

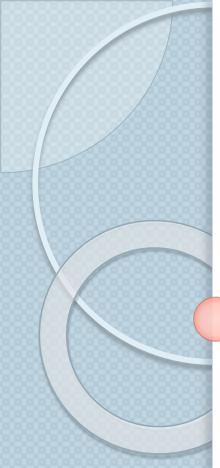


POLÍTICAS DE SEGURIDAD

- Se denomina **SEGURIDAD de la BD** a las medidas que tomamos para protegerla del acceso mal intencionado.

Niveles de protección de la Seguridad:

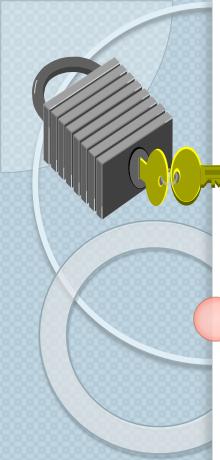
- **Físico:** Protección del local contra el ingreso de intrusos
- **Humano:** Concesión de usuarios y contraseñas sólo a las personas autorizadas
- **Sistema Operativo:** Las mismas medidas de restricción y contraseñas en la Base de Datos, debe establecerse también a nivel del S.O.
- **Sistema de Base de Datos:** El Sistema debe denegar el acceso sobre los objetos a los usuarios a quienes no les fue concedido de manera específica.
- **Red:** Debe también establecerse la seguridad en el software de la red, tanto si ésta fuera una red privada, como en Internet.



SEGURIDAD

Al nivel de la BD, se establece la seguridad en los siguientes niveles:

- Acceso a la creación, modificación y/o borrado de los propios objetos del esquema (tablas y vistas)
- Acceso a la manipulación de los datos de un objeto.
- Acceso selectivo de ciertos datos a través del mecanismo de las vistas.
- Acceso a privilegios de traslado (concesión) de autorizaciones



APLICACIÓN DE UNA POLÍTICA DE SEGURIDAD

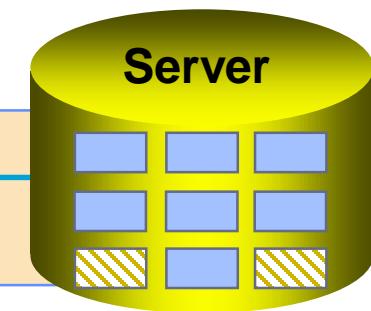
- En un ambiente multiusuario es necesario que se establezca una política de seguridad para cada aplicación.
- La base de datos proporciona elementos para controlar:
 - El acceso a la BD
 - Dar privilegios totales o parciales a los usuarios
 - Definir vistas y crear sinónimos para los objetos
 - Proteger los datos con encriptación

