla, ya que las filas solo se pueden almacenar en un orden



Los índices agrupados suelen ser la clave principal de una tabla



Los índices agrupados no tienen requisitos de unicidad



# ¿Qué son los índices no agrupados?



**Nonclustered indexes** son índices secundarios que se utilizan para ayudar al rendimiento de las consultas no atendidas por el índice agrupado



Puede crear múltiples índices no agrupados en una tabla

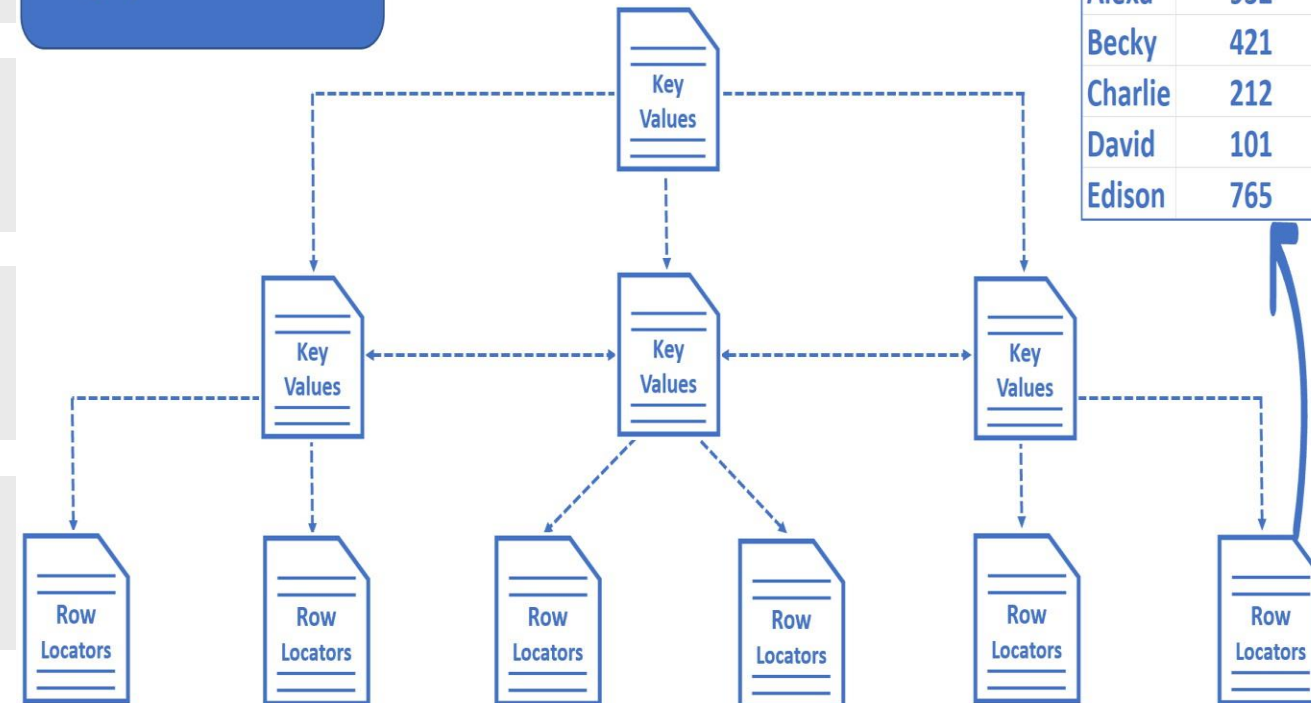


También puede crear índices filtrados, para tablas con una gran asimetría de datos



El costo de los índices no agrupados es el espacio y el rendimiento de inserción/actualización

