

Apellidos: _____
Nombre: _____
Fecha: _____ Usuario: _____

La última versión de este documento se puede descargar de <https://alvarogonzalezsotillo.github.io/apuntes-clase/sistemas-gestores-bbdd-asir2/apuntes/3/asgbd-03-trabajo-scripts-2022.pdf>

1. *Scripts* de inicio y parada de Oracle (1 punto)

Crea dos *scripts* para iniciar y parar **Oracle**

- /home/alumno/scripts/oraclestart.sh
- /home/alumno/scripts/oraclestop.sh

2. Arrancar automáticamente Oracle cuando se inicie el servidor (2 puntos)

- **Oracle** debe levantarse cuando la máquina se inicie, y apagarse cuando la máquina se cierre.
- Oracle se iniciará solo si se indica en el fichero /etc/oratab.
- En el fichero /home/alumno/logs/oracle.log se dejará una traza de cuando se arrancó y se paró la máquina, y si fue necesario arrancar o parar **Oracle**. Por ejemplo:

```
2022-02-10-12:40:00 - Solicitud de arrancar Oracle
2022-02-10-12:40:01 - Oracle arrancando porque /etc/oratab indica Y
2022-02-10-12:40:20 - Oracle arrancado
```

Listado 1: Ejemplo de /home/alumno/logs/oracle.log cuando **Oracle** se arranca

```
2022-02-10-12:41:00 - Solicitud de parar Oracle
2022-02-10-12:41:20 - Oracle parado
```

Listado 2: Ejemplo de /home/alumno/logs/oracle.log cuando **Oracle** se para

```
2022-02-10-12:40:00 - Solicitud de arrancar Oracle
2022-02-10-12:40:01 - Oracle no se arranca porque /etc/oratab indica N
```

Listado 3: Ejemplo de /home/alumno/logs/oracle.log cuando **Oracle** no se arranca

Aviso

- Los *scripts* no cambian el fichero `/etc/oratab`, solo lo consultan.
- Asegúrate de que el *script* no lee líneas de comentarios, o de otros SID

3. Crea usuarios de base de datos (2 puntos)

- Crea un script de nombre `/home/alumno/scripts/nuevos-usuarios-oracle.sh` que lea de su entrada estándar
- Cada línea será un nombre de usuario a crear
- Cada usuario se creará con permisos `connect` y `resource`
- Si el usuario ya existe, no se hará nada con él

```
echo pepe juan manolo | ./nuevos-usuarios-oracle.sh
pepe: no existe, se crea con contraseña pepe y se le dan roles
juan: ya existe, no se hace nada
manolo: no existe, se crea con contraseña manolo y se le dan roles
```

Listado 4: Ejemplo de uso

4. Borra un usuario de la base de datos (2 puntos)

- Crea un script de nombre `/home/alumno/scripts/quita-usuario-oracle.sh` que bloquee el usuario pasado como parámetro
 - Si el usuario no existe, informará de ello y tendrá un `errorlevel` de 1
 - Si se invoca con primer parámetro `--drop` y segundo el usuario, borrará el usuario.
 - Si se invoca sin parámetros, o incorrectos, mostrará el texto de ayuda del listado 5

```
Borra o bloquea un usuario de oracle.

Uso:  quita-usuario-oracle.sh <usuario>
      quita-usuario-oracle.sh --drop <usuario>
```

Listado 5: Ayuda del script `quita-usuario-oracle.sh`**Aviso**

En la salida del *script* debe quedar claro si el usuario se borra o bloquea

5. Almacena información periódicamente en la base de datos (3 puntos)

Programa un *script* para que cada minuto almacene en la tabla ESTADO la siguiente información

- uptime: Tiempo activo del servidor (`uptime -p`)
- procesos: Número de procesos activos en el servidor
- disco: Número de KB usados en el punto de montaje raíz
- conexiones: Número de conexiones TCP abiertas

```
create table ESTADO(  
  uptime varchar(40),  
  procesos varchar(40),  
  disco varchar(40),  
  conexiones varchar(40)  
);
```

Listado 6: Creación de la tabla ESTADO

Aviso

Pistas para realizar el *script*:

- Los *heredocs* pueden **contener variables**
- **Cortar columnas** con `awk`
- **Quitar las primeras líneas** de la salida con el comando `tail`

6. Envía un correo periódicamente (2 puntos)

- Programa un *script* para que cada minuto envíe un correo con la información de la tabla ESTADO.
- El correo se enviará a `alvarogonzalez.profesor@gmail.com`
- Con copia a `alvaro@alvarogonzalez.no-ip.biz`
- Indica tu nombre en el asunto del correo
- El fichero tendrá **formato CSV**. Se debe poder abrir directamente con **excel** y visualizar su resultado en filas y columnas.

7. Instrucciones de entrega

- El ejercicio se realizará y entregará de manera individual. Solo se puede hacer por parejas si no hay ordenadores suficientes.
- El profesor comprobará el funcionamiento del sistema el día indicado.
- Sube en la tarea del aula virtual un ZIP con todos los ficheros que has creado o modificado:
 - *Scripts*
 - *units* de systemd
 - Ficheros de cron / anacron