Control de Acceso de Usuarios

Administrador
de Base de
Datos

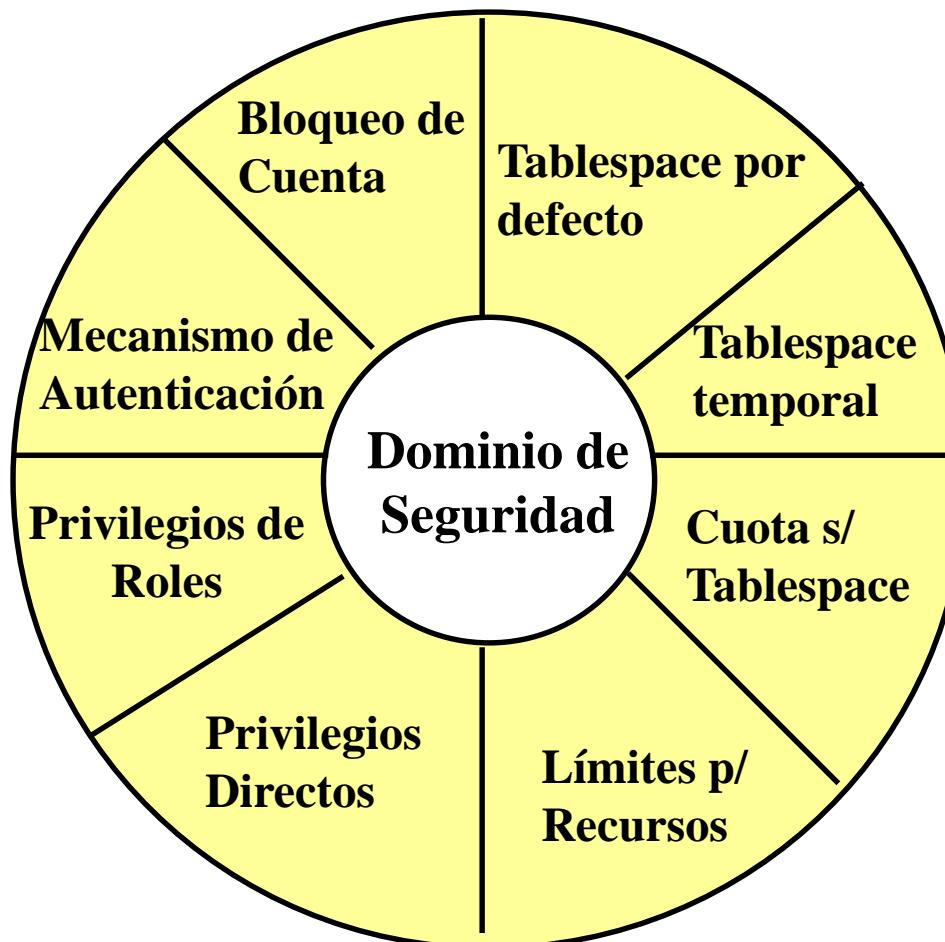


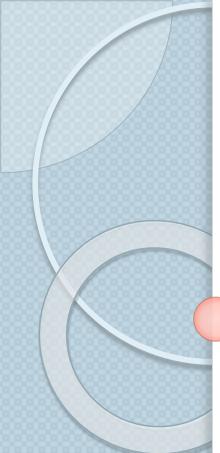
Usuario y Contraseña
Privilegios



Usuarios

USUARIOS Y SEGURIDAD





Cuando se crea un usuario se debe definir:

- El nombre del usuario
- El método de autenticación
- El tablespace por default
- El tablespace temporal
- Cuotas sobre los tablespaces
- El perfil

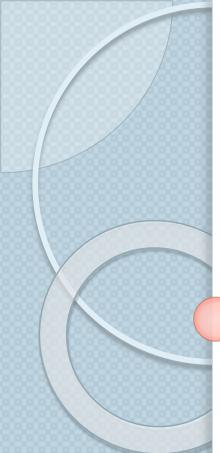
Mecanismo de Autenticación:

Un usuario que necesita acceder a la base de datos puede ser autenticado por la BD , el Sistema Operativo y por la red. El medio de autenticación es especificado en el momento de definición del usuario y puede ser modificado posteriormente.

En el curso sólo cubriremos el mecanismo de autenticación por la BD

Check List para crear un usuario

- Escoger un nombre de usuario
- Identificar el tablespace que requiere el usuario para sus objetos y definir cupos (quotas)
- Crear el usuario
- Asignar el tablespace temporal y por default (default y temporary)
- Otorgar (grant) privilegios y roles al usuario

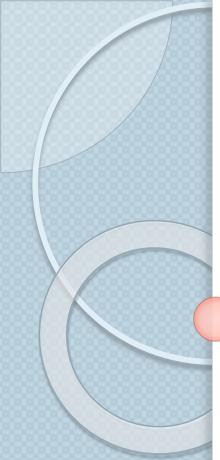


• ***Creando un Usuario: Sintaxis***

```
SQL> CREATE USER <usuario>
IDENTIFIED {BY <contraseña> | EXTERNALLY}
[ DEFAULT TABLESPACE <tablespace>]
[TEMPORARY TABLESPACE <tablespace>]
[QUOTA {tamaño K|M|UNLIMITED} ON TABLESPACE]
[PASSWORD EXPIRE]
[ACCOUNT {LOCK | UNLOCK}]
[PROFILE {<nombre_perfil>}|DEFAULT]
```

