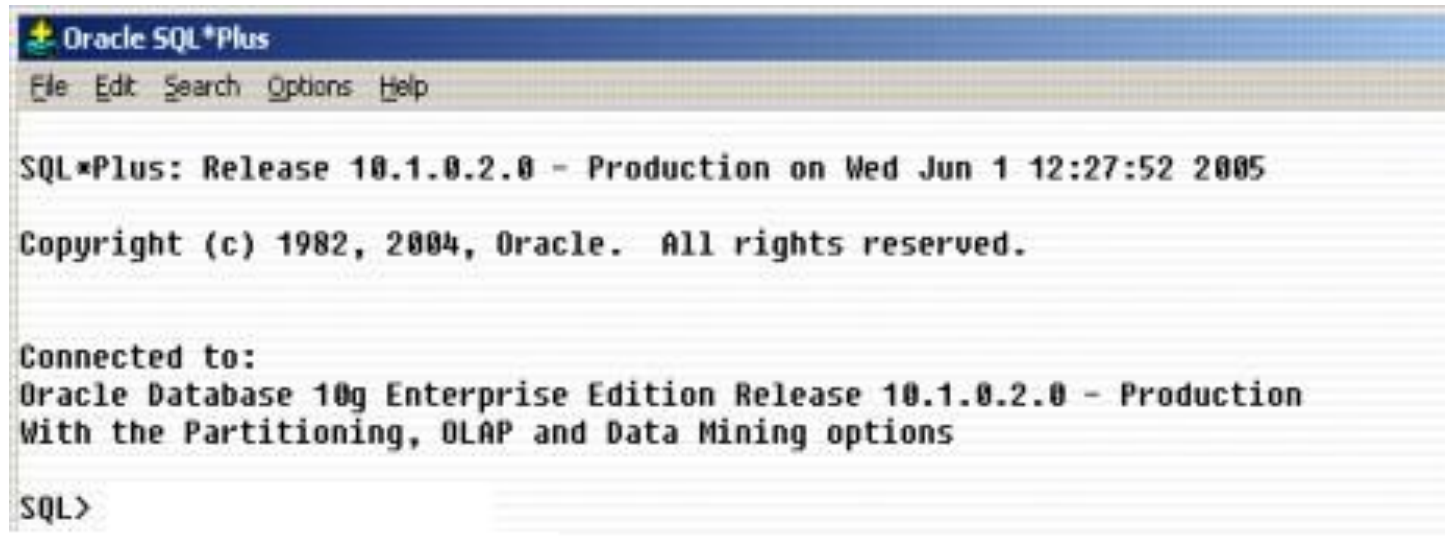




# BASE DE DATOS II

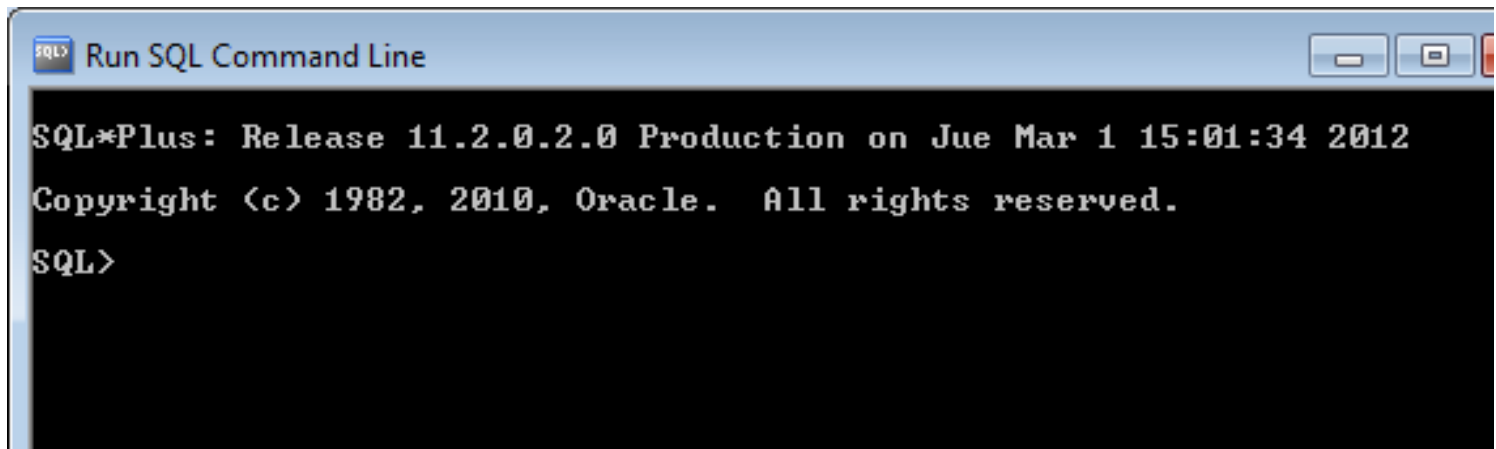
## SQL\*Plus

# SQL\*Plus



A screenshot of the Oracle SQL\*Plus command-line interface. The window has a title bar that says "Oracle SQL\*Plus" and a menu bar with "File", "Edit", "Search", "Options", and "Help". The main text area shows the following output:

```
SQL*Plus: Release 10.1.0.2.0 - Production on Wed Jun 1 12:27:52 2005  
Copyright (c) 1982, 2004, Oracle. All rights reserved.  
  
Connected to:  
Oracle Database 10g Enterprise Edition Release 10.1.0.2.0 - Production  
With the Partitioning, OLAP and Data Mining options  
  
SQL>
```



A screenshot of a window titled "Run SQL Command Line". The window has a standard Windows-style title bar with minimize, maximize, and close buttons. The main text area shows the following output:

```
SQL*Plus: Release 11.2.0.2.0 Production on Jue Mar 1 15:01:34 2012  
Copyright (c) 1982, 2010, Oracle. All rights reserved.  
  
SQL>
```

# Características de SQL\*Plus

- Interfaz de usuario interactiva para el SGBD Oracle

Permite:

- Escritura de sentencias de manipulación y definición de datos
- Depuración de dichas sentencias
- Visualización de los resultados en pantalla (o su volcado a un fichero de texto)
  - Es posible dar formato al resultado de una consulta
  - Y consultar la ayuda en línea

# Conexión con la B.D. desde SQL\*Plus

- Autenticación del usuario: **usuario** y **contraseña**
