# Capítulo V

## Introducción a las Consultas

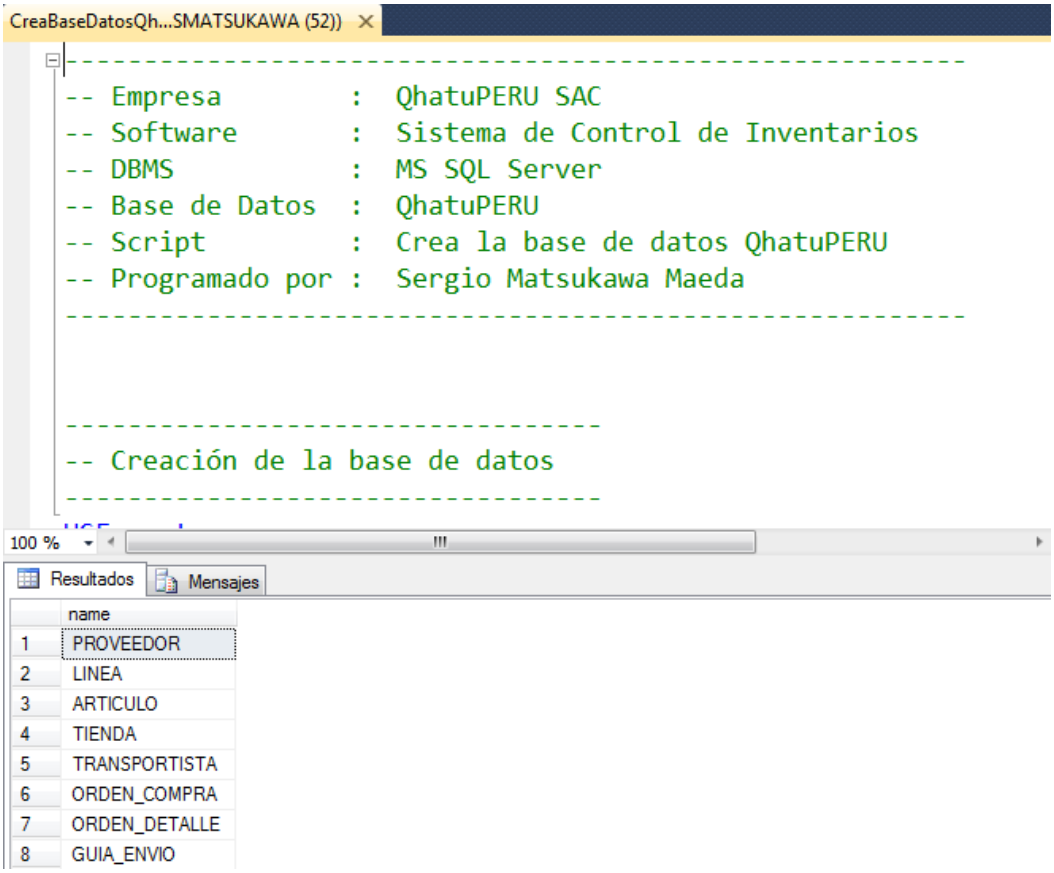
En este capítulo conoceremos cómo leer los datos de una tabla y organizarlos en el informe de salida usando la instrucción SELECT.

# 1. PREPARACIÓN DEL ESCENARIO

Para desarrollar los temas de este capítulo utilizaremos la base de datos **QhatuPERU**, cuyo modelo de datos se presentó en el capítulo 3. En la carpeta **ScriptsBD** del CD incluido en el libro encontrará el script **CreaBaseDatosQhatuPERU.sql**; al ejecutarlo se creará la base de datos, sus tablas y relaciones, y además insertará una cantidad adecuada de registros de datos, necesarios para ilustrar los temas a presentar.

**Ejercicio 5.1: Ejecución del script que crea la base de datos QhatuPERU**

1. Inicie una sesión SQL Server Management Studio en su servidor SQL.
2. En el menú **Archivo**, seleccione **Abrir**, y luego **Archivo**.
3. Abra el archivo **CreaBaseDatosQhatuPERU.sql**.
4. Para ejecutar el script, haga clic en **Ejecutar** de la barra de herramientas ó pulse la tecla [F5].

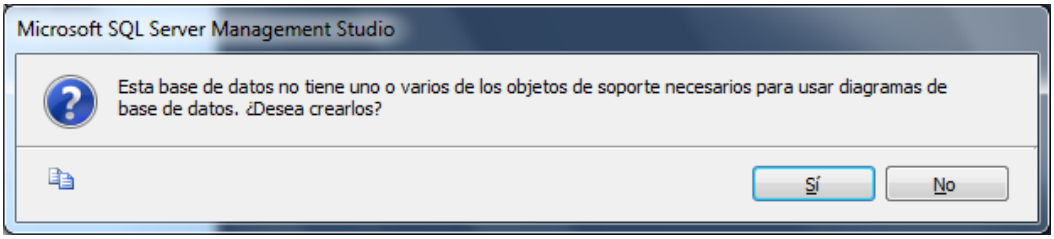


**Ejercicio 5.2: Creación de un diagrama de la base de datos**

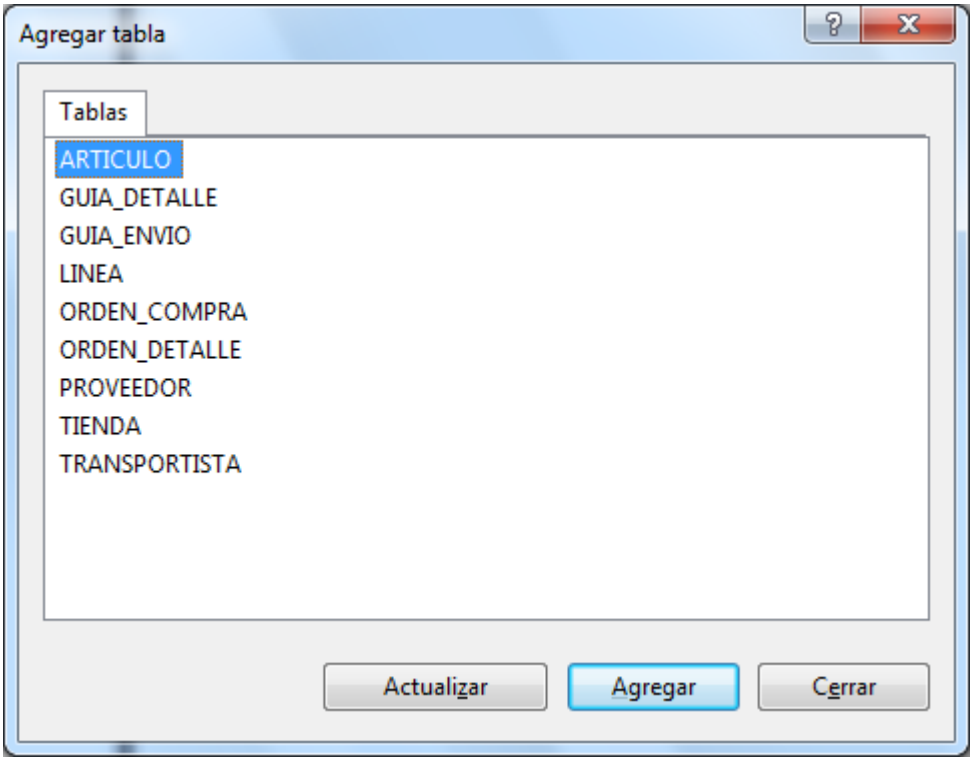
1. Si bajo el nodo **Bases de datos** de su servidor no ve la base de datos **QhatuPERU**, haga clic secundario sobre **Bases de datos** y ejecute **Actualizar**.

2. Expanda el nodo de la base de datos, haga clic secundario sobre **Diagramas de bases de datos**, y luego clic sobre **Nuevo diagrama de base de datos**.