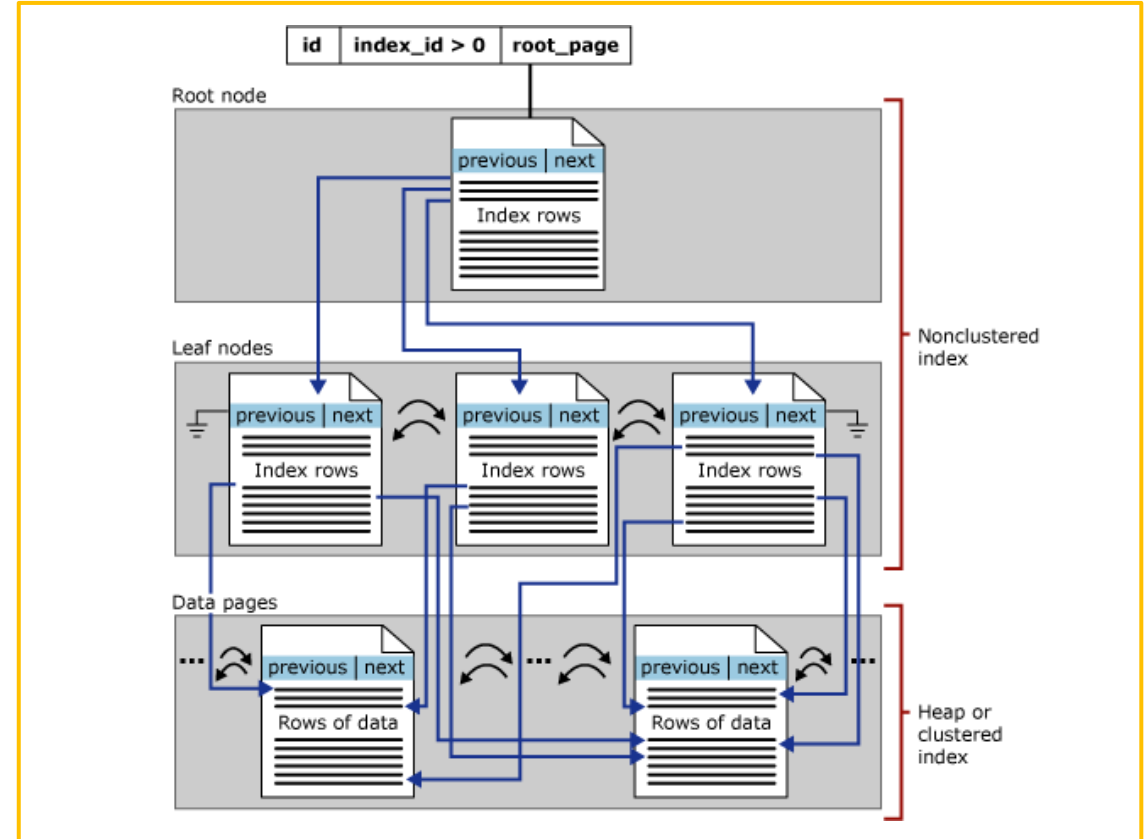
Non-Clustered Index on  
Employee Name Column



# Nonclustered indexes



- Cada página en un árbol b de índice se denomina nodo de índice, y el nodo superior del árbol b se denomina nodo raíz.
- Los nodos inferiores de un índice se denominan nodos de hoja y la colección de nodos de hoja es el nivel de hoja.



# Índices de almacén de columnas o column store



Los datos se almacenan en columnas, no en filas



Esto permite niveles más altos de compresión, lo que reduce el espacio de almacenamiento y memoria de los datos.



Las columnas a las que no se hace referencia en una consulta no se escanean, lo que reduce la cantidad de E/S necesaria



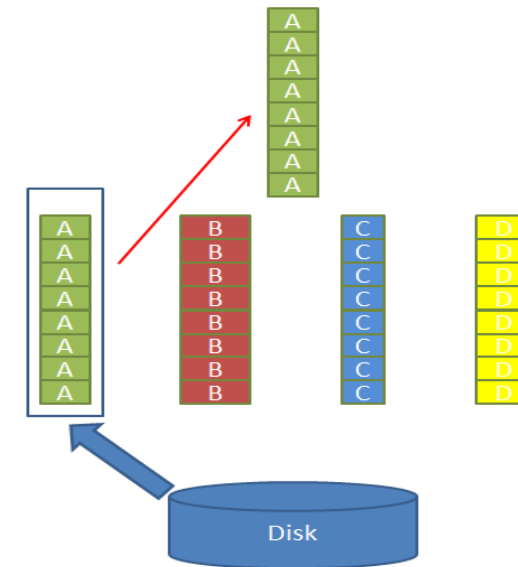
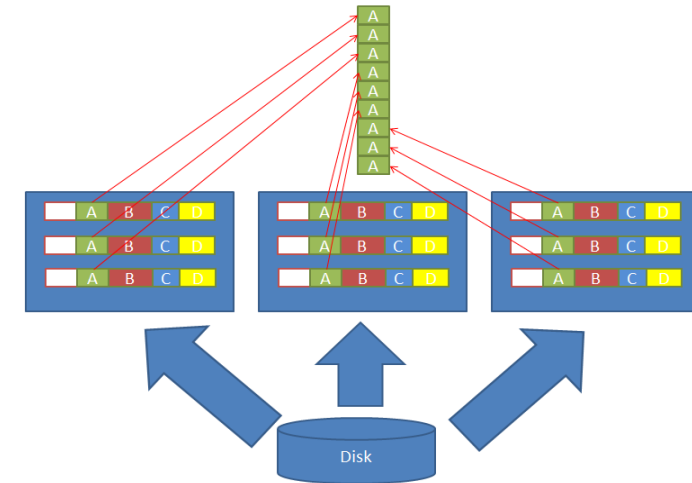
Se utiliza mejor en tablas grandes (por ejemplo, tablas de hechos de almacenamiento de datos, tablas de historial temporal)