Creando un Usuario:Sintaxis

| | |
|-------------------------------------|---|
| BY <password> | Indica la contraseña que se requiere para acceder a la autenticación de la BD |
| EXTERNALLY | Especifica que el usuario será autenticado por el SO |
| DEFAULT/TEMPORARY TABLESPACE | Identifica el tablespace por defecto y el tablespace temporal |
| QUOTA | Define el máximo espacio permitido para los objetos del usuario en el tablespace definido (puede expresarse en entero (Bytes), K (KiloBytes) o M (MegaBytes). Si se pone UNLIMITED el usuario puede disponer de todo el espacio que pueda tener disponible en el Tablespace |
| PASSWORD EXPIRE | Obliga a que el usuario cambie su contraseña al ingresar por primera vez a través del SQL*PLUS |
| ACCOUNT LOCK/UNLOCK | Puede ser usado para bloquear explícitamente un usuario (por defecto el usuario está desbloqueado) |
| PROFILE | Utilizado para controlar el uso de recursos y especificar el mecanismo de control de password para los usuarios que adopten el profile. El perfil por defecto es DEFAULT |



Creando un Usuario: Ejemplo

```
SQL> CREATE USER JOSE  
          IDENTIFIED BY contraseña  
          DEFAULT TABLESPACE DATA  
          TEMPORARY TABLESPACE TEMP  
          QUOTA 10M      ON DATA  
          QUOTA 5M       ON TEMP  
          PASSWORD EXPIRE;
```

Creando un Usuario: Observaciones

Al crear a un usuario en la base de datos, se establece los medios por los que el Servidor de Base de Datos concede acceso al usuario. Usted puede asignarle estas propiedades opcionalmente al usuario:

- Tablespace predefinido
- Tablespace temporal
- cuotas por asignar espacio en tablespaces
- perfil que contiene límites del recurso

Cambiando su contraseña

- Al crear un usuario, debe crearse su contraseña
- Los usuarios pueden cambiar su contraseña usando la orden ALTER USER.

```
SQL> ALTER USER scott IDENTIFIED BY  
mipase;  
User altered.
```

Para borrar usuarios

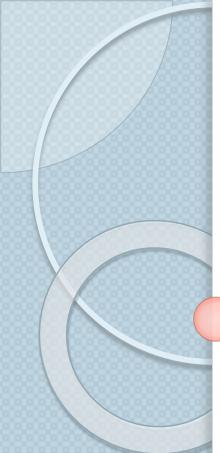
- **DROP USER <usuario> [CASCADE];**
- La opción CASCADE borra todos los objetos en el esquema antes de borrar el usuario.
- Sólo puede borrarse al usuario cuando está desconectado



Para verificar usuarios y su asignación de recursos

- **DBA_USERS**
- **DBA_TS_QUOTAS**

Administración de los Privilegios



Privilegios

Se llama PRIVILEGIO al derecho de ejecutar un tipo particular de sentencia SQL o de acceder a los objetos de otro usuario.

Dos tipos de privilegios

- **SYSTEM:** permite a los usuarios ejecutar determinadas acciones en la BD (crear, borrar, alterar tablas, vistas, etc.)
- **OBJECT:** permite acceder y manipular objetos específicos (seleccionar, insertar, ejecutar un determinado objeto)

Concediendo privilegios

- Se puede conceder privilegios y role a otros usuarios y roles usando la sentencia GRANT
- Para conceder un privilegio de sistema a otros usuarios (GRANT) se requieren los siguientes privilegios:
 - Para grantear un privilegio de sistema se debe tener el privilegio ADMIN OPTION o GRANT ANY PRIVILEGE
 - Para grantear un role de sistema se debe tener el privilegio ADMIN OPTION o GRANT ANY ROLE.
- Para que un usuario pueda conectarse DEBE tener el privilegio CREATE SESSION.