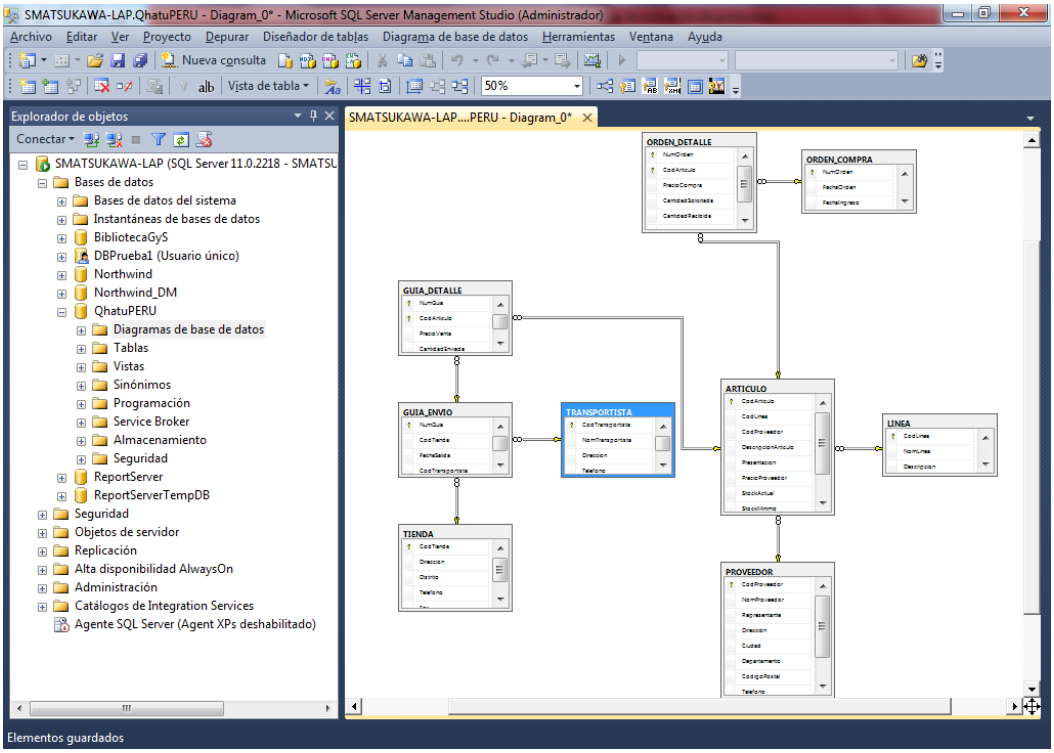
3. Si se muestra el siguiente mensaje, haga clic en **Sí**.




4. Se muestra el diálogo **Agregar tabla**.



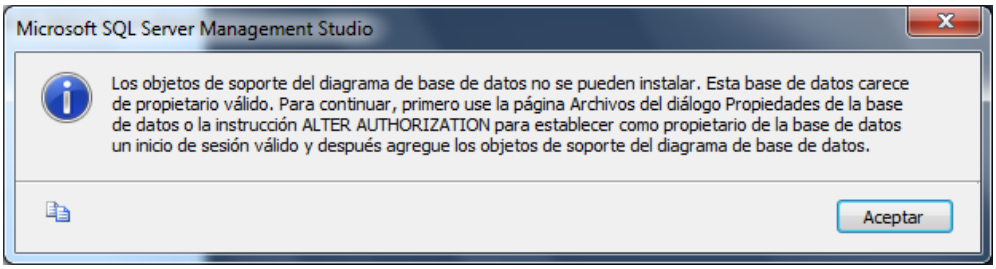
5. Seleccione las tablas que desea mostrar en el diagrama y haga clic en **Agregar**, y para finalizar haga clic en **Cerrar**.




6. Guarde el diagrama bajo el nombre ***ModeloCompleto***.

---

**Nota:** Si al intentar crear un diagrama para una base de datos recibe el siguiente mensaje:

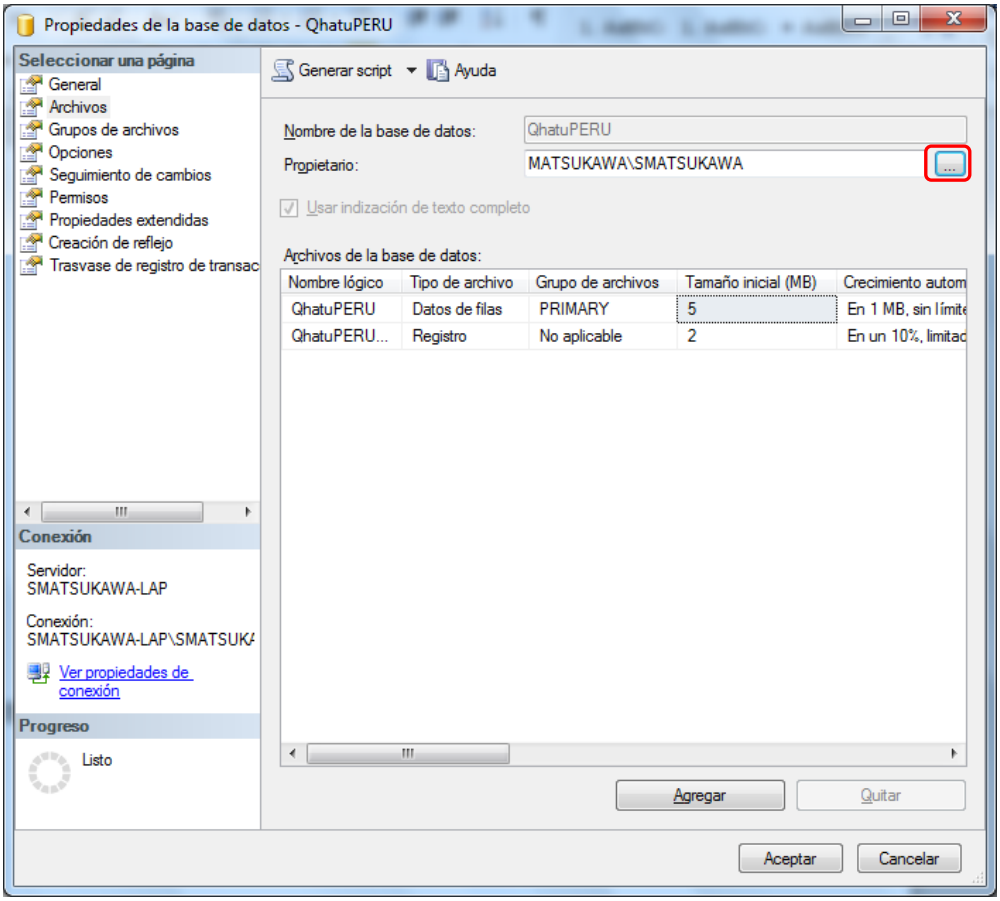



Proceda del siguiente modo:

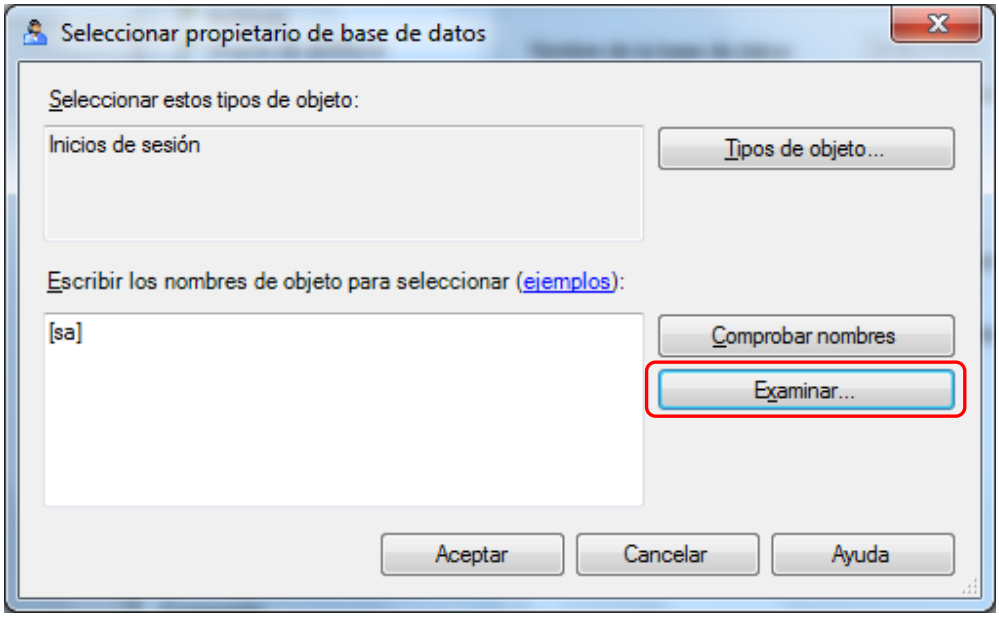
1. Haga clic secundario sobre la base de datos, y en el menú contextual seleccione **Propiedades**.

2. En la ventana **Propiedades de la base de datos** seleccione la página **Archivos**.

3. En el campo **Propietario** haga clic sobre el botón **Buscar**.




4. En la ventana **Seleccionar propietario de base de datos** haga clic en **Examinar**, busque la cuenta **sa**, selecciónela y haga clic en **Aceptar**. **sa** es la cuenta **System Administrator**.




5. Haga clic en **Aceptar** para regresar a la página **Archivos**, y luego en **Aceptar** para cerrar la ventana **Propiedades de la base de datos**.