El almacén de columnas agrupado debe incluir todas las columnas de la tabla



Use el índice de almacén de columnas agrupado para las tablas de hechos y las tablas de dimensiones grandes para las cargas de trabajo de DW. Use el índice de almacén de columnas no agrupado para realizar análisis en tiempo real en cargas de trabajo de OLTP





## Columnstore indexes – data load performance comparison

```
SQLQuery3.sql - VM...VM1\jdantoni (52))* - X
SET STATISTICS TIME ON

INSERT INTO FactResellerSales_CCI_Demo
SELECT TOP 1024000 *
FROM FactResellerSalesXL_CCI

INSERT INTO FactResellerSales_Page_Demo
SELECT TOP 1024000 *
FROM FactResellerSalesXL_CCI

100 %
Messages
SQL Server parse and compile time:
  CPU time = 0 ms, elapsed time = 6 ms.

SQL Server Execution Times:
  CPU time = 7609 ms, elapsed time = 7902 ms.
(1024000 rows affected)

SQL Server Execution Times:
  CPU time = 17031 ms, elapsed time = 19685 ms.
(1024000 rows affected)

Completion time: 2020-04-22T14:02:07.5526154+00:00
```

Clustered columnstore index

Clustered index with page compression

8 secs

20 secs

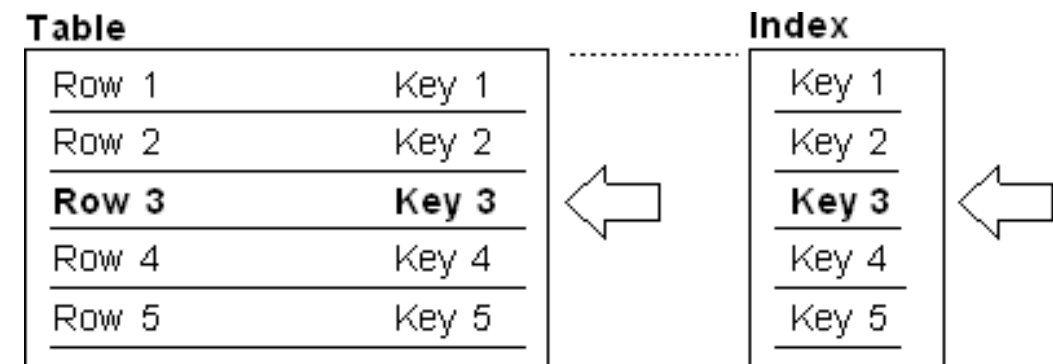


# Tipos de Índices

- Los índices son objetos de base de datos que puede crear para mejorar el rendimiento de algunas consultas.
- El servidor también puede crear índices automáticamente cuando crea una clave primaria o una restricción única

Considere:

- Los índices mal diseñados y la falta de índices son las principales demoras de la base de datos.
- Un índice bien diseñado consigue un buen rendimiento de una base de dato.

