

# Consultas del AR

En esta práctica se aplicará el lenguaje formal del modelo relacional que se ha estudiado: el álgebra relacional (AR). Para ello se usará la herramienta educativa DES ([des.sourceforge.net](http://des.sourceforge.net)) en versión de escritorio (Java o terminal del sistema operativo) o más recomendable en su versión on-line en [desweb.fdi.ucm.es](http://desweb.fdi.ucm.es). Alternativamente se puede usar también WinRDBI, que se puede consultar en el apéndice de "Understanding Relational Database Query Languages" (Suzanne W. Dietrich, Arizona State University, 2001, ISBN 0-13-028652-4, y que se puede encontrar en la biblioteca de la facultad).

## 1. Primeros pasos

Si se decide usar la aplicación de escritorio, en primer lugar habría que instalar el sistema DES desde la dirección [http://des.sourceforge.net/](http://des.sourceforge.net) según se indican en las instrucciones del sitio web. Es necesario tener la versión 1.8 de Java, no funciona con versiones posteriores. Puedes consultar también el [manual del sistema](#) para obtener más información. No obstante, para los primeros pasos es mejor usar la aplicación on-line [desweb.fdi.ucm.es](http://desweb.fdi.ucm.es) y los documentos de ayuda que encontrarás en el panel del directorio y en el icono de ayuda.

En segundo lugar se crearán las tablas para una base de datos de una empresa de servicios informáticos con el esquema y los datos que se muestran más abajo. Como ayuda, se proporciona el archivo **datos.ra** con casi todas las instrucciones necesarias. En este archivo se indica qué es lo que falta y debes completar. Por lo tanto, es lo primero que debes hacer.

- **programadores(dni string, nombre string, dirección string, teléfono string)**

<b>dni</b>	<b>nombre</b>	<b>dirección</b>	<b>teléfono</b>
1	Jacinto	Jazmín 4	91-8888888
2	Herminia	Rosa 4	91-7777777
3	Calixto	Clavel 3	91-1231231
4	Teodora	Petunia 3	91-6666666

Ejemplo:

```
create table programadores(dni string primary key, nombre string,  
dirección string, teléfono string);
```

Esta definición de tabla en SQL es equivalente a los dos asertos siguientes en un archivo de Datalog (con extensión `.dl`):

```
:-type(programadores(dni:string, nombre:string, dirección:string,  
teléfono:string)).
```

```
:-pk(programadores,[dni]).
```

Para insertar una tupla en un archivo con extensión `.ra` o `.sql`:

```
insert into programadores values('1','Jacinto','Jazmín 4','91-8888888');
```

O bien:

```
/assert programadores('1','Jacinto','Jazmín 4','91-8888888')
```

En el caso de un archivo de Datalog, se incluiría simplemente:

```
programadores('1','Jacinto','Jazmín 4','91-8888888').
```

Las tablas deben quedar como se muestra a continuación:

- `analistas(dni string, nombre string, dirección string, teléfono string)`

<u>dni</u>	nombre	dirección	teléfono
4	Teodora	Petunia 3	91-6666666
5	Evaristo	Luna 1	91-1111111
6	Luciana	Júpiter 2	91-8888888
7	Nicodemo	Plutón 3	NULL

- `distribución(códigopr string, dniemp string, horas int)`

<u>códigopr</u>	<u>dniemp</u>	horas
P1	1	10
P1	2	40
P1	4	5
P2	4	10
P3	1	10
P3	3	40
P3	4	5
P3	5	30
P4	4	20
P4	5	10

- `proyectos(código string, descripción string, dnidir string)`

<u>código</u>	descripción	dnidir
P1	Nómina	4
P2	Contabilidad	4
P3	Producción	5
P4	Clientes	5
P5	Ventas	6

## 2. Ejemplos

Una vez que hayas completado y procesado el archivo datos.ra puedes practicar con algunas consultas de ejemplo, que no es necesario que entregues.

1. Códigos de proyectos en los que trabaja el empleado con DNI 4.

- AR:  $\pi_{\text{códigopr}}(\sigma_{\text{dniemp}=4}(\text{distribución}))$

```
project códigopr (select dniemp='4' (distribución));
```

El lenguaje textual usado para expresar la consulta del álgebra relacional permitir asociar casi literalmente esta solución con su expresión formal. No obstante, las cadenas se delimitan con comillas simples, y todos los identificadores de tablas y columnas comienzan en minúscula (si fuese necesario usar un identificador que no comenzase por letra minúscula, se puede encerrar el identificador entre comillas simples). Además, las consultas deben terminar en punto y coma (solo es estrictamente necesario si el modo **multiline** está activado).

