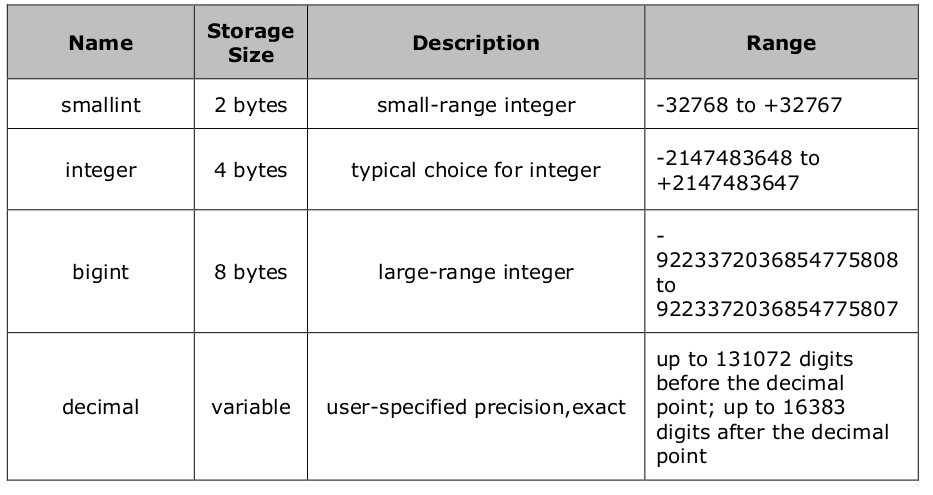
PostgreSQL

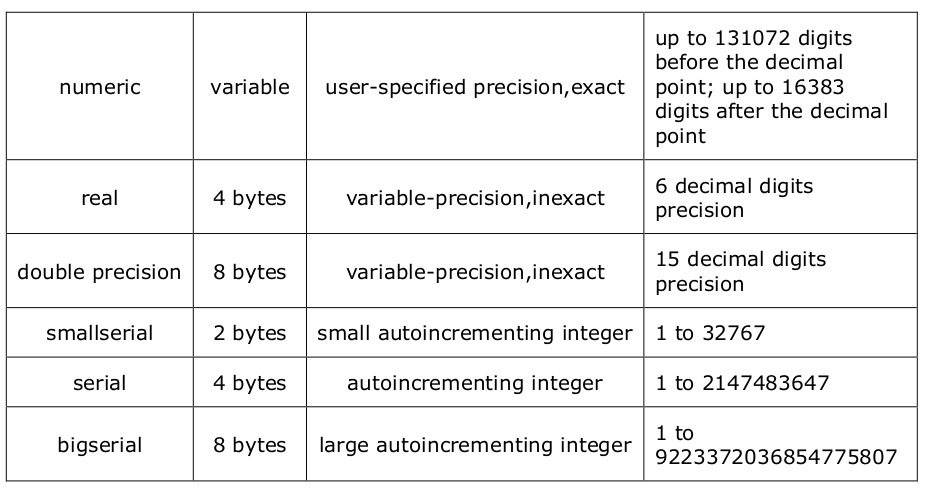
Javier Ramos Fernández

# Tipos de datos

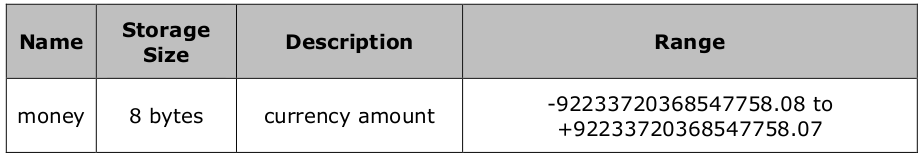
## Primitivos

Numéricos

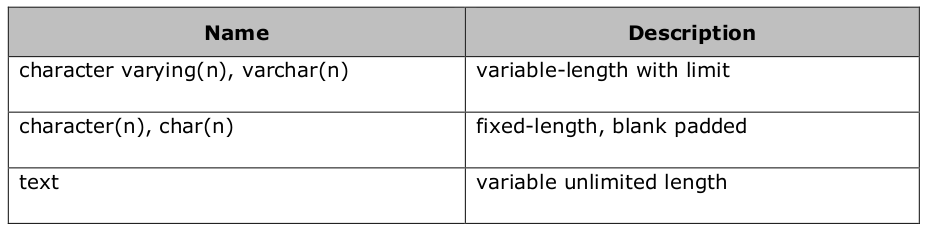




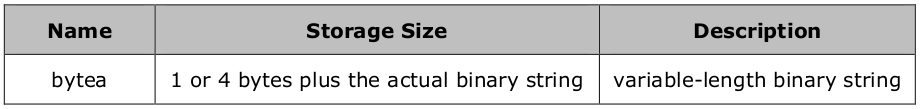
### Monetarios



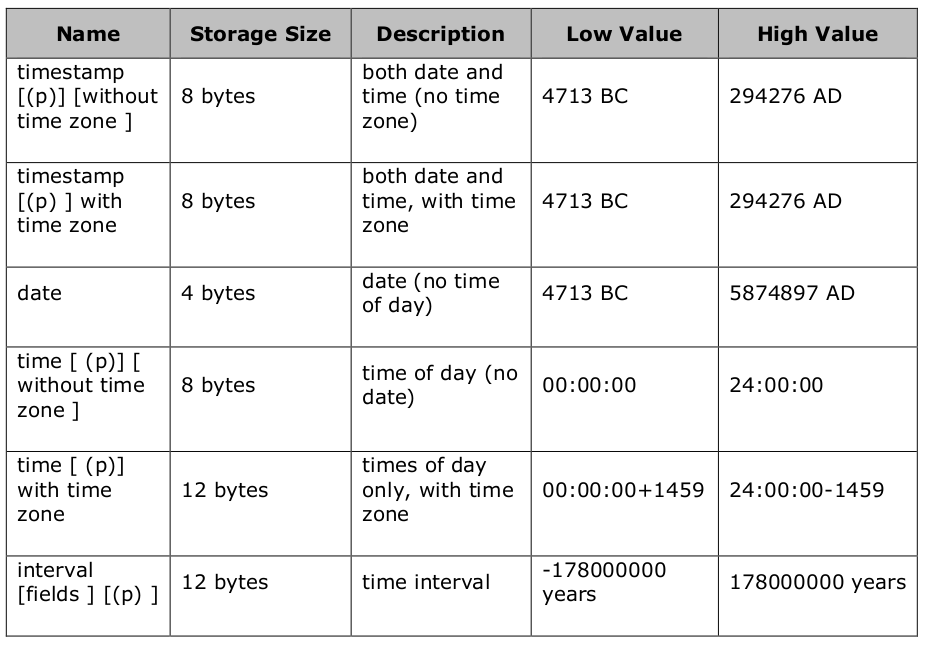
### Caracteres



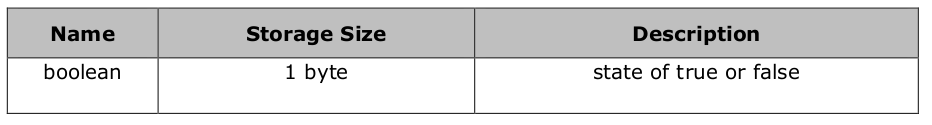
### Binarios



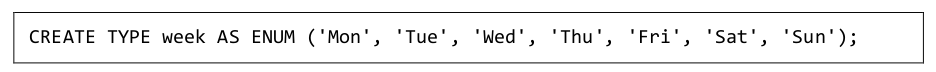
### Fecha/Tiempo



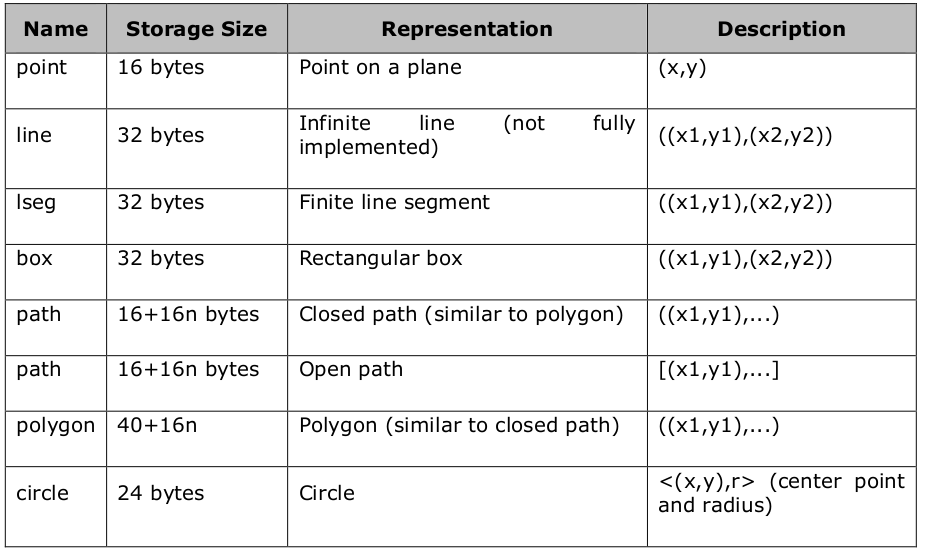