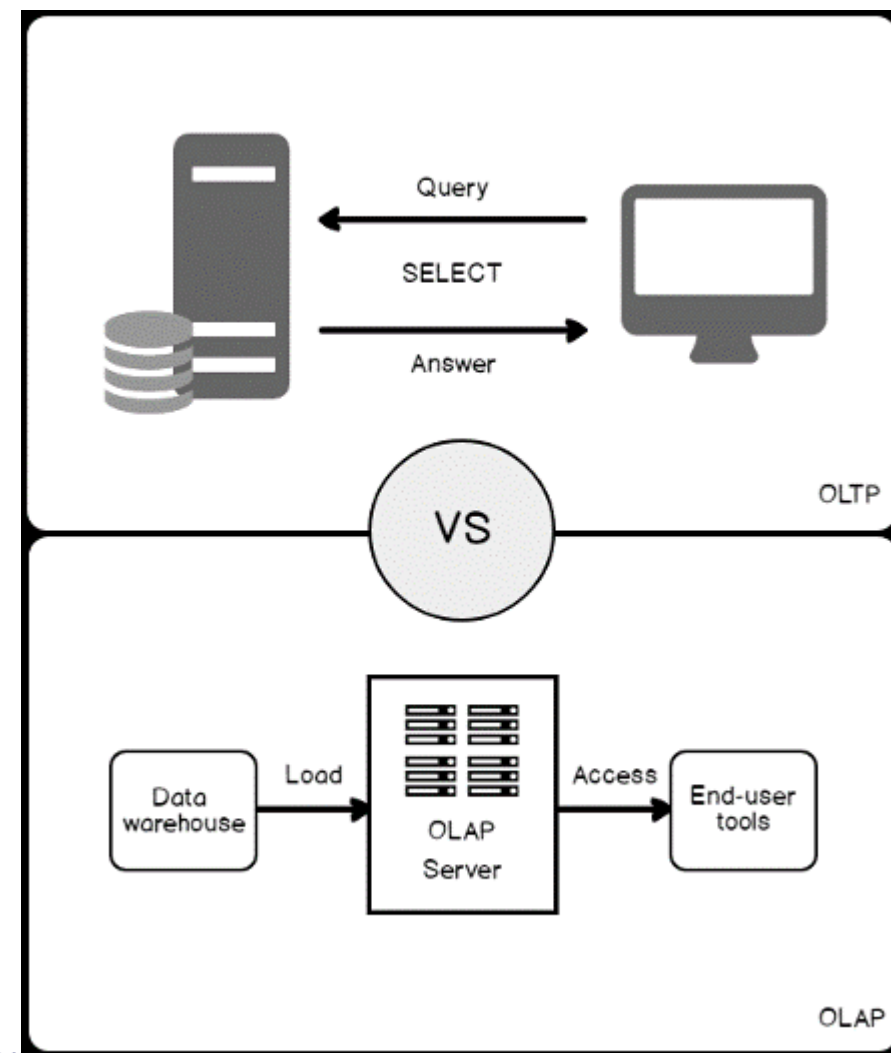


# Tipos de Índices

## Antes de crear un índice:

Identifique el tipo de carga de trabajo de la base de datos.

- En la base de datos del Procesamiento Transaccional Online (OLTP), las cargas de trabajo son usadas para sistemas transaccionales en los cuales, la mayoría de las consultas presentadas son consultas de modificación de información.
- En cambio, las cargas de trabajo de la base de datos del Procesamiento Analítico Online (OLAP) son usadas para sistemas de almacenamiento de información, en los cuales la mayoría de las consultas presentadas son consultas de recuperación de información que filtra, agrupa, agrega y une grupos grandes de información rápidamente. La diferencia entre las bases de dato OLTP y el OLAP



# Tipos de Índices



Tipo de índice	Descripción
Hash	Con un índice hash, se accede a los datos a través de una tabla hash en memoria. Los índices hash utilizan una cantidad fija de memoria, que es una función del número de cubos.
Índice no agrupado optimizado para memoria	Para los índices no clúster optimizados para memoria, el consumo de memoria depende del número de filas y del tamaño de las columnas de clave de índice.
Clúster	Un índice clúster ordena y almacena las filas de datos de la tabla o vista por orden en función de la clave del índice clúster. El índice clúster se implementa como una estructura de árbol b que admite la recuperación rápida de las filas a partir de los valores de las claves del índice clúster.
No agrupado	Los índices no clúster se pueden definir en una tabla o vista con un índice clúster o en un montón. Cada fila del índice no clúster contiene un valor de clave no agrupada y un localizador de fila. Este localizador apunta a la fila de datos del índice clúster o el montón que contiene el valor de clave. Las filas del índice se almacenan en el mismo orden que los valores de la clave del índice, pero no se garantiza que las filas de datos estén en un determinado orden a menos que se cree un índice clúster en la tabla.

# Tipos de Índices



**Universidad  
Nacional de  
Cajamarca**  
"Norte de la Universidad Peruana"

Tipo de índice	Descripción
Único	<p>Un índice único se asegura de que la clave de índice no contenga valores duplicados y, por tanto, cada fila de la tabla o vista sea en cierta forma única.</p> <p>La unicidad puede ser una propiedad tanto de índices clúster como de índices no clúster.</p>
columnstore	<p>El índice de almacén de columnas en memoria almacena y administra los datos mediante el almacenamiento de datos basado en columnas y el procesamiento de consultas basado en columnas.</p> <p>Los índices de almacén de columnas funcionan correctamente para las cargas de trabajo de almacenamiento de datos que ejecutan principalmente cargas masivas y consultas de solo lectura. Use el índice de almacén de columnas para aumentar hasta en diez veces el rendimiento de las consultas en relación con el almacenamiento tradicional orientado a filas, y hasta en siete veces la compresión de datos en relación con el tamaño de los datos sin comprimir.</p>
Índice con columnas incluidas	<p>Índice no clúster que se extiende para incluir columnas sin clave además de las columnas de clave.</p>



# Tipos de Índices

Tipo de índice	Descripción
Índice en columnas calculadas	Índice de una columna que se deriva del valor de una o varias columnas, o algunas entradas deterministas.
Filtered	Índice no clúster optimizado, especialmente indicado para cubrir consultas que seleccionan de un subconjunto bien definido de datos. Utiliza un predicado de filtro para indizar una parte de las filas de la tabla. Un índice filtrado bien diseñado puede mejorar el rendimiento de las consultas y reducir los costos de almacenamiento del índice en relación con los índices de tabla completa, así como los costos de mantenimiento.
Índice con columnas incluidas	Índice no clúster que se extiende para incluir columnas sin clave además de las columnas de clave.