- Identificación de la Base de Datos mediante la **cadena de host** (cadena de conexión) que incluye los nombres del equipo *servidor*, *protocolo* de comunicación y *base de datos*
  - Para la conexión en el laboratorio, utilizaremos la cadena de conexión '**ORCL**'
  - Si se trabaja con una base de datos local (*Personal Oracle*), no se necesita cadena de host (se deja en blanco)

# Conexión con la B.D. desde SQL\*Plus

- Formas de **conexión** a una cuenta de la base de datos:
  - - Al *iniciar* SQL\*Plus (ventana 'Conectar'),
  - - Desde *dentro* de SQL\*Plus: connect <cuenta>/<contraseña>@<cadena-host>
  - Ejemplo:  
Connect baseda001/baseda001@ORCL

# Sentencias SQL en SQL\*Plus

El *prompt* tiene la apariencia siguiente: **SQL>**

- Una sentencia SQL puede constar de **varias líneas** para mayor legibilidad
- Un ENTER **no** acaba la sentencia, sino que inicia una nueva línea en la misma
- Las sentencias SQL terminan con un ';' (si tras él se pulsa ENTER, la sentencia se ejecutará)
- El sistema almacena la consulta en un **búfer**, en un fichero de texto llamado **afiedt.buf**
- El búfer siempre **contiene la última sentencia SQL ejecutada**

# Comandos de Archivo SQL\*Plus

- SAVE archivo
- GET archivo
- START archivo
- @ archivo
- EDIT archivo
- SPOOL archivo
- EXIT

# Comandos del SQL\*Plus

- No son sentencias SQL, por eso **no** necesitan finalizar con ‘;’
- DESC[RIBE] <nombre\_tabla>
  - DESCRIBE, o abreviado DESC, muestra información sobre las columnas de la tabla denominada <nombre\_tabla>, es decir, tipo de datos, longitud, si admite el nulo...