# 2. Consultas a las Tablas – Instrucción SELECT

**Sintáxis básica**

```
SELECT * | lista_columnas
FROM nombre_tabla
[ WHERE condición_filas ]
```

- **lista_columnas** es la lista de columnas a mostrar en el resultado de la consulta. Si se especifica ***** se mostrarán todas las columnas de la tabla.
- **condición_filas** es una expresión lógica que indica que las filas a mostrar son aquellas para las que el valor de la expresión es verdadero.

**Ejercicio 5.3: Lectura de todos los datos de una tabla**

```
USE QhatuPERU
go

SELECT * FROM ARTICULO
```

```
go
```

Muestra todas las columnas y todas las filas de la tabla ARTICULO de la base de datos **QhatuPERU**.

Resultados | Mensajes

| | CodArticulo | CodLinea | CodProveedor | DescripcionArticulo | Presentacion | PrecioProveedor |
|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 14 | CARAMELOS BASTON VIENA ARCOR | PAQUETE 454 GR | 1,50 |
| 2 | 2 | 1 | 15 | CARAMELOS SURTIDO DE FRUTAS | PAQUETE 450 GR | 1,00 |
| 3 | 3 | 1 | 14 | CARAMELOS FRUTAS SURTIDA ARCOR | PAQUETE 520 GR | 1,50 |
| 4 | 4 | 1 | 14 | CARAMELOS FRUTAS MASTICABLES | PAQUETE 454 GR | 1,30 |
| 5 | 5 | 1 | 15 | CHUPETES LOLY AMBROSOLI | KILOGRAMO | 1,20 |
| 6 | 6 | 1 | 15 | FRUNA SURTIDA DONOFRIO | PAQUETE X 24 UNIDADES | 1,80 |
| 7 | 7 | 1 | 15 | CHOCOLATE DOÑA PEPA FIELD | PAQUETE X 6 UNIDADES | 2,20 |
| 8 | 8 | 1 | 15 | CHOCOLATE CUA CUA FIELD | PAQUETE X 6 UNIDADES | 1,60 |
| 9 | 9 | 1 | 15 | MELLOWS FAMILIAR FIELD | PAQUETE 454 GR | 2,10 |
| 10 | 10 | 1 | 15 | WAFER CHOCOLATE FIELD | PAQUETE X 9 UNIDADES | 0,70 |

**Ejercicio 5.4: Lectura de columnas seleccionadas de una tabla**

```
SELECT CodProveedor, NomProveedor,
     Direccion, Ciudad
FROM PROVEEDOR
go
```

El resultado muestra las columnas **CodProveedor**, **NomProveedor**, **Direccion** y **Ciudad** de la tabla PROVEEDOR.

Resultados | Mensajes

| | CodProveedor | NomProveedor | Direccion | Ciudad |
|---|---|---|---|---|
| 1 | 1 | LACTEOS DEL CENTRO | LIBERTAD 345 URB. EL PINO | HUANCAYO |
| 2 | 2 | DISTRIBUIDORA ALEMANA | AV. SAN VICENTE 1276 ATE-VITARTE | LIMA |
| 3 | 3 | EMBUTIDOS EL GORDITO | AV. VENEZUELA 5434 BELLAVISTA | CALLAO |
| 4 | 4 | DISTRIBUIDORA NANDO | LOS CONDORES 345 PARQUE INDUSTRIAL EL ALAMO | CALLAO |
| 5 | 5 | DISTRIBUIDORA ALBRICIAS | LEONCIO PRADO 625 MAGDALENA | LIMA |
| 6 | 6 | DISTRIBUIDORA DEL HOGAR | SAN MARTIN 1187 SMP | LIMA |
| 7 | 7 | PAPELERA PACHACAMAC | LOTIZACION INDUSTRIAL MZA. H LOTE 34 | CAÑETE |
| 8 | 8 | DISTRIBUIDORA SAN ANTONIO | AV. DEL PACIFICO 5634 SAN JUAN | LIMA |
| 9 | 9 | EMBOTELLADORA LA PREFERIDA | HEROES DEL CENEPA 342 CERCADO | TRUJILLO |
| 10 | 10 | DROGUERIA MAHAN | AV. CIRCUNVALACION 625 ZONA INDUSTRIAL | AREQUIPA |
| 11 | 11 | QUIMICA DEL NORTE | JOSE CARLOS MARIATEGUI 473 CERCADO | CHICLAYO |

**Ejercicio 5.5: Uso de alias para los títulos de columnas**

```
SELECT    CodArticulo AS Código,
```

```
            DescripcionArticulo AS Descripción,
            Presentacion AS Unidad,
            PrecioProveedor AS 'Precio unitario'
FROM ARTICULO
go
```

Permite mostrar títulos alternativos para las columnas. Si el alias contiene espacios se debe escribir entre comillas.

Resultados | Mensajes

| | Código | Descripción | Unidad | Precio unitario |
|---|---|---|---|---|
| 1 | 1 | CARAMELOS BASTON VIENA ARCOR | PAQUETE 454 GR | 1,50 |
| 2 | 2 | CARAMELOS SURTIDO DE FRUTAS | PAQUETE 450 GR | 1,00 |
| 3 | 3 | CARAMELOS FRUTAS SURTIDA ARCOR | PAQUETE 520 GR | 1,50 |
| 4 | 4 | CARAMELOS FRUTAS MASTICABLES | PAQUETE 454 GR | 1,30 |
| 5 | 5 | CHUPETES LOLY AMBROSOLI | KILOGRAMO | 1,20 |
| 6 | 6 | FRUNA SURTIDA DONOFRIO | PAQUETE X 24 UNIDADES | 1,80 |
| 7 | 7 | CHOCOLATE DOÑA PEPA FIELD | PAQUETE X 6 UNIDADES | 2,20 |
| 8 | 8 | CHOCOLATE CUA CUA FIELD | PAQUETE X 6 UNIDADES | 1,60 |
| 9 | 9 | MELLOWS FAMILIAR FIELD | PAQUETE 454 GR | 2,10 |
| 10 | 10 | WAFER CHOCOLATE FIELD | PAQUETE X 9 UNIDADES | 0,70 |
| 11 | 11 | CHOCOLATE BARRA REGULAR | BARRA 2 ONZAS | 0,40 |

Las siguientes son formas alternas para definir los títulos de columnas usando alias.

```
SELECT      Código = CodArticulo,
            Descripción = DescripcionArticulo,
            Unidad = Presentacion,
            'Precio unitario' = PrecioProveedor
FROM ARTICULO
go


SELECT      CodArticulo Código,
            DescripcionArticulo Descripción,
            Presentacion Unidad,
            PrecioProveedor 'Precio unitario'
FROM ARTICULO
go
```

**Ejercicio 5.6: Uso de columnas calculadas**

Una columna computada es una columna que se muestra en el resultado de una consulta, pero que no existe físicamente como tal en la tabla. La columna computada muestra el resultado de ejecutar alguna operación con las columnas de la tabla.

```
SELECT CodArticulo, PrecioVenta, CantidadEnviada,
     Monto = PrecioVenta * CantidadEnviada
FROM GUIA_DETALLE
go
```

El contenido de la columna calculada **Monto** se obtiene operando con los valores de las columnas **PrecioVenta** y **CantidadEnviada**.

Resultados | Mensajes

| | CodArticulo | PrecioVenta | CantidadEnviada | Monto |
|---|---|---|---|---|
| 1 | 1 | 2,25 | 20 | 45,00 |
| 2 | 2 | 1,50 | 20 | 30,00 |
| 3 | 3 | 2,25 | 20 | 45,00 |
| 4 | 4 | 1,95 | 20 | 39,00 |
| 5 | 6 | 2,70 | 30 | 81,00 |
| 6 | 7 | 3,30 | 25 | 82,50 |
| 7 | 8 | 2,40 | 25 | 60,00 |
| 8 | 9 | 3,15 | 20 | 63,00 |
| 9 | 10 | 1,05 | 20 | 21,00 |
| 10 | 11 | 0,60 | 50 | 30,00 |
| 11 | 12 | 2,25 | 50 | 112,50 |

**Ejercicio 5.7: Uso del operador + para concatenar cadenas en el resultado**

```
SELECT CodTienda,
     Direccion + ' - ' + Distrito AS Ubicación
FROM TIENDA
go
```

En el ejemplo se concatena las cadenas almacenadas en los campos **Direccion** y **Distrito**, y se muestran como una sola cadena en la columna **Ubicación** del resultado.

Resultados | Mensajes

| | CodTienda | Ubicación |
|---|---|---|
| 1 | 1 | AV. LA PAZ 659 - MIRAFLORES |
| 2 | 2 | AV. BOLIVAR 1789 - PUEBLO LIBRE |
| 3 | 3 | AV. SAENZ PEÑA 590 - CALLAO |
| 4 | 4 | PANAMERICANA NORTE KM. 17.5 - LOS OLIVOS |
| 5 | 5 | AV. ESPAÑA 775 - BREÑA |

# 3. Estableciendo filtros de fila en la instrucción SELECT

En la cláusula WHERE, **condición_filas** es una expresión lógica que establece la condición que deben cumplir las filas a mostrar en el resultado de la consulta. Para construir la expresión lógica utilice operadores relacionales ó de comparación, operadores lógicos como NOT, AND y OR, y operadores lógicos SQL como LIKE, BETWEEN e IN.