# Tipos de Índices

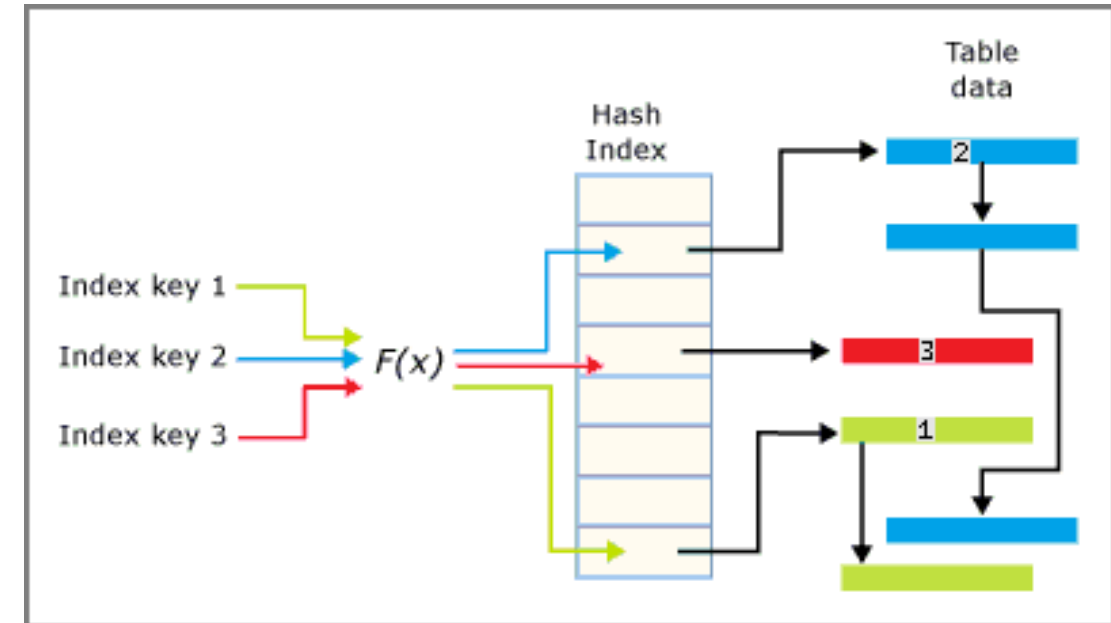
## Índices HASH

### Consideraciones de rendimiento

- Evite agregar columnas que no sean necesarias.
- Disminuye la cantidad de filas que caben en una página.
- Se necesita más espacio en disco para almacenar el índice.
- Evite usar tipos de datos varchar(max) , nvarchar(max) , varbinary(max) o xml.
- Considere que el índice puede aumentar el tiempo para realizar operaciones de Insert, Update, Delete.

*Evalué entonces:*

Si la mejora del rendimiento esta compensada en la actualización de los datos de la tabla y en los requisitos de espacio en disco adicionales.



# Tipos de Índices

## Índice No Agrupado

### Consideraciones de rendimiento

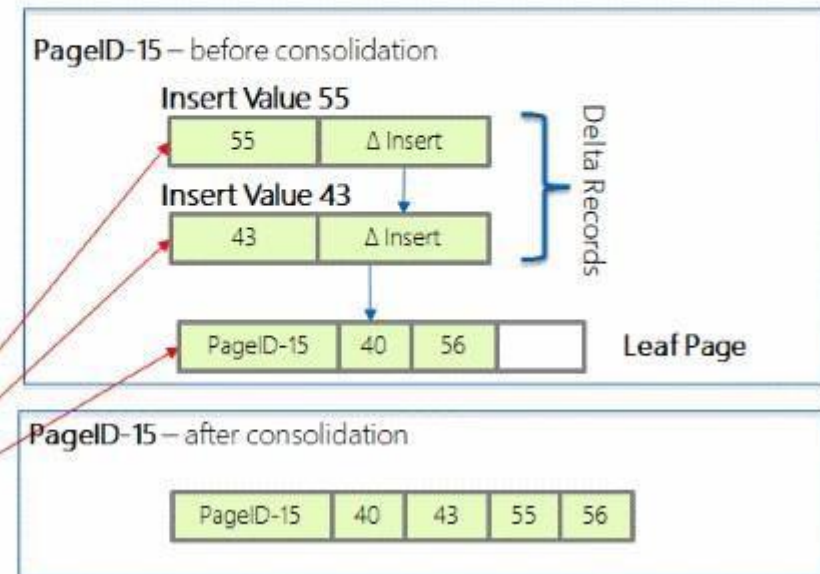
Al consultar una tabla optimizada para memoria con predicados de desigualdad, el rendimiento de los índices no agrupados es superior al de los índices de hash no agrupados.