### Booleano



### Enumerado



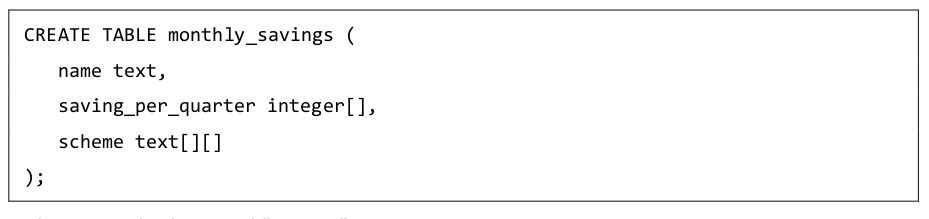
### Geométricos



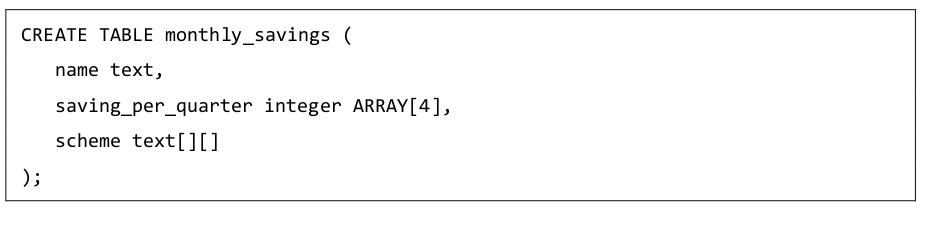
### Arrays

#### Declaración

Los arrays se pueden declarar de dos formas distintas:

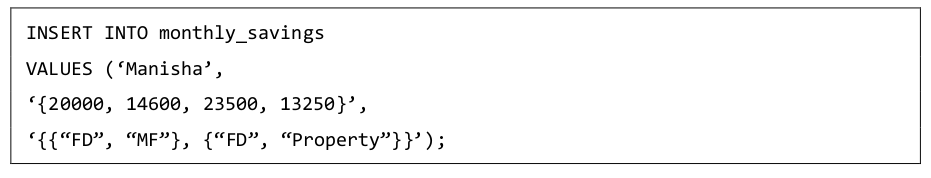


* Utilizando la palabra clave “ARRAY”:



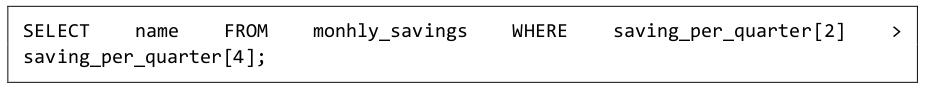
#### Inserción

Los valores de un array se pueden insertar como constantes literales, encerrando los valores del elemento entre llaves y separándolos por comas. Un ejemplo es el siguiente:



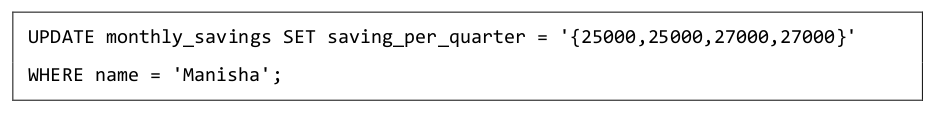
#### Acceso

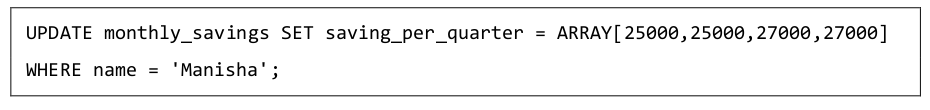
A continuación se muestra un ejemplo para acceder a los arrays. El comando que se da a continuación seleccionará las personas cuyos ahorros sean mayores en el segundo trimestre que en el cuarto trimestre.



#### Modificación

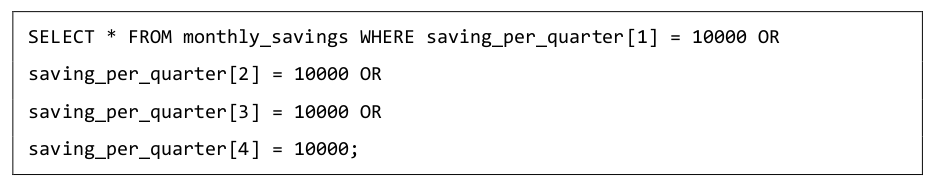
Se puede modificar un array de dos formas distintas:



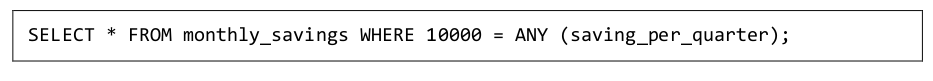


### Búsqueda

Un ejemplo de búsqueda de un array es el siguiente:



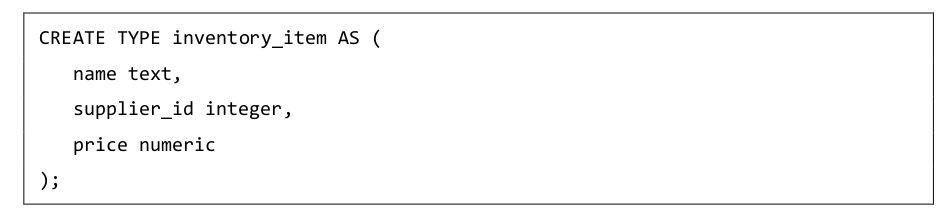
Si se conoce el tamaño de la matriz, se puede utilizar el método de búsqueda indicado anteriormente. Si no, el siguiente ejemplo muestra cómo buscar un array cuando no se conoce el tamaño.



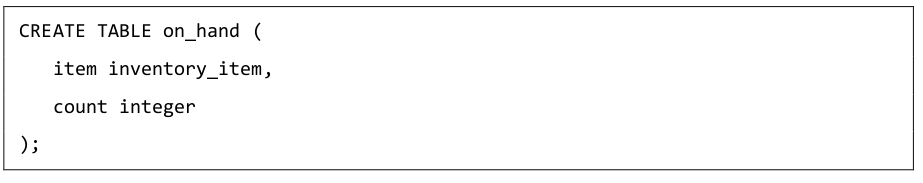
## Compuestos

### Declaración

El siguiente ejemplo muestra cómo declarar un tipo compuesto:



Este tipo de datos se puede utilizar a la hora de crear tablas como se indica a continuación:



### Valores de entrada

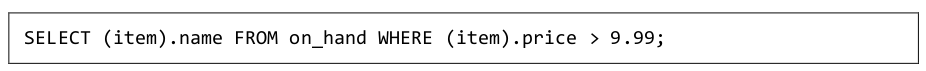
Los valores compuestos se pueden insertar como una constante literal, encerrando los valores de campo entre paréntesis y separándolos por comas. A continuación se muestra un ejemplo:



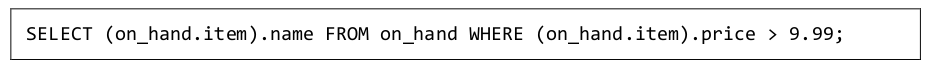
Esto es válido para el *inventory\_item* definido anteriormente. La palabra clave **ROW** es en realidad opcional siempre y cuando se tenga más de un campo en la expresión.

### Acceso

Para acceder a un campo de una columna compuesta, se utiliza un punto seguido del nombre del campo, al igual que se selecciona un campo de un nombre de tabla. Por ejemplo, para seleccionar algunos subcampos de nuestra tabla de ejemplo *on\_hand*, la consulta sería como se muestra a continuación:



También se puede utilizar el nombre de tabla (por ejemplo, en una consulta multitabla), de la siguiente manera:



## 

## Rangos

Los tipos de rango representan tipos de datos que utilizan un rango de datos. Los rangos pueden ser rangos **discretos** (por ejemplo, todos los valores enteros 1 a 10) o rangos continuos (por ejemplo, cualquier punto en el tiempo entre las 10:00am y las 11:00am). Los tipos disponibles incluyen los siguientes rangos:

* **int4range** - Rango de enteros
* **int8range** - Rango de *bigint*
* **numrange** - Rango de *numeric*
* **tsrange** - Rango de marcas temporales sin zona horaria
* **tstzrange** - Rango de marcas temporales con zona horaria
* **daterange** - Rango de fechas

Los tipos de rango admiten límites de rango inclusivos y exclusivos utilizando los caracteres **[]** y **()** respectivamente.

# Exportar una tabla a un archivo .csv

**COPY *tabla\_a\_exportar* TO 'ruta\_del\_archivo' DELIMITER ',' CSV HEADER;**

# Limitar la longitud de lineas largas en las tablas

**\pset format wrapped**

# Acceso remoto a PostgreSQL

En el ordenador servidor:

1. Acceder al fichero **/etc/postgresql/9.3/main/pg\_hba.conf** con permisos de root.
2. Añadir una línea que sea del tipo **host all all [ip\_ordenador\_cliente]/32 md5**.
3. Ejecutar la sentencia **sudo service postgresql restart**.

En el ordenador cliente, ejecutar la sentencia **psql -h [ip\_servidor] -U [nombre\_usuario] -d [base\_de\_datos]**.