**Ejercicio 5.8: Uso del operador de igualdad (=)**

```
USE QhatuPERU
go

SELECT CodArticulo, DescripcionArticulo,
     Presentacion, PrecioProveedor
FROM ARTICULO
WHERE CodLinea = 5
go
```

Muestra la lista de artículos de la línea **5** (licores y gaseosas).

Resultados | Mensajes

| | CodArticulo | DescripcionArticulo | Presentacion | PrecioProveedor |
|---|---|---|---|---|
| 1 | 109 | INCA KOLA DIET DESCARTABLE | BOTELLA 1.5 LT | 2,56 |
| 2 | 110 | INCA KOLA DESCARTABLE | BOTELLA 1.5 LT | 2,56 |
| 3 | 111 | INCA KOLA DIET | BOTELLA 500 ML | 1,36 |
| 4 | 112 | INCA KOLA PLASTIFORMA DESCARTABLE | BOTELLA 500 ML | 1,36 |
| 5 | 113 | SPRITE DESCARTABLE | BOTELLA 2 LT | 2,80 |
| 6 | 114 | SPRITE CONTOUR | BOTELLA 500 ML | 1,36 |
| 7 | 115 | SPRITE DESCARTABLE | BOTELLA 1.5 LT | 2,40 |
| 8 | 116 | SPRITE RETORNABLE | BOTELLA 2 LT | 2,40 |
| 9 | 117 | TRIPLE DIET NO RETORNABLE | BOTELLA 1.5 LT | 2,04 |
| 10 | 118 | 7 UP DESCARTABLE | BOTELLA 1.5 LT | 2,24 |
| 11 | 119 | WHISKY JOHNNIE WALKER ETIQUETA ROJA | BOTELLA 750 ML | 12,56 |

**Ejercicio 5.9: Uso del operador diferente (<>, !=)**

```
SELECT CodProveedor, NomProveedor,
     Telefono, Departamento
FROM PROVEEDOR
WHERE Departamento <> 'Lima'
go
```

Muestra a los proveedores que no están ubicados en el departamento de **Lima**.

Resultados | Mensajes

| | CodProveedor | NomProveedor | Telefono | Departamento |
|---|---|---|---|---|
| 1 | 1 | LACTEOS DEL CENTRO | NULL | JUNIN |
| 2 | 9 | EMBOTELLADORA LA PREFERIDA | NULL | LA LIBERTAD |
| 3 | 10 | DROGUERIA MAHAN | NULL | AREQUIPA |
| 4 | 11 | QUIMICA DEL NORTE | NULL | LAMBAYEQUE |
| 5 | 14 | GOLOSINAS Y ANTOJOS | NULL | AREQUIPA |

**Ejercicio 5.10: Uso del operador menor que (<)**

```
SELECT CodArticulo, DescripcionArticulo,
     Presentacion, PrecioProveedor
FROM ARTICULO
WHERE PrecioProveedor < 1.50
go
```

Muestra los artículos con un precio menor a **1.50**.

Resultados | Mensajes

| | CodArticulo | DescripcionArticulo | Presentacion | PrecioProveedor |
|---|---|---|---|---|
| 1 | 2 | CARAMELOS SURTIDO DE FRUTAS | PAQUETE 450 GR | 1,00 |
| 2 | 4 | CARAMELOS FRUTAS MASTICABLES | PAQUETE 454 GR | 1,30 |
| 3 | 5 | CHUPETES LOLY AMBROSOLI | KILOGRAMO | 1,20 |
| 4 | 10 | WAFER CHOCOLATE FIELD | PAQUETE X 9 UNIDADES | 0,70 |
| 5 | 11 | CHOCOLATE BARRA REGULAR | BARRA 2 ONZAS | 0,40 |
| 6 | 13 | CHOCOLATE BARRA MILKY WAY | BARRA 2.15 ONZAS | 0,80 |
| 7 | 14 | SNICKERS BAR KING SIZE | BARRA 3.7 ONZAS | 1,20 |
| 8 | 15 | CHOCOLATE BARRA MILK DOVE | UNIDAD | 1,30 |
| 9 | 16 | CHOCOLATE BARRA DARK DOVE | UNIDAD | 1,30 |
| 10 | 18 | GALLETAS CHIPS AHOY | PAQUETE X 6 UNIDADES | 1,00 |
| 11 | 19 | GALLETAS TUAREG COSTA | PAQUETE X 6 UNIDADES | 1,20 |

**Ejercicio 5.11: Uso del operador mayor que (>)**

```
SELECT CodArticulo, DescripcionArticulo
FROM ARTICULO
WHERE DescripcionArticulo > 'T'
go
```

Muestra los artículos cuya descripción es posterior a **t** cuando todos los artículos se ordenan alfabéticamente por su descripción.

Resultados | Mensajes

| | CodArticulo | DescripcionArticulo |
|---|---|---|
| 1 | 117 | TRIPLE DIET NO RETORNABLE |
| 2 | 10 | WAFER CHOCOLATE FIELD |
| 3 | 122 | WHISKY CHIVAS REGAL |
| 4 | 120 | WHISKY JOHNNIE WALKER ETIQUETA NEGRA |
| 5 | 119 | WHISKY JOHNNIE WALKER ETIQUETA ROJA |
| 6 | 121 | WHISKY SOMETHING SPECIAL |
| 7 | 123 | WHISKY YE MONKS |
| 8 | 95 | YOGURT GLORIA FRESA |
| 9 | 97 | YOGURT LAIVE FRESA |
| 10 | 96 | YOGURT YOLEIT FRESA |

**Ejercicio 5.12: Manipulación de datos de tipo fecha-hora con los operadores de comparación**

1. Ejecute la siguiente instrucción en su ventana de consulta:

```
SET DATEFORMAT DMY
go

SELECT NumOrden, FechaOrden
FROM ORDEN_COMPRA
go
```

La lista muestra 2 órdenes cuya fecha de emisión es el **11 de abril del 2013**.

Resultados | Mensajes

| | NumOrden | FechaOrden |
|---|---|---|
| 1 | 1 | 2013-04-09 20:38:06.400 |
| 2 | 2 | 2013-04-09 20:38:06.410 |
| 3 | 3 | 2013-04-09 20:38:06.417 |
| 4 | 4 | 2013-04-09 20:38:06.420 |
| 5 | 5 | 2013-04-11 20:38:06.423 |
| 6 | 6 | 2013-04-11 20:38:06.440 |
| 7 | 7 | 2013-04-12 20:38:06.440 |
| 8 | 8 | 2013-04-12 20:38:06.443 |
| 9 | 9 | 2013-04-12 20:38:06.443 |
| 10 | 10 | 2013-04-12 20:38:06.453 |
| 11 | 11 | 2013-04-12 20:38:06.477 |

2. Ahora, ejecute la siguiente instrucción:

```
SELECT NumOrden, FechaOrden
FROM ORDEN_COMPRA
WHERE FechaOrden = '11/04/2013'
go
```

El listado resultante no muestra ninguna orden para el **11 de abril del 2013**. El problema no es de formato de fecha ya que previamente se le informó al servidor que la cadena de fecha se le está enviando en formato **DMY**.

La columna **FechaOrden** es de tipo fecha-hora; por lo tanto almacena la fecha y además una hora, es decir, que representa un instante en el tiempo. La instrucción SELECT que se ha enviado no especifica la hora, por lo que el motor de búsqueda asume que estamos buscando la fecha '11/94/2013 00:00:00.000'.

3. Ahora, ejecute la siguiente consulta:

```
SELECT NumOrden, FechaOrden
FROM ORDEN_COMPRA
WHERE FechaOrden <= '11/04/2013'
go
```

El listado resultante muestra las órdenes cuya fecha de salida es anterior al **11 de abril del 2013**, pero las órdenes de dicha fecha no se muestran.

Al no indicar la hora en la fecha de búsqueda, el motor asume que la condición de búsqueda es la siguiente

```
WHERE FechaOrden <= '11/04/2013 00:00:00.000'
go
```

Entonces, ¿cómo debemos especificar la búsqueda sin tener que indicar un instante preciso del tiempo como valor de búsqueda?

## 3.1. Uso de la función CONVERT() con datos de tipo fecha-hora

**Sintáxis**

```
CONVERT( char(n), expresión_fecha, estilo )
```

Convierte **expresión_fecha** a una cadena con formato de fecha. La cadena tiene longitud **n,** y el formato del dato se establece con el valor de **estilo**.