PRIVILEGIOS DEL SISTEMA

| | |
|-------------------|--|
| INDICES | [CREATE ALTER DROP] ANY INDEX |
| TABLAS | [CREATE ALTER DROP SELECT UPDATE DELETE] [ANY] TABLE |
| SESIÓN | CREATE SESSION ALTER SESSION RESTRICTED SESSION |
| TABLESPACE | [CREATE ALTER DROP] TABLESPACE UNLIMITED TABLESPACE |

SINTAXIS

CONCEDER PRIVILEGIOS

```
GRANT {privilegio_sistema | role}  
TO {usuario | role | PUBLIC}  
[,{usuario | role | PUBLIC}]  
[WITH ADMIN OPTION];
```

SACAR PRIVILEGIOS

```
REVOKE {privilegio_sistema | role}  
FROM {usuario | role | PUBLIC};
```

PRIVILEGIOS ESPECIALES

ANY: El usuario tiene privilegio sobre cualquier esquema. Ejemplo:

- CREATE ANY TABLE o CREATE TABLE

Ejemplos de concesión de privilegios del Sistema

```
GRANT CREATE SESSION TO usuario1;
```

```
GRANT CREATE TABLE TO usuario1;
```

```
GRANT CREATE USER TO usuario1;
```

```
GRANT DBA TO usuario1;
```

PRIVILEGIOS ESPECIALES SYSDBA y SYSOPER

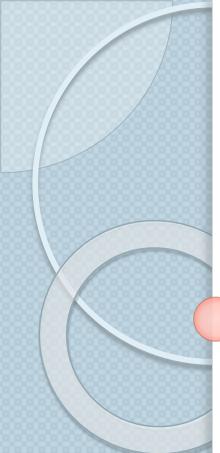
Son privilegios especiales que permite realizar operaciones de mantenimiento en la BD

PARA VER PRIVILEGIOS ASIGNADOS

- DBA_TAB_PRIVS
- DBA_COL_PRIVS

PARA EL USUARIO:

- SESSION_PRIVS
- USER_TAB_PRIVS



Privilegios sobre OBJETOS

Permiten realizar una acción particular sobre un objeto de un esquema específico.

Algunos objetos del esquema como los índices, triggers y links de base de datos, no tienen asociados privilegios de objetos, sino que su uso es controlado por privilegios del sistema.

Granteando privilegios al Objeto: Ejemplo

- Granteando Privilegios SELECT (Queries) en su tabla de b_empleados.

```
SQL> GRANT SELECT    ON b_empleados  
  2 TO usu1, usu2;
```

Granteando Privilegios UPDATE (de ACTUALIZACIÓN) en las columnas específicas a los usuarios y roles.

```
SQL> GRANT update (nombre_area, activa)  
  2 ON b_areas  
  3 TO scott, manager;
```

Privilegios de OBJETOS

| Privilegio | Table | View | Sequence | Procedure |
|------------|-------|------|----------|-----------|
| ALTER | X | | X | |
| DELETE | X | X | | |
| EXECUTE | | | | X |
| INDEX | X | | | |
| INSERT | X | X | | |
| REFERENCES | X | | | |
| SELECT | X | X | X | |
| UPDATE | X | X | | |

WITH GRANT OPTION

Da autoridad al usuario para grantear nuevamente los privilegios que le fueron concedidos.

```
SQL> GRANT select    ON b_employees  
  2 TO scott  
  3 WITH GRANT OPTION;
```