### Recomendaciones

- Las consultas tienen una cláusula ORDER BY en la columna indexada.
- Las consultas en las que solo se comprueban las primeras columnas de un índice con varias columnas.
- Las consultas prueban la columna indexada mediante el uso de una cláusula WHERE con:
  - Una desigualdad: WHERE StatusCode != 'Done'
  - Un examen de intervalo de valores: WHERE Quantity >= 100

Page Mapping Table

PAGE 0
PAGE 1
PAGE 2
PAGE 3
PAGE 4
PAGE 5
PAGE 6
PAGE 7
PAGE 8
PAGE 9
PAGE 10
PAGE 11
PAGE 12
PAGE 13
PAGE 14
PAGE 15





# Tipos de Índices

## *Índice Cluster*

## Implementación

### **Restricciones PRIMARY KEY y UNIQUE**

Cuando se crea una restricción PRIMARY KEY, se crea automáticamente un índice clúster único en las columnas si aún no existe un índice clúster en la tabla o no se ha especificado un índice no clúster. La columna de clave principal no puede permitir valores NULL.

Cuando cree una restricción UNIQUE, se creará un índice no clúster único para exigir una restricción UNIQUE de forma predeterminada. Puede especificarse un índice clúster único si todavía no existe un índice clúster en la tabla. Un índice creado como parte de la restricción recibe automáticamente el mismo nombre que la restricción. Para obtener más información

### **Índice independiente de una restricción**

Puede crear un índice clúster en una columna que no sea la de clave principal si se especificó una restricción de clave principal no agrupada.

# Tipos de Índices

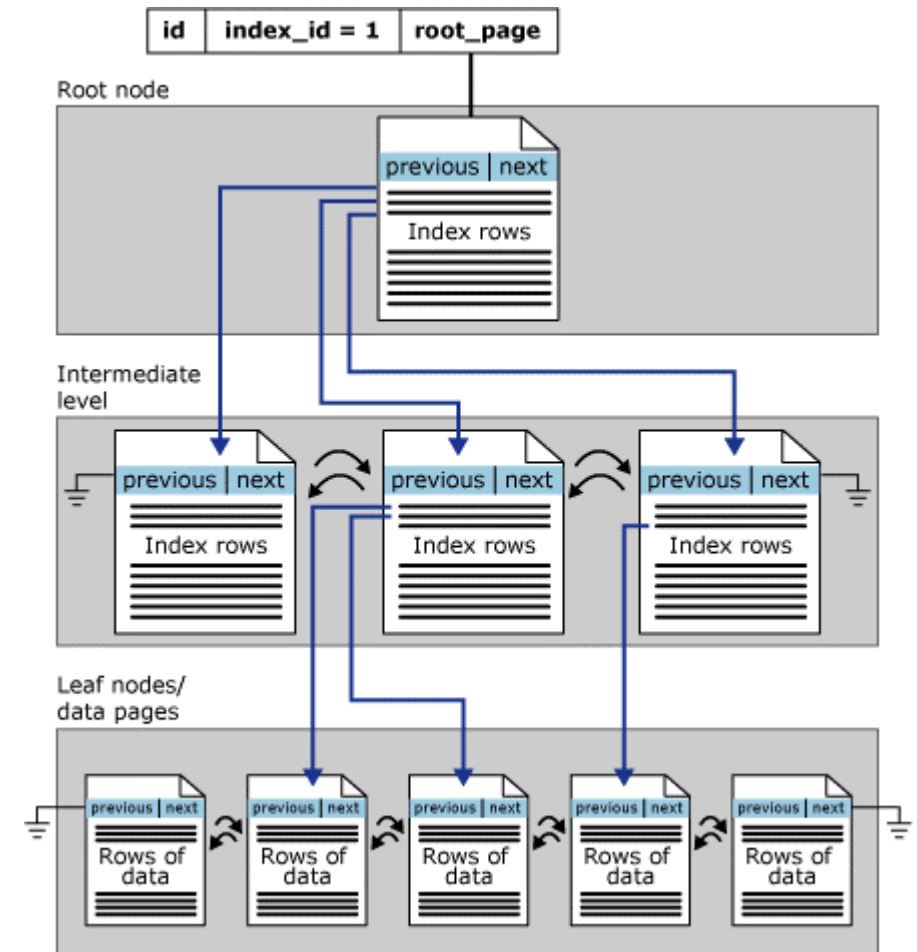


Universidad  
Nacional de  
Cajamarca  
"Norte de la Universidad Peruana"

## Índice Cluster

### Consideraciones sobre consultas

- Devuelven un intervalo de valores mediante el uso de operadores como BETWEEN, >, >=, < y <=.
- Devuelven grandes conjuntos de resultados.
- Usan cláusulas JOIN; por lo general, son columnas de clave externa.
- Usan cláusulas ORDER BY o GROUP BY.
- Por regla general, debe definir la clave de índice clúster con el menor número de columnas posible.