- **char(n)**, es el tipo y la longitud de la cadena con formato de fecha resultante.
- **expresión_fecha**, representa la fecha cuyo formato de presentación se desea cambiar.
- **estilo**, indica el formato de presentación de la cadena resultante.

El siguiente cuadro muestra algunos valores de estilo para la función CONVERT.

| Año con 2 dígitos | Año con 4 dígitos | Formato | Salida |
|---|---|---|---|
| 1 | 101 | USA | mm/dd/aa |
| 2 | 102 | ANSI | aa.mm.dd |

| | | | |
|---|---|---|---|
| 3 | 103 | Británico/Francés | dd/mm/aa |
| 4 | 104 | Alemán | dd.mm.aa |
| 5 | 105 | Italiano | dd-mm-aa |
| 6 | 106 | | dd mes aa |
| 7 | 107 | | Mes dd, aa |

**Ejercicio 5.13: Búsqueda en columna de tipo fecha-hora**

Se desea obtener el listado de las órdenes cuya fecha de emisión es anterior ó igual al **11 de abril del 2013**.

```
SELECT NumOrden, FechaOrden
FROM ORDEN_COMPRA
WHERE CONVERT(CHAR(10), FechaOrden, 102)
     <= '2013.04.11'
go

SELECT NumOrden, FechaOrden
FROM ORDEN_COMPRA
WHERE CONVERT(CHAR(10), FechaOrden, 103)
     <= '11/04/2013'
go
```

Resultados | Mensajes

| | NumOrden | FechaOrden |
|---|---|---|
| 1 | 1 | 2013-04-09 20:38:06.400 |
| 2 | 2 | 2013-04-09 20:38:06.410 |
| 3 | 3 | 2013-04-09 20:38:06.417 |
| 4 | 4 | 2013-04-09 20:38:06.420 |
| 5 | 5 | 2013-04-11 20:38:06.423 |
| 6 | 6 | 2013-04-11 20:38:06.440 |

Los datos de tipo fecha-hora se convierten a cadenas y se comparan con una cadena con el formato de fecha adecuado.

**Nota:** a partir de la versión 2012 de MS SQL Server ya puede crear columnas que solo almacenen la fecha ya que se ha incluido el tipo de dato **date**.

## 3.2. Búsqueda basada en cadena de caracteres – El operador LIKE

**Sintáxis**

```
SELECT * | lista_columnas
FROM nombre_tabla
WHERE columna LIKE expresión_cadena_a_buscar
```

- **columna**, es la columna en la que se busca la cadena de caracteres.
- **expresión_cadena_a_buscar**, indica como debe ser la cadena que se está buscando en columna. La expresión admite comodines.

**Los comodines en el operador LIKE**

El siguiente cuadro muestra los comodines que puede utilizar con el operador LIKE.

| Comodín | Descripción |
|---|---|
| **%** | Indica que en la posición del comodín puede ir cualquier cadena de caracteres, incluso una cadena nula. |
| **_** | Indica que en la posición del comodín puede ir cualquier carácter no nulo. |
| **[abc]** | Establece el conjunto de caracteres válidos en la posición del comodín. |
| **[a-b]** | Establece el rango de caracteres válidos en la posición del comodín. |
| **^** | Excluir. Indica que el caracter, conjunto de caracteres, ó rango de caracteres que sigue al símbolo **^** no debe figurar en el resultado de la consulta. |

**Ejercicio 5.14: Uso del comodín %**

Se desea obtener una relación de los artículos que contienen la cadena **'gloria'** en su descripción.

```
SELECT CodArticulo, DescripcionArticulo
FROM ARTICULO
WHERE DescripcionArticulo LIKE 'gloria'
```

```
go
```

El resultado tiene o filas. La búsqueda especifica que el contenido de la columna **DescripcionArticulo** debe ser exactamente la cadena **'gloria'**.

```
SELECT CodArticulo, DescripcionArticulo
FROM ARTICULO
WHERE DescripcionArticulo LIKE 'gloria%'
go
```

El resultado tiene o filas. La expresión de búsqueda indica que el contenido de la columna **DescripcionArticulo** debe empezar con la cadena **'gloria'**.

```
SELECT CodArticulo, DescripcionArticulo
FROM ARTICULO
WHERE DescripcionArticulo LIKE '%gloria'
go
```

El resultado tiene o filas. La expresión de búsqueda indica que el contenido de la columna **DescripcionArticulo** debe terminar con la cadena **'gloria'**.

```
SELECT CodArticulo, DescripcionArticulo
FROM ARTICULO
WHERE DescripcionArticulo LIKE '%gloria%'
go
```

La expresión de búsqueda indica que la cadena **'gloria'** puede encontrarse en cualquier posición del contenido de la columna **DescripcionArticulo**.

Resultados | Mensajes

| | CodArticulo | DescripcionArticulo |
|---|---|---|
| 1 | 95 | YOGURT GLORIA FRESA |

**Ejercicio 5.15: Uso del comodín _**

```
SELECT CodArticulo, DescripcionArticulo
FROM ARTICULO
WHERE DescripcionArticulo LIKE '%blanc_'
go
```

La expresión de búsqueda indica que el contenido de la columna **DescripcionArticulo** debe tener la cadena **'blanc'** antes del último caracter de la descripción. El último caracter puede ser cualquiera.

Resultados | Mensajes

| | CodArticulo | DescripcionArticulo |
|---|---|---|
| 1 | 126 | DETERGENTE LIMON ECOLOGICO MAGIA BLANCA |
| 2 | 64 | JABON DOVE BLANCO |
| 3 | 65 | JABON ROSAS Y LIMON BLANCO |
| 4 | 68 | PASTA DENTAL KOLYNOS SUPER BLANCO |

**Ejercicio 5.16: Uso del comodín [abc]**

```
SELECT CodArticulo, DescripcionArticulo
FROM ARTICULO
WHERE DescripcionArticulo LIKE '[pdf]%'
ORDER BY DescripcionArticulo
go
```

La expresión de búsqueda indica que el primer caracter en el contenido de la columna **DescripcionArticulo** puede ser cualquiera del conjunto **(pdf)**; es decir, que el primer caracter puede ser **p**, ó **d**, ó **f**.

Resultados | Mensajes

| | CodArticulo | DescripcionArticulo |
|---|---|---|
| 1 | 124 | DETERGENTE C/BLANQUEADOR ARIEL |
| 2 | 131 | DETERGENTE C/BLANQUEADOR ARIEL |
| 3 | 128 | DETERGENTE LIMON ARIEL |
| 4 | 125 | DETERGENTE LIMON ARIEL |
| 5 | 129 | DETERGENTE LIMON ARIEL |
| 6 | 126 | DETERGENTE LIMON ECOLOGICO MAGIA BLANCA |
| 7 | 130 | DETERGENTE LIMON INVICTO |
| 8 | 132 | DETERGENTE LIMON OPAL |
| 9 | 127 | DETERGENTE PODER LIMON ACE |
| 10 | 106 | DORINA CLASICA |
| 11 | 6 | FRUNA SURTIDA DONOFRIO |
| 12 | 23 | FUDGE SHOPPE DELUXE GRAHAMS |
| 13 | 24 | FUDGE SHOPPE STICKS KEEB |
| 14 | 77 | P.H. BLANCO ROLL KLEENEX |
| 15 | 76 | P.H. BLANCO ROLL KLEENEX |
| 16 | 74 | P.H. BLANCO SUAVE (ROJA) |
| 17 | 75 | P.H. BLANCO SUAVE DOBLE HOJA (AZUL) |
| 18 | 81 | P.H. ELITE BLANCO EXTRA |

**Ejercicio 5.17: Uso del comodín [a-b]**

```
SELECT CodArticulo, DescripcionArticulo
FROM ARTICULO
WHERE DescripcionArticulo LIKE '[d-p]%'
ORDER BY DescripcionArticulo
go
```

La expresión de búsqueda indica que el primer caracter en el contenido de la columna **DescripcionArticulo** puede ser cualquiera del rango que va desde el caracter **d** hasta el caracter **p**.