2. Todos los empleados de la empresa.

- AR:  $\text{programadores} \cup \text{analistas}$

```
programadores union analistas;
```

3. Nombres de los directores de proyecto que sean analistas.

- AR:  $\pi_{\text{nombre}}(\sigma_{\text{dnidir}=\text{dni}}(\text{proyectos} \times \text{analistas}))$

```
project nombre (select dni=dnidir (proyectos product  
analistas));
```

4. Nombres, direcciones y teléfonos de los empleados que dirigen algún proyecto.  
Para resolver esta consulta vamos a usar vistas, nuevas relaciones que asociamos a consultas del álgebra relacional. Para definir una vista se usa el operador **:=**, como se ve por ejemplo en el primer paso para resolver esta consulta.

- 1 Datos de todos los empleados

```
empleados ← programadores ∪ analistas  
empleados := programadores union analistas;
```

- 2 Hacemos el producto cartesiano con los DNI de los directores de proyecto:

```
dnidirporeempleados ←  $\pi_{\text{DNIDir}}(\text{proyectos}) \times \text{empleados}$   
dnidirporeempleados :=  
(project dnidir (proyectos)) product empleados;
```

- 3 Nos quedamos con los datos de los directores de proyecto

```
DatosDirProyecto ←  
  
 $\pi_{\text{nombre, dirección, teléfono}}(\sigma_{\text{dnidir}=\text{dni}}(\text{dnidirporeempleados}))$   
datosdirproyecto :=  
project nombre, dirección, teléfono (select dnidir =  
dni (dnidirporeempleados));
```

### 3. Instrucciones para la elaboración de la práctica

1. Puedes incluir comentarios: una línea que empiece por "--", o bien varias delimitadas entre "\\*" y "\\*/".
2. El resultado de esta práctica debe ser un fichero de texto **práctica\_ar.ra** que se debe subir al Campus Virtual cuando se termine la práctica.
3. Este fichero debe incluir al principio los comandos:

**/abolish** (borrado de la base de datos previa, si hubiera)

**/multiline on** (acordaos de terminar las consultas con ";")

seguidos de *todas* las instrucciones de creación de tabla y de inserción de datos.

4. Prueba **una a una** las consultas que se piden en esta práctica en el *prompt* de DES. Cuando compruebes que su resultado es correcto, añádela al fichero de texto **práctica\_ar.ra**.
5. Añade al final del archivo las instrucciones de selección necesarias para comprobar el buen funcionamiento de las vistas:

```
select true (vista1);
```

```
select true (vista2);
```


```
select true (vista3);
```

```
select true (vista4);
```

```
select true (vista5);
```

```
select true (vista6);
```

```
select true (vista7);
```

6. Ejecuta el archivo para comprobar que se reproduzca correctamente la ejecución de todas las consultas. En DESweb simplemente se pulsaría el botón Run. Si usas la aplicación Java, pulsa el botón Play . Si usas el terminal de DES en una ventana del sistema operativo, escribe el comando **/process práctica\_ar.ra**. Este comando procesa el archivo en la carpeta desde la que se arranque DES. Si el archivo estuviese en otra carpeta debes indicar la ruta completa o relativa, como por ejemplo **/process z:\hlocal\práctica\_ar.ra**. **Nota:** Algunos comandos que pueden resultar útiles son **/pwd**, que informa la carpeta actual, y **/cd Carpeta**, que cambia a la carpeta indicada. No obstante, esto no será necesario en la aplicación on-line.
7. Sube este archivo cuando esté completo al Campus Virtual.

## 4. Ejercicios a incluir en práctica\_ar.ra

Hay que programar en **álgebra relacional** (sin usar operaciones de la extendida) las siguientes consultas. Se puede intentar resolver el mismo problema con distintas alternativas semánticamente equivalentes. Denomina a la vista de cada apartado como **vista $N$** , siendo  $N$  el número del apartado (recuerda usar el operador de definición de vistas " $\leftarrow$ ", que en su versión textual es ":=").

En lo que sigue, "*asignado*" significa que "*aparezca asignado en la tabla **distribución***".

**IMPORTANTE:** No cambies las tablas que se proporcionan, ya que se usará corrección automática. Si usas otros identificadores, la solución no sería válida.

1. DNI de todos los empleados. Esquema: **vista1(dni)**.
2. DNI de los empleados que son a la vez programadores y analistas. Esquema: **vista2(dni)**.
3. DNI de los empleados sin trabajo (ni están asignados a proyectos ni son directores de ellos). Esquema: **vista3(dni)**.
4. Código de los proyectos sin analistas asignados (en la relación **distribución**) Esquema: **vista4(código)**.
5. DNI de los analistas que dirijan proyectos pero que no sean programadores. Esquema: **vista5(dni)**.
6. Descripción de los proyectos con los nombres de los programadores y horas asignados a ellos. Esquema: **vista6(descripción,nombre,horas)**.
7. Listado de teléfonos compartidos por empleados (solo hay que indicar el número de teléfono). Esquema: **vista7(teléfono)**.