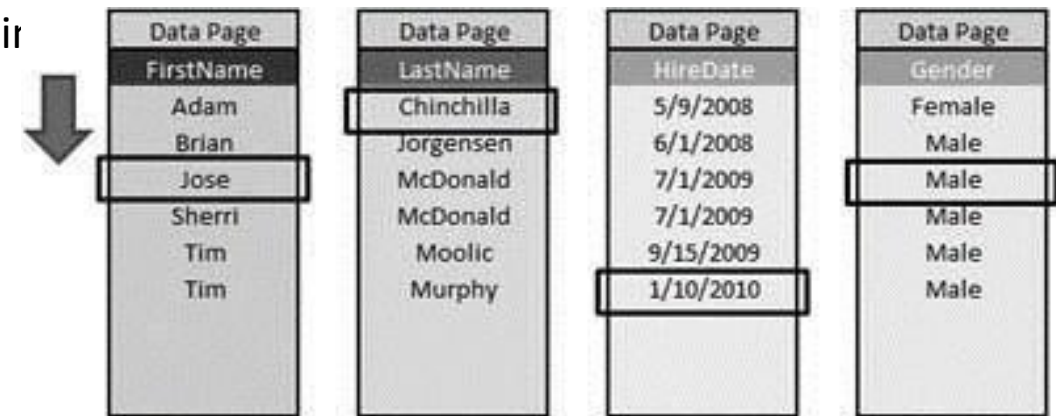


# Tipos de Índices

## ***Índices de almacén de columnas***

Recomendaciones:

- Use el índice para almacenar tablas de hechos y tablas de dimensiones grandes.
- Este método mejora el rendimiento de las consultas. Para más ir columnas para el almacenamiento de datos.
- Use un índice de almacén de columnas no agrupado para
- realizar análisis en tiempo real en una carga de trabajo OLTP



Los datos de la tabla se almacenan en *páginas separadas* para cada columnas

# Tipos de Índices

## Revisando el uso de los índices

```
SELECT OBJECT_NAME(IX.OBJECT_ID) Table_Name
```

```
,IX.name AS Index_Name
```

```
,IX.type_desc Index_Type
```

```
,SUM(PS.[used_page_count]) * 8 IndexSizeKB
```

```
,IXUS.user_seeks AS NumOfSeeks
```

```
,IXUS.user_scans AS NumOfScans
```

```
,IXUS.user_lookups AS NumOfLookups
```

```
,IXUS.user_updates AS NumOfUpdates
```

```
,IXUS.last_user_seek AS LastSeek
```

```
,IXUS.last_user_scan AS LastScan
```

```
,IXUS.last_user_lookup AS LastLookup
```

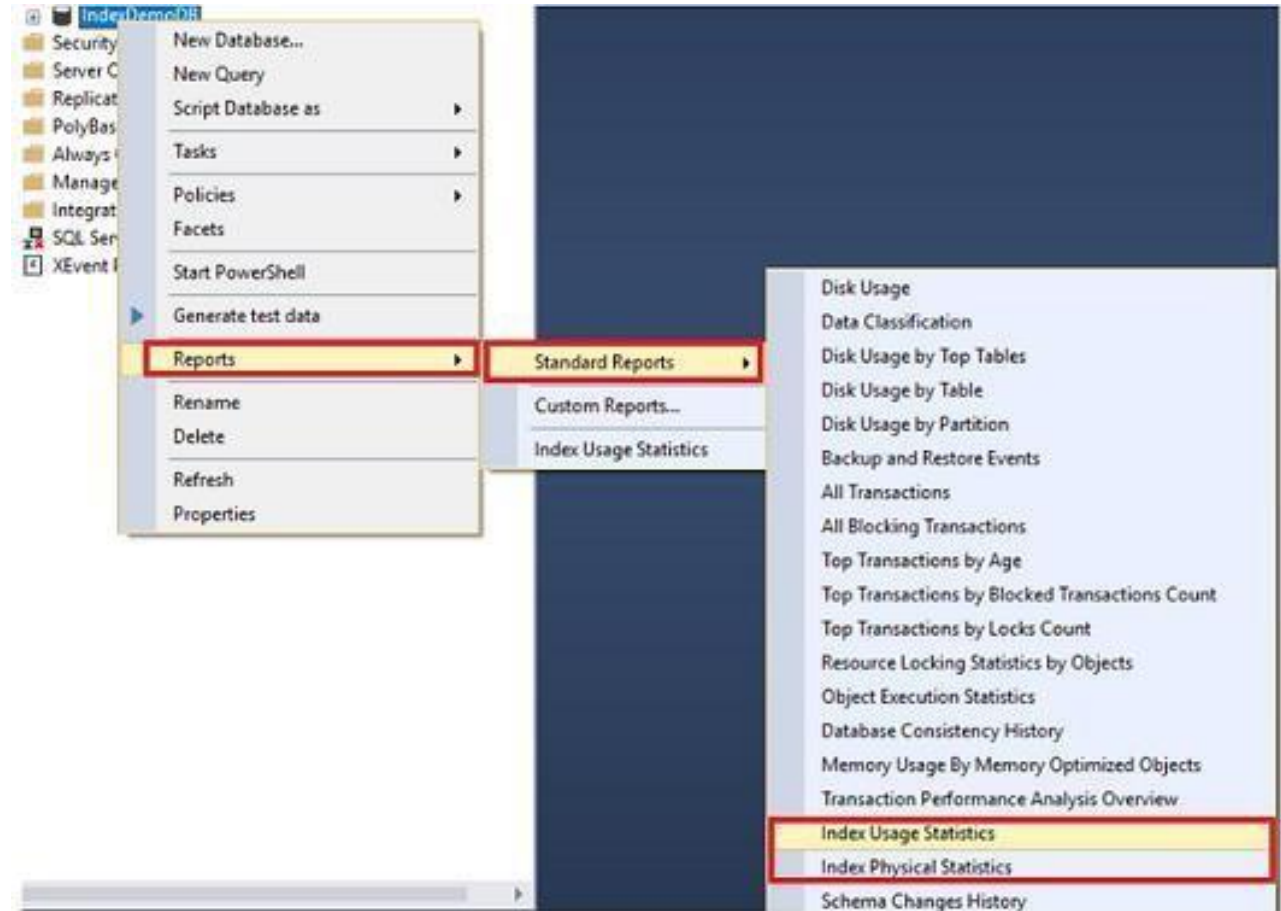
```
,IXUS.last_user_update AS LastUpdate FROM sys.indexes IX
```