Resultados | Mensajes

| | CodArticulo | DescripcionArticulo |
|---|---|---|
| 1 | 124 | DETERGENTE C/BLANQUEADOR ARIEL |
| 2 | 131 | DETERGENTE C/BLANQUEADOR ARIEL |
| 3 | 128 | DETERGENTE LIMON ARIEL |
| 4 | 125 | DETERGENTE LIMON ARIEL |
| 5 | 129 | DETERGENTE LIMON ARIEL |
| 6 | 126 | DETERGENTE LIMON ECOLOGICO MAGIA BLANCA |
| 7 | 130 | DETERGENTE LIMON INVICTO |
| 8 | 132 | DETERGENTE LIMON OPAL |
| 9 | 127 | DETERGENTE PODER LIMON ACE |
| 10 | 106 | DORINA CLASICA |
| 11 | 6 | FRUNA SURTIDA DONOFRIO |
| 12 | 23 | FUDGE SHOPPE DELUXE GRAHAMS |
| 13 | 24 | FUDGE SHOPPE STICKS KEEB |
| 14 | 18 | GALLETAS CHIPS AHOY |
| 15 | 25 | GALLETAS DELICE |
| 16 | 21 | GALLETAS SURTIDAS BUTTER COOKIES |
| 17 | 19 | GALLETAS TUAREG COSTA |
| 18 | 20 | GALLETAS VAINILLA COSTA |

Ahora, repita la consulta anterior, pero invierta el rango en la expresión de búsqueda.

```
SELECT CodArticulo, DescripcionArticulo
FROM ARTICULO
WHERE DescripcionArticulo LIKE '[p-d]%'
ORDER BY DescripcionArticulo
go
```

El resultado no devuelve filas, pero el sistema no genera ningún mensaje de error. ¿Por qué? Revise el tema **Intercalación** ó **Server Collation** en la ayuda del producto.

**Ejercicio 5.18: Uso del comodín ^**

```
SELECT CodArticulo, DescripcionArticulo
FROM ARTICULO
WHERE DescripcionArticulo LIKE '[^pdf]%'
ORDER BY DescripcionArticulo
go
```

La expresión de búsqueda indica que el primer caracter en el contenido de la columna **DescripcionArticulo** puede ser cualquiera menos los especificados en el conjunto **(pdf)**; es decir, que el primer caracter no puede ser **p**, ó **d**, ó **f**.

Resultados | Mensajes

| | CodArticulo | DescripcionArticulo |
|---|---|---|
| 1 | 118 | 7 UP DESCARTABLE |
| 2 | 53 | ACEITE BABY JOHNSONS |
| 3 | 51 | ACEITE BABY JOHNSONS C/ALOE Y VIT. E |
| 4 | 54 | ACEITE BABY JOHNSONS C/ALOE Y VIT. E |
| 5 | 52 | ACEITE JOHNSONS |
| 6 | 56 | ACEITE JOHNSONS CREMOSO |
| 7 | 58 | ACEITE P/BEBES CHICCO |
| 8 | 55 | ACEITE P/BEBES DR. ZAIDMAN |
| 9 | 57 | ACEITE P/BEBES NINET |
| 10 | 1 | CARAMELOS BASTON VIENA ARCOR |
| 11 | 4 | CARAMELOS FRUTAS MASTICABLES |
| 12 | 3 | CARAMELOS FRUTAS SURTIDA ARCOR |
| 13 | 2 | CARAMELOS SURTIDO DE FRUTAS |
| 14 | 62 | CEPILLO ADVANTAGE-60 CONTROL |
| 15 | 61 | CEPILLO DENTAL FLEX ADULTO |
| 16 | 63 | CEPILLO MASTER ADULTO |
| 17 | 59 | CEPILLO ODONTOLOGICA EXTRA MEDIA... |
| 18 | 60 | CEPILLO ODONTOLOGICA EXTRA SUAVE |
| 19 | 16 | CHOCOLATE BARRA DARK DOVE |

Ahora, ejecute la siguiente consulta:

```
SELECT CodArticulo, DescripcionArticulo
FROM ARTICULO
WHERE DescripcionArticulo LIKE '[^d-p]%'
ORDER BY DescripcionArticulo
go
```

La expresión de búsqueda indica que el primer caracter en el contenido de la columna **DescripcionArticulo** puede ser cualquiera menos los que se que encuentran en el rango que va desde el caracter **d** hasta el caracter **p**.

Resultados | Mensajes

| | CodArticulo | DescripcionArticulo |
|---|---|---|
| 34 | 5 | CHUPETES LOLY AMBROSOLI |
| 35 | 98 | CREAM CHESSE LAIVE |
| 36 | 93 | CREMA DE LECHE DUPRE |
| 37 | 92 | CREMA DE LECHE LAIVE |
| 38 | 94 | CREMA DE LECHE NESTLE |
| 39 | 99 | CREMA DE QUESO LAIVE |
| 40 | 100 | QUESO CREMA MILKITO |
| 41 | 42 | SALCHICHA DE HUACHO |
| 42 | 84 | SHAMPOO ALBERTO VO5 CABELLO NOR... |
| 43 | 86 | SHAMPOO ALBERTO VO5 JOJOBA |
| 44 | 85 | SHAMPOO HERBAL CABELLO FINO |
| 45 | 87 | SHAMPOO HERBAL CABELLO GRASO |
| 46 | 83 | SHAMPOO HERBAL CABELLO NORMAL |
| 47 | 82 | SHAMPOO HERBAL CABELLO SECO/DA... |
| 48 | 91 | SHAMPOO PANTENE CABELLO GRASO |
| 49 | 89 | SHAMPOO PANTENE CABELLO NORMAL |

## 3.3. Búsqueda basada en rango de valores – El operador BETWEEN

El operador BETWEEN permite ejecutar consultas que ejecutan búsquedas basadas en rango de valores numéricos, valores de cadena, ó valores de fecha.

**Sintáxis**

```
SELECT * | lista_columnas
FROM nombre_tabla
WHERE columna BETWEEN valor_inicial AND
valor_final
```

BETWEEN indica que el valor en columna debe encontrarse en el rango definido por **valor_inicial** y **valor_final**.

- **columna**, es la columna en la que se busca según el rango especificado por valor_inicial y valor_final.
- **valor_inicial**, **valor_final**, establecen los límites del rango de valores en el que se basa la búsqueda.

BETWEEN define siempre un intervalo cerrado.

**Ejercicio 5.19: Búsqueda basada en rango numérico**

```
USE QhatuPERU
go

SELECT CodArticulo, DescripcionArticulo,
     Presentacion, PrecioProveedor
FROM ARTICULO
WHERE PrecioProveedor BETWEEN 5.50 AND 7
go
```

Entrega una lista de artículos cuyo **PrecioProveedor** se encuentra en el rango que va desde **5.50** hasta **7.00** incluyendo a los límites.

Resultados | Mensajes

| | CodArticulo | DescripcionArticulo | Presentacion | PrecioProveedor |
|---|---|---|---|---|
| 1 | 37 | HOT DOG LAIVE PELADO | KILOGRAMO | 5,50 |
| 2 | 38 | HOT DOG LA SEGOVIANA | KILOGRAMO | 6,80 |
| 3 | 84 | SHAMPOO ALBERTO VO5 CABELLO NORMAL | FRASCO 15 ONZAS | 5,50 |
| 4 | 86 | SHAMPOO ALBERTO VO5 JOJOBA | FRASCO 15 ONZAS | 5,50 |
| 5 | 87 | SHAMPOO HERBAL CABELLO GRASO | FRASCO 355 ML | 6,90 |
| 6 | 88 | SHAMPOO SEDAL CERAMIDAS 2 EN 1 | FRASCO 315 ML | 5,90 |
| 7 | 90 | SHAMPOO SEDAL DUO | FRASCO 315 ML | 5,90 |
| 8 | 108 | LECHE CULTIVADA MILKITO FRESA | ENVASE 1 LT | 6,00 |

**Ejercicio 5.20: Búsqueda basada en rango de valores cadena**

```
SELECT CodArticulo, DescripcionArticulo
FROM ARTICULO
WHERE DescripcionArticulo BETWEEN 'fru' AND 'hot'
go
```

Entrega una lista de artículos cuya descripción se encuentra en el rango que va desde la cadena **'fru'** hasta la cadena **'hot'**.

Resultados | Mensajes

| | CodArticulo | DescripcionArticulo |
|---|---|---|
| 1 | 6 | FRUNA SURTIDA DONOFRIO |
| 2 | 23 | FUDGE SHOPPE DELUXE GRAHAMS |
| 3 | 24 | FUDGE SHOPPE STICKS KEEB |
| 4 | 18 | GALLETAS CHIPS AHOY |
| 5 | 25 | GALLETAS DELICE |
| 6 | 21 | GALLETAS SURTIDAS BUTTER COOKIES |
| 7 | 19 | GALLETAS TUAREG COSTA |
| 8 | 20 | GALLETAS VAINILLA COSTA |

Observe que el resultado no incluye al producto 'HOT DOG' ya que esta cadena se encuentra después de la cadena 'HOT', por lo tanto está fuera del rango especificado.

**Ejercicio 5.21: Búsqueda basada en rango de valores fecha para columnas fecha-hora**

Primero, veamos que fechas tenemos registradas en la tabla GUIA_ENVIO.