Grantear los privilegios SELECT a todos los usuarios en la secuencia de S_ORDEN_ID.

```
SQL> GRANT select  
  2 ON     s_orden_id  
  3 TO     PUBLIC;
```

Revocando privilegios de Objetos:Ejemplo

- Use la orden REVOKE para revocar privilegios concedidos a otros usuarios.
- Los privilegios concedidos a otros a través de WITH GRANT OPTION también se revocarán.

```
SQL> REVOKE select, insert  
  2  ON      p_departamento  
  3  FROM    Scott;
```

CREACIÓN DE ROLES O GRUPOS



Usuarios



Manager

Privilegios

Privilegios asignados
sin un rol

Privilegios asignados
con un rol

Uso de los roles

- La administración de los privilegios se hace más fácil usando roles. Un ROLE es una denominación para un grupo de privilegios relacionados que se asignan, como grupo, a usuarios u otros roles.
- Dentro de cada BD cada nombre de role debe ser único (no es contenido en un esquema específico). De allí que si un usuario se elimina (y todo su esquema), no tiene efecto en el role.

SINTAXIS

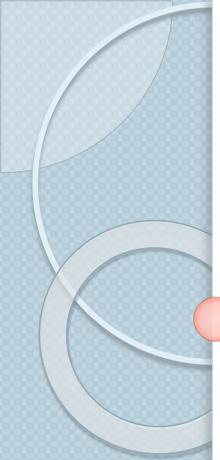
```
CREATE ROLE <nombre_rol>
[NOT IDENTIFIED | IDENTIFIED { BY
<password> | EXTERNALLY}]
```

ROLES PREDEFINIDOS

| | |
|----------------------------|---|
| CONNECT | Incluye los siguientes privilegios: ALTER SESSION, CREATE CLUSTER, CREATE DATABASE LINK, CREATE SEQUENCE, CREATE SESSION, CREATE SYNONYM, CREATE TABLE, CREATE VIEW |
| RESOURCE | Incluye los siguientes privilegios: CREATE CLUSTER, CREATE INDEXTYPE, CREATE OPERATOR, CREATE PROCEDURE, CREATE SEQUENCE, CREATE TABLE, CREATE TRIGGER, CREATE TYPE |
| DBA | Todos los privilegios tienen WITH ADMIN OPTION Típicamente: CREATE USER, DROP USER, DROP ANY TABLE, BACKUP ANY TABLE, SELECT ANY TABLE, CREATE ANY TABLE |
| EXP_FULL_DATABASE | Privilegios para exportar la Base de Datos |
| IMP_FULL_DATABASE | Privilegios para importar la Base de Datos |
| DELETE_CATALOG_ROLE | Borra privilegios sobre las tablas del Diccionario de Datos |
| SELECT_CATALOG_ROLE | Selecciona privilegios en las tablas del Diccionario de Datos |

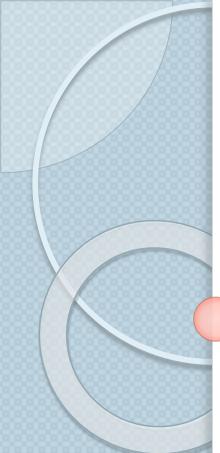
ADMINISTRACIÓN DE PERFILES

- Agrupan un conjunto de límites para los recursos y contraseñas del usuario
- Se pueden asignar a los usuarios en sentencias CREATE y ALTER
- Pueden ser habilitados o deshabilitados
- Pueden ser relacionados a un profile denominado DEFAULT



RECURSOS QUE PUEDEN CONTROLARSE CON EL PROFILE

- Tiempo de CPU
- Operaciones de I/O
- Tiempo ocioso/ Tiempo de conexión
- Sesiones concurrentes
- Edad y expiración de las contraseñas
- Bloqueo de cuenta
- otros