- `INNER JOIN sys.dm_db_index_usage_stats IXUS ON IXUS.index_id = IX.index_id AND IXUS.OBJECT_ID = IX.OBJECT_ID`
- `INNER JOIN sys.dm_db_partition_stats PS on PS.object_id=IX.object_id`
- `WHERE OBJECTPROPERTY(IX.OBJECT_ID, 'IsUserTable') = 1`
- `GROUP BY OBJECT_NAME(IX.OBJECT_ID) ,IX.name ,IX.type_desc ,IXUS.user_seeks ,IXUS.user_scans`
- `,IXUS.user_lookups,IXUS.user_updates ,IXUS.last_user_seek ,IXUS.last_user_scan ,IXUS.last_user_lookup`
- `,IXUS.last_user_update`

El número de **Búsquedas** es el número de veces que el índice es usado, el número de **Escaneos** muestra el número de veces que las páginas hoja en el índice son escaneadas, el número de **Consultas** indica el número de veces que un índice Agrupado es usado por el índice No Agrupado para buscar la fila entera y el número de **Actualizaciones** muestra el número de veces que la información del índice es modificada.

# Tipos de Índices

El SQL Server nos provee de dos reportes incorporados que nos ayudan a monitorear la base de datos de fragmentación de índices y el uso de estadísticas, el Index Usage Statistics y el Index Physical Statistics. Estos reportes standard usan los DMOs previamente descritos, y la información de los reportes será actualizada cuando el servicio de SQL Server es reiniciado. Ambos reportes pueden ser vistos haciendo clic derecho en la base de datos, de la cual necesitas monitorear sus índices, escoge Reports -> Standard Reports y selecciona el reporte Index Usage Statistics o Index Physical Statistics, como se muestra abajo:



# Revisando el uso de los índices

- Todos los valores cero significa que la tabla no es usada, o el servicio SQL Server reinició recientemente.
- Un índice con cero o pequeño número de búsquedas, escaneos o consultas y un gran número de actualización es un índice inútil y debe ser removido, después de verificar con el propietario del sistema, ya que el principal objetivo de añadir el índice es hacer más rápidas las operaciones de lectura.
- Un índice que es escaneado excesivamente con cero o pequeño número de búsquedas significa que el índice es mal usado y debería ser remplazado por uno más óptimo.
- Un índice con un gran número de consultas significa que necesitamos optimizar el índice al añadir las columnas frecuentemente consultadas a las columnas de índice no-clave existentes usando la cláusula INCLUDE.
- Una tabla con un gran número de Escaneos indica que las consultas SELECT \* son excesivamente usadas, recuperando más columnas de lo que es requerido, o las estadísticas de índice deben ser actualizadas.
- Un índice agrupado con un gran número de Escaneos significa que un nuevo índice No agrupado debería ser creado para cubrir las consultas no cubiertas.
- Las fechas con valores NULL significan que esta acción no ha ocurrido todavía.
- Grandes escaneos están BIEN en pequeñas tablas
- Si tu índice no está aquí, entonces ninguna acción es realizada en ese índice todavía.



Fin de la sesión

