

BASES DE DATOS

2° CUATRIMESTRE DE 2019



Práctica 4

Dependencias Funcionales