Creación de un profile -SINTAXIS

```
CREATE PROFILE <nombre_perfil>
LIMIT
[parámetros de recurso]
[parámetros de contraseña]
```

Creación de un profile –SINTAXIS

Parámetros de Recurso

```
CREATE PROFILE <nombre_perfil> LIMIT
[SESSIONS_PER_USER <cant_sesiones>|
UNLIMITED | DEFAULT]
[CPU_PER_SESSION <centésim_seg>| UNLIMITED |
DEFAULT]
[IDLE_TIME <minutos>| UNLIMITED | DEFAULT]
[CONNECT_TIME <minutos>]
[LOGICAL_READ_PER_SESSION <bloques> |
UNLIMITED | DEFAULT]
[LOGICAL_READ_PER_CALLS <bloques> |
UNLIMITED | DEFAULT]
[COMPOSITE_LIMIT <Unidades de servicio> |
UNLIMITED | DEFAULT]
PRIVATE_SGA <size_clause> | UNLIMITED | DEFAULT]
```

Parametros de Recursos

SESSIONS_PER_USER: Especifica el número de sesiones simultáneas a las que desea limitar el usuario.

CPU_PER_SESSION: Especifica el límite de tiempo de CPU para una sesión, expresado en centésimas de segundos.

CPU_PER_CALL: Especifica el límite de tiempo de la CPU para una llamada (un parse, execute o fetch), expresado en centésimas de segundos.

CONNECT_TIME: Especifique el tiempo total transcurrido para una sesión, expresado en minutos.

IDLE_TIME: Especifica los períodos permitidos de tiempo de inactividad durante una sesión, expresados en minutos. Las consultas de larga duración y otras operaciones no están sujetas a este límite.

LOGICAL_READS_PER_SESSION: Especifica el número permitido de bloques de datos leídos en una sesión, incluidos los bloques leídos de la memoria y del disco.

LOGICAL_READS_PER_CALL: Especifica el número permitido de bloques de datos leídos para una llamada para procesar una sentencia SQL (analizar, ejecutar o recuperar).

Parametros de Recursos

PRIVATE_SGA: Especifica la cantidad de espacio privado que una sesión puede asignar en el conjunto compartido del área global del sistema (SGA). Consulte la cláusula size_clause para obtener información sobre esa cláusula.

COMPOSITE_LIMIT: Especifica el costo total del recurso para una sesión, expresado en unidades de servicio. Oracle Database calcula el total de unidades de servicio como una suma ponderada de CPU_PER_SESSION, CONNECT_TIME, LOGICAL_READS_PER_SESSION y PRIVATE_SGA.

Creación de un profile –SINTAXIS

Parámetros de Contraseña

```
CREATE PROFILE <nombre> LIMIT
[FAILED_LOGIN_ATTEMPTS <cant_veces> | UNLIMITED|
DEFAULT]
[PASSWORD_LIFE_TIME <cant_días> | UNLIMITED| DEFAULT]
[PASSWORD_GRACE_TIME <cant_días> | UNLIMITED| DEFAULT]
[PASSWORD_REUSE_TIME <cant_días> | UNLIMITED| DEFAULT]
[PASSWORD_REUSE_MAX <cant_veces> | UNLIMITED | DEFAULT]
[PASSWORD_LOCK_TIME <cant_días> | UNLIMITED| DEFAULT]
[PASSWORD_VERIFY_FUNCTION <nombre_funcion>]
```

La función debe ser creada en el esquema de SYS

Parámetros de Contraseñas

FAILED_LOGIN_ATTEMPTS: Especifica el número de intentos fallidos de iniciar sesión en la cuenta de usuario antes de bloquear la cuenta.

PASSWORD_LIFE_TIME: Especifica el número de días en que se puede utilizar la misma contraseña para la autenticación. Si también establece un valor para **PASSWORD_GRACE_TIME**, la contraseña caduca si no se cambia dentro del período de gracia y se rechazan otras conexiones. Si no establece un valor para **PASSWORD_GRACE_TIME**, su valor predeterminado de **UNLIMITED** provocará que la base de datos emita una advertencia pero permita que el usuario continúe conectándose indefinidamente.

PASSWORD_REUSE_TIME y **PASSWORD_REUSE_MAX**:

Estos dos parámetros deben ajustarse conjuntamente entre sí.

PASSWORD_REUSE_TIME especifica el número de días antes de que una contraseña no pueda ser reutilizada. **PASSWORD_REUSE_MAX** especifica el número de cambios de contraseña necesarios antes de que se pueda volver a usar la contraseña actual. Para que estos parámetros tengan algún efecto, debe especificar un entero para ambos.

Parámetros de Contraseñas

PASSWORD_LOCK_TIME: Especifica el número de días que una cuenta se bloqueará después del número especificado de intentos de inicio de sesión fallidos consecutivos.

PASSWORD_GRACE_TIME: Especifica el número de días después de que comience el período de gracia durante el cual se emite una advertencia y se permite el inicio de sesión. Si la contraseña no se cambia durante el período de gracia, la contraseña caduca.

PASSWORD_VERIFY_FUNCTION: permite que una secuencia de comandos de verificación de complejidad de contraseña PL / SQL se pase como un argumento a la sentencia CREATE PROFILE. Oracle Database proporciona un script predeterminado, pero puede crear su propia rutina o utilizar software de terceros en su lugar.

Creación de un profile -Ejemplo

```
CREATE PROFILE desarrollo LIMIT  
SESSIONS_PER_USER 2  
CPU_PER_SESSION 1000  
IDLE_TIME 60  
CONNECT_TIME 480;
```

Asignación del perfil a un usuario

- CREATE USER usu1 IDENTIFIED BY usu1 PROFILE desarrollo;
- ALTER USER usu1 PROFILE desarrollo;

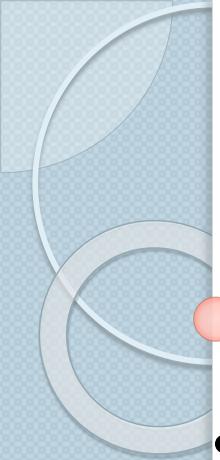
Borrar un perfil

`DROP PROFILE <nombre_perfil> [CASCADE];`

La opción CASCADE desasigna el perfil de los usuarios y les asigna el perfil DEFAULT.

Para consultar el perfil del usuario

- DBA_USERS
- DBA_PROFILES



Sinónimos

- Simplifica acceso a los objetos creando un sinónimo (otro nombre) para un objeto. Es particularmente útil cuando el objeto está fuera del esquema o fuera de la BD
- Abrevia nombres de los objetos largos

```
SQL> CREATE SYNONYM b_personas  
2 FOR alice.b_personas;
```

Creando un Sinónimo: Ejemplos

- Cree un nombre abreviado para la vista de DEPT_SUM_VU.

```
SQL> CREATE SYNONYM d_sum  
2 FOR dept_sum_vu;
```

Los sinónimos públicos pueden ser creados y eliminados por el DBA

```
SQL> CREATE PUBLIC SYNONYM b_localidad  
2 FOR alice.b_localidad;
```

Información de Sinónimos

```
DESCRIBE user_synonyms
```

| Name | Null? | Type |
|--------------|----------|---------------|
| SYNONYM_NAME | NOT NULL | VARCHAR2(30) |
| TABLE_OWNER | | VARCHAR2(30) |
| TABLE_NAME | NOT NULL | VARCHAR2(30) |
| DB_LINK | | VARCHAR2(128) |

```
SELECT *
FROM user_synonyms;
```

| SYNONYM_NAME | TABLE_OWNER | TABLE_NAME | DB_LINK |
|--------------|-------------|------------|---------|
| EMP | ORA1 | EMPLOYEES | |