Resultados | Mensajes

| | NumGuia | FechaSalida |
|---|---|---|
| 56 | 14 | 2013-04-04 20:38:04.663 |
| 57 | 19 | 2013-04-04 20:38:04.733 |
| 58 | 24 | 2013-04-04 20:38:04.770 |
| 59 | 29 | 2013-04-04 20:38:04.833 |
| 60 | 34 | 2013-04-05 20:38:04.880 |
| 61 | 39 | 2013-04-05 20:38:04.913 |
| 62 | 44 | 2013-04-05 20:38:04.960 |
| 63 | 49 | 2013-04-05 20:38:05.023 |
| 64 | 54 | 2013-04-05 20:38:05.070 |
| 65 | 59 | 2013-04-05 20:38:05.130 |
| 66 | 64 | 2013-04-06 20:38:05.197 |

Observe que tenemos varias guías cuya **FechaSalida** es **5 de abril del 2013**.

Ahora, trate de obtener una lista de las guías emitidas desde el **25 de marzo del 2013** al **5 de abril del 2013**.

```
SET DATEFORMAT DMY
go

SELECT NumGuia, FechaSalida
```

```
FROM GUIA_ENVIO
WHERE FechaSalida BETWEEN '25/03/2013'
     AND '05/04/2013'
ORDER BY FechaSalida
go
```

Resultados | Mensajes

| | NumGuia | FechaSalida |
|---|---|---|
| 29 | 53 | 2013-03-31 20:38:05.060 |
| 30 | 58 | 2013-03-31 20:38:05.113 |
| 31 | 63 | 2013-04-01 20:38:05.180 |
| 32 | 68 | 2013-04-01 20:38:05.220 |
| 33 | 73 | 2013-04-01 20:38:05.283 |
| 34 | 78 | 2013-04-01 20:38:05.330 |
| 35 | 83 | 2013-04-01 20:38:05.400 |
| 36 | 88 | 2013-04-01 20:38:05.487 |
| 37 | 9 | 2013-04-04 20:38:04.597 |
| 38 | 14 | 2013-04-04 20:38:04.663 |
| 39 | 19 | 2013-04-04 20:38:04.733 |
| 40 | 24 | 2013-04-04 20:38:04.770 |
| 41 | 29 | 2013-04-04 20:38:04.833 |

Las guías del **5 de abril del 2013** no se muestran en el listado. ¿Por qué?

Para la consulta ejecutada, el rango establecido señala que empieza a las **00:00 horas del 25 de marzo del 2013**, y termina a las **00:00 horas del 5 de abril del 2013**. Por lo tanto, cualquier guía emitida después de las 00:00 horas del día 5 de abril se encuentra fuera del rango.

Para incluir las guías del día 5 de abril podemos escribir la consulta así:

```
SELECT NumGuia, FechaSalida
FROM GUIA_ENVIO
WHERE FechaSalida BETWEEN '25/03/2013'
     AND '05/04/2013 23:59:59.999'
ORDER BY FechaSalida
go
```

Resultados | Mensajes

| | NumGuia | FechaSalida |
|---|---|---|
| 35 | 83 | 2013-04-01 20:38:05.400 |
| 36 | 88 | 2013-04-01 20:38:05.487 |
| 37 | 9 | 2013-04-04 20:38:04.597 |
| 38 | 14 | 2013-04-04 20:38:04.663 |
| 39 | 19 | 2013-04-04 20:38:04.733 |
| 40 | 24 | 2013-04-04 20:38:04.770 |
| 41 | 29 | 2013-04-04 20:38:04.833 |
| 42 | 34 | 2013-04-05 20:38:04.880 |
| 43 | 39 | 2013-04-05 20:38:04.913 |
| 44 | 44 | 2013-04-05 20:38:04.960 |
| 45 | 49 | 2013-04-05 20:38:05.023 |
| 46 | 54 | 2013-04-05 20:38:05.070 |
| 47 | 59 | 2013-04-05 20:38:05.130 |

## 3.4. Búsqueda basada en conjunto de valores – El operador IN

El operador IN permite ejecutar consultas que ejecutan búsquedas basadas en conjuntos de valores numéricos, valores de cadena, ó valores de fecha.

**Sintáxis**

```
SELECT * | lista_columnas
FROM nombre_tabla
WHERE columna [NOT]IN ( conjunto_de_valores )
```

IN indica que el valor en **columna** debe encontrarse (IN), ó no debe encontrarse (NOT IN) en el conjunto definido por **conjunto_de_valores**.

- **columna**, es la columna en la que se busca según el conjunto de valores especificado en conjunto_de_valores.
- **conjunto_de_valores**, establece el conjunto de valores en el que se basa la búsqueda.

**Ejercicio 5.22: Búsqueda basada en conjunto de valores**

```
SELECT CodArticulo, DescripcionArticulo,
CodProveedor
FROM ARTICULO
WHERE CodProveedor IN (7,1,3)
go
```

Muestra un listado de los artículos registrados para los proveedores cuyo **CodProveedor** es **7** ó **1** ó **3**.

Resultados | Mensajes

| | CodArticulo | DescripcionArticulo | CodProveedor |
|---|---|---|---|
| 1 | 26 | JAMONADA LAIVE | 1 |
| 2 | 32 | JAMON INGLES LAIVE | 1 |
| 3 | 36 | JAMON YORK SALCHICHERIA ALEMANA | 3 |
| 4 | 37 | HOT DOG LAIVE PELADO | 1 |
| 5 | 40 | HOT DOG CERDEÑA | 3 |
| 6 | 44 | CHORIZO PARRILLERO LAIVE | 1 |
| 7 | 47 | CHORIZO PARRILLERO CERDEÑA | 3 |
| 8 | 50 | CHORIZO PARRILLERO CATALANES | 3 |
| 9 | 74 | P.H. BLANCO SUAVE (ROJA) | 7 |
| 10 | 75 | P.H. BLANCO SUAVE DOBLE HOJA (AZUL) | 7 |
| 11 | 78 | P.H. ELITE DOBLE HOJA | 7 |
| 12 | 79 | P.H. ELITE BLANCO EXTRA | 7 |
| 13 | 81 | P.H. ELITE BLANCO EXTRA | 7 |
| 14 | 92 | CREMA DE LECHE LAIVE | 1 |
| 15 | 97 | YOGURT LAIVE FRESA | 1 |
| 16 | 98 | CREAM CHESSE LAIVE | 1 |
| 17 | 99 | CREMA DE QUESO LAIVE | 1 |

# 4. MANIPULACIÓN DE VALORES NULL

Un valor NULL indica que el valor es desconocido, no aplicable, ó que simplemente se registrará posteriormente. Un valor NULL es distinto a una cadena vacía ó al valor cero (0), y también es distinto a cualquier otro valor NULL.

La comparación ú operación entre dos valores NULL, ó entre un valor NULL y cualquier otro valor retorna un valor desconocido (otro valor NULL), ya que cada valor NULL es desconocido y distinto a cualquier otro NULL.

**Ejercicio 5.23: Búsqueda de valores NULL**

Para ilustrar este tema necesitamos una tabla que contenga una columna NULL, y que además dicha columna tenga registrados algunos valores distintos a NULL. Vamos a crear dicha tabla ejemplo.

```
USE QhatuPERU
```

```
go

CREATE TABLE Empleado(
   IdEmpleado int PRIMARY KEY,
   Apellido varchar(30) not null,
   HaberBasico money not null,
   PorcentajeComision decimal(3,1) null )
go

INSERT INTO Empleado
   VALUES(1, 'CASTRO ARENAS', 1200, 5)
INSERT INTO Empleado
   VALUES(2, 'LUNA ESPEJO', 1000, 10)
INSERT INTO Empleado
   VALUES(3, 'SOTO BUENO', 1400, NULL)
INSERT INTO Empleado
   VALUES(4, 'MARQUEZ ARIZAGA', 1500, NULL)
INSERT INTO Empleado
   VALUES(5, 'DAVILA SANCHEZ', 1200, 7.5)
go
```

La siguiente consulta ejecuta una búsqueda basada en valores NULL.

```
SELECT idEmpleado, apellido, haberBasico,
   porcentajeComision
FROM Empleado
WHERE porcentajeComision IS NOT NULL
go
```

Genera una lista de empleados cuyo contenido en la columna **porcentajeComision** es distinto a NULL.

Resultados | Mensajes

| | idEmpleado | apellido | haberBasico | porcentajeComision |
|---|---|---|---|---|
| 1 | 1 | CASTRO ARENAS | 1200,00 | 5.0 |
| 2 | 2 | LUNA ESPEJO | 1000,00 | 10.0 |
| 3 | 5 | DAVILA SANCHEZ | 1200,00 | 7.5 |

**Ejercicio 5.24: Cálculos con columnas que contienen valores NULL**

Asumiendo que cada empleado obtuvo en este mes, ventas por 10,000 Nuevos Soles, genere un listado que muestre el monto total a abonarle a cada uno por su remuneración del mes. El monto total es la suma del haber básico más el monto a pagarle por comisión de las ventas.

```
SELECT idEmpleado, apellido, haberBasico,
   porcentajeComision,
   monto = haberBasico + 10000 *
      porcentajeComision / 100
FROM Empleado
go
```

Resultados | Mensajes

| | idEmpleado | apellido | haberBasico | porcentajeComision | monto |
|---|---|---|---|---|---|
| 1 | 1 | CASTRO ARENAS | 1200,00 | 5.0 | 1700.000000 |
| 2 | 2 | LUNA ESPEJO | 1000,00 | 10.0 | 2000.000000 |
| 3 | 3 | SOTO BUENO | 1400,00 | NULL | NULL |
| 4 | 4 | MARQUEZ ARIZAGA | 1500,00 | NULL | NULL |
| 5 | 5 | DAVILA SANCHEZ | 1200,00 | 7.5 | 1950.000000 |

Observe que el monto resulta NULL para aquellos empleados cuyo **porcentajeComision** es NULL.

**Ejercicio 5.25: Cálculos con columnas que contienen valores NULL – Uso de ISNULL**

**Sintáxis**

```
ISNULL(expresión_a_verificar, valor_de_reemplazo)
```

- **expresión_a_verificar**, es una expresión cuyo valor puede ser NULL.
- **valor_de_reemplazo**, es el valor con que se reemplaza expresión_a_verificar en el caso que ésta fuera NULL.

Calculemos el monto a abonarle a cada empleado haciendo uso de ISNULL.

```
SELECT idEmpleado, apellido, haberBasico,
   porcentajeComision,
   monto = haberBasico +
      10000*ISNULL(porcentajeComision,0)/100
FROM Empleado
go
```

Resultados | Mensajes

| | idEmpleado | apellido | haberBasico | porcentajeComision | monto |
|---|---|---|---|---|---|
| 1 | 1 | CASTRO ARENAS | 1200,00 | 5.0 | 1700.000000 |
| 2 | 2 | LUNA ESPEJO | 1000,00 | 10.0 | 2000.000000 |
| 3 | 3 | SOTO BUENO | 1400,00 | NULL | 1400.000000 |
| 4 | 4 | MARQUEZ ARIZAGA | 1500,00 | NULL | 1500.000000 |
| 5 | 5 | DAVILA SANCHEZ | 1200,00 | 7.5 | 1950.000000 |

# 5. FUNCIONES PARA MANIPULACIÓN DE FECHAS

El siguiente cuadro muestra una lista de las funciones para manipular fechas.

| **Función** | **Descripción** |
|---|---|
| **getdate()** | Retorna la fecha y hora del sistema. |
| **getucdate()** | Retorna la fecha y hora del Meridiano de Greenwich. El valor se obtiene a partir de la configuración regional del sistema. |
| **dateadd(***parte_fecha*, *n*, *fecha***)** | Genera una nueva fecha a partir de *fecha*, añadiéndole a fecha *n* unidades de *parte_fecha*. |
| **datediff(***parte_fecha*, *fecha1*, *fecha2***)** | Entrega la diferencia entre *fecha1* y *fecha2* en las unidades indicadas en *parte_fecha*. |
| **datepart(***parte_fecha*, *fecha***)** | Devuelve, a partir de *fecha*, un valor entero con la *parte_fecha* especificada. |
| **datename(***parte_fecha*, *fecha***)** | Devuelve, a partir de *fecha*, una cadena con la *parte_fecha* especificada. |
| **day(***fecha***)** | Devuelve un entero con la parte del día de *fecha*. |
| **month(***fecha***)** | Devuelve un entero con la parte del mes de *fecha*. |
| **year(***fecha***)** | Devuelve un entero con la parte del año de *fecha*. |

| | |
|---|---|
| **datefromparts(***año, mes, día***)** | Devuelve un dato de tipo date para el *año*, *mes* y *día* especificados. |
| **datetimefromparts(***año, mes, día, hora, minutos, segundos, milisegundos***)** | Devuelve un dato de tipo datetime para los valores especificados. |
| **eomonth(***fecha***)** | Devuelve un dato de tipo date que representa el último día del mes para la *fecha* especificada. |
| **isdate(***expresión***)** | Devuelve 1 si expresión es un valor de tipo date, time ó datetime válido. |

El siguiente cuadro muestra las **parte_fecha** que podemos especificar en las funciones de fecha listadas arriba.

| parte_fecha | Abreviatura | Descripción |
|---|---|---|
| **year** | yy, yyyy | Año de la fecha |
| **quarter** | q, qq | Trimestre del año de la fecha |
| **month** | m, mm | Mes del año de la fecha |
| **week** | ww, wk | Semana del año de la fecha |
| **day** | d, dd | Día de la fecha |
| **dayofyear** | y, dy | Día del año de la fecha |
| **weekday** | dw | Día de la semana de la fecha |
| **hour** | hh | Hora de la fecha |
| **minute** | n, mi | Minutos de la hora de la fecha |
| **second** | s, ss | Segundos de la hora de la fecha |
| **millisecond** | ms | Milisegundos de la hora de la fecha |

**Ejercicio 5.26: Uso de las funciones de fecha y hora**

**Uso de GETDATE()**

```
SELECT GETDATE()
go
```

Entrega la fecha y hora del sistema.

**Uso de GETUTCDATE()**

```
SELECT GETUTCDATE()
go
```

Entrega la fecha y hora del Meridiano de Greenwich.

**Uso de DATEADD()**

```
SELECT NumOrden, FechaOrden,
     FechaPago = DATEADD(day, 30, FechaOrden)
FROM ORDEN_COMPRA
go
```

Genera una nueva fecha añadiéndole 30 días a FechaOrden.

**Uso de DATEDIFF()**

```
SELECT NumOrden, FechaOrden,
     SemanasTranscurridas =
     DATEDIFF(week, FechaOrden, GETDATE())
FROM ORDEN_COMPRA
go
```

Obtiene la diferencia en semanas entre FechaOrden y la fecha del sistema.

**Uso de DATEPART() y DATENAME()**

```
SELECT DATEPART(month, GETDATE())
SELECT DATENAME(month, GETDATE())
go
```

Entrega el número y el nombre del mes de la fecha del sistema

**Uso de DAY(), MONTH() y YEAR()**

```
SELECT DAY(GETDATE())
```

```
SELECT MONTH(GETDATE())
SELECT YEAR(GETDATE())
go
```

Entrega el día, el mes y el año de la fecha del sistema.

**Uso de EOMONTH()**

```
SELECT EOMONTH(GETDATE())
go
```

Entrega la fecha del último día del mes para la fecha del sistema.

# 6. EJERCICIOS PROPUESTOS

Para los siguientes ejercicios debe utilizar las bases de datos RH y EDUCA. Los scripts para crearlas en su servidor los encontrará en el CD que acompaña al libro.

Escriba las instrucciones SELECT para obtener:

1. (RH) El nombre, apellido y teléfono de los empleados.
2. (EDUCA) El nombre, vacantes y precio de cada curso.
3. (EDUCA) El importe que se obtendría si se logra vender todas las vacantes por cada curso.
4. (EDUCA) El importe de lo recaudado hasta el momento de los cursos vendidos.
5. (EDUCA) El importe de lo que se tiene comprometido (cobrado y no cobrado) por los cursos vendidos hasta el momento.
6. (RH) El nombre y apellido de un empleado en una sola columna.
7. (RH) El nombre completo, sueldo, comisión y el total a abonar de cada empleado.
8. (RH) Los empleados del departamento de contabilidad.
9. (RH) Los empleados que se desempeñan como gerentes.
10. (RH) Los empleados de contabilidad cuyo sueldo es mayor a 5,000.00.
11. (EDUCA) Los cursos que aún no tienen profesor.

12. (EDUCA) Los cursos que aún no tienen alumnos matriculados.
13. (RH) Todos los empleados cuyos sus ingresos totales están entre 6,000.00 y 15,000.00, debe incluir los extremos.
14. (RH) Los empleados del departamento de VENTAS cuyo sueldo es menor a 5,000.00.
15. (RH) Los empleados cuyo nombre finaliza con la letra "O".
16. (RH) Los empleados cuyo apellido tiene en la segunda posición la letra "A" u "O".
17. (RH) Los empleados que tienen un sueldo mayor a 3,000 y menor de 10,000.
18. (RH) Los empleados de los departamentos de contabilidad y ventas.
19. (RH) Una lista de los empleados ordenada por fecha de ingreso.
20. (RH) Los empleados cuyos ingresos totales son menores a 8,000.
21. (RH) Los empleados que ingresaron cuyo mes de ingreso a la empresa fue ENERO.
22. (EDUCA) Las matriculas del último mes.
23. (RH) Los locales (ubicaciones) que se encuentran en LIMA.
24. (RH) El importe de la planilla del departamento de ventas, debe incluir el sueldo y la comisión.
25. (RH) Los departamentos que tienen por lo menos un trabajador.