# Comandos del SQL\*Plus

- EDIT : Invoca el editor de textos (por defecto el Bloc de Notas, *Notepad*) con el contenido del búfer y permite su modificación
- L[IST] : muestra el contenido del búfer
- R[UN] : Muestra y ejecuta el contenido del búfer.
- / : Ejecuta el contenido del búfer. A diferencia de RUN, no muestra previamente la sentencia
- EDIT <nombre\_fichero.extensión> Abre el editor *Notepad* con el contenido del fichero indicado

# Comandos del SQL\*Plus

- **SAVE** <nombre\_fichero.extensión>[REP[LACE]]  
Guarda el contenido del búfer en el fichero indicado, Si no se indica extensión, por defecto se toma *'sql'*  
Si el fichero ya existe y se desea sobrescribirlo es necesario incluir REPLACE (o abreviado REP)
- **STA[RT]** <nombre\_fichero.extensión> Ejecuta el contenido del fichero indicado (debe contener sentencias SQL) . Si no se indica extensión, asume *'sql'*
- **GET** <nombre\_fichero.extensión > Carga en el búfer el contenido del fichero indicado (pero no lo ejecuta)

# Comandos del SQL\*Plus

- SPO[OL] <nombre\_fichero.extensión>... SPOOL OFF

Crea un fichero con el nombre indicado en el que, a partir de ese momento, se irá almacenando todo lo que vaya apareciendo por pantalla (**eco**): sentencias, resultados, mensajes de error...

- Si no se indica extensión para el fichero, por defecto se toma `‘.lst’`
  - El eco finaliza cuando se escribe SPOOL OFF
- EXIT (o QUIT) Finaliza la sesión de trabajo y cierra el entorno SQL\*Plus
- Cerrar la ventana de SQL\*Plus sin haber ejecutado EXIT o QUIT finaliza la sesión de igual forma que si hubiera ocurrido un error

# Desplegando la estructura de la Tabla

- En SQL\*Plus el comando DESCRIBE despliega la estructura de una tabla (nombre de columna, valores NOT NULL, y tipo de datos)

```
SQL> DESCRIBE b_areas
```

- NULLABLE especifica si la columna admite Nulos

Name	Type	Nullable	Default	Comments
-----	-----	-----	-----	-----
ID	NUMBER (6)			
NOMBRE_AREA	VARCHAR2 (40)			
FECHA_CREA	DATE			
ACTIVA	VARCHAR2 (1)		'S'	
ID_AREA_SUPERIOR	NUMBER (6)	Y		

# SAVE Ejemplo

- Para grabar lo retenido en el buffer se utiliza el comando SAVE archivo

```
SQL> SELECT DISTINCT barrio  
      2  from b_empleados;  
SQL > SAVE p2
```

- De esta forma la sentencia seleccionada se almacena en el archivo p2

# START Ejemplo

- Para ejecutar el archivo p2 se utiliza el comando START archivo

```
SQL> START p2  
Barrio  
.....
```

- La sentencia automáticamente es ejecutada
- La barra (/) ejecuta la última sentencia que se encuentra en el búffer

# SPOOL Ejemplo

- Permite generar un archivo de texto con los datos desplegados. Se graba la información al utilizar el comando SPOOL OFF.

```
SQL> spool barrios.lst
Started spooling to C:\developer\barrios.lst
SQL> SELECT DISTINCT barrio from b_empleados;
BARRIO
-----
Jara
San Lorenzo
Carmelitas
3 rows selected
SQL> spool off
Stopped spooling to C:\developer\barrios.lst
```



# Comandos SQL\*Plus de formato de visualización de los datos

- SHOW ALL : muestra por pantalla el valor actual de las variables del entorno SQL\*Plus
- SET PAGES[IZE] <n° de filas> : N° de líneas (del resultado de una consulta) mostradas por pantalla antes de volver a imprimir los nombres de las columnas de nuevo. Por defecto suele ser 24.
- SET PAU[SE] ON y SET PAU[SE] OFF : Para el *scroll* cuando alcanza el PAGESIZE activo, y espera un ENTER para continuar (para mostrar las siguientes PAGESIZE filas). Por defecto está OFF
- SET LIN[ESIZE] <n°> Establece la máxima longitud de una línea en pantalla. Por defecto suele ser 100



# Comando COLUMN

- COLUMN <nombre\_columna> FORMAT <formato>
- COLUMN <nombre\_columna> FORMAT An
  - Visualiza una columna de tipo char o varchar2 con una anchura de *n* caracteres alfanuméricos
- COLUMN <nombre\_columna> FORMAT 999,999.999
  - Visualiza una columna de tipo number con el formato indicado, donde cada 9 significa 'cualquier dígito del 0 al 9'.
- COLUMN <nombre\_columna> DEFAULT
  - Establece el formato por defecto para la columna indicada
- COLUMN <nombre\_columna> CLEAR
  - Elimina el formato para la columna indicada
- COLUMN <nombre\_columna>
  - Muestra el formato actual establecido para la columna indicada

# Comando COLUMN : Sintaxis

- CLE[AR] – Limpia formato de columna.
- HEA[DING] text – Setea la columna con un texto.
- JUS[TIFY] {align} – Alinea la columna seteada a la izquierda, centro o derecha.

```
COL[UMN] [{column|alias} [option]]
```

# Comando COLUMN : Ejemplos

- Seteando columnas con heading

```
COLUMN apellido HEADING 'Nombre| Empleado' FORMAT A15  
COLUMN asignacion JUSTIFY LEFT FORMAT $99,990.00  
COLUMN fecha_ing FORMAT A10 NULL 'No contratado'
```

- Desplegando el seteado para la columna APELLIDO

```
COLUMN apellido
```

- Limpiar seteos para la columna APELLIDO

```
COLUMN apellido CLEAR
```

# Modelos de formatos COLUMN

- **Tipo de datos date y character**
  - **An**            Setea el despliegue con n. Ej.A15
- **Number columns**
  - **9**            Cero. Supresión de dígitos
  - **0**            Fuerza el cero
  - **\$**            Designa el signo de dolar
  - **.**            Posición punto decimal
  - **,**            Separador de miles

# SQL\*Plus: Commandos de Edición

- A[PPEND] text
- C[HANGE] / old/ new
- CL[EAR] BUFF[ER]
- DEL
- I[NPUT] text
- L[IST] n
- n text

# SQL\*Plus: Comandos de Edición

- **append (a)** : Añade texto al final de la línea:

SQL> a and and cdclasnm = '02' genera:

1 select distinct(cddni) from bperret

2 where cdhabil = 'DEM' and cdclasnm = '02'

- **change (c)** : Cambia una expresión por otra  
(c/antigua/nuevo) - SQL> c /02/03

3\* and cdclasnm = '03'

Sino ponemos nada en nuevo, nos eliminará esa expresión.

- **clear buffer** (cl buff) Limpia el buffer

# SQL\*Plus: Commandos de Edición

- **del** Para borrar líneas. Tiene varias opciones:
- **del** Borra la línea actual
- **del n** Borra la línea número n
- **del \*** Borra la línea actual
- **del n \*** Borra desde la línea n hasta la actual
- **del last** Borra la última línea
- **del m n** Borra las líneas situadas entre ellas.
- **del \* n** Borra la actual línea hasta la línea n.
- **input (i)** Inserta el texto que quieras

# SQL\*Plus: Commandos de Edición

- **list (l)** Muestra las líneas que hay en el buffer. Puede tener las siguientes combinaciones:
  - **list n** Muestra la línea n
  - **list \*** Muestra la línea actual.
  - **list n \*** Muestra las líneas entre la actual y el final
  - **list last** Muestra la última línea
  - **list m n** Muestra las líneas entre m y n



# Variables de sustitución

- Se puede usar el símbolo (&) como variable de sustitución para:
  - Condiciones WHERE
  - Cláusulas ORDER BY
  - Expresiones de columnas
  - Nombres de tablas
  - Sentencias SELECT completas
- Cuando se ejecuta, la sentencia solicita interactivamente el ingreso de los valores

# Ejemplo de variables de sustitución

- Para strings y fechas, ponga la variable entre apóstrofes
- Use (&&) si se desea reutilizar el valor de la variable sin tener que repetir el valor por cada ejecución

```
SQL> SELECT nombre || ' ' || apellido as Empleado  
2   FROM b_empleados  
3   WHERE cedula = &c;
```

```
SQL> SELECT nombre || ' ' || apellido as Empleado  
2   FROM b_empleados  
3   WHERE apellido = '&apellido';
```

# Variables de Usuario

- **DEFINE** : Crea una variable de usuario de tipo CHAR.
- **ACCEPT** : Acepta un valor introducido por el usuario y lo almacena en una variable, además de permitir introducir literales informativos así como definir el tipo de variable.
- **UNDEFINE** : elimina la definición de una variable

# Uso de DEFINE para crear variables

```
SQL> DEFINE ced_empleado = 200
SQL> SELECT cedula, apellido
      2  FROM b_empleados
      3  WHERE cedula = &ced_empleado;
SQL> UNDEFINE ced_empleado
```

- **SHOW ALL** muestra la lista completa de todas las variables definidas

- 
- A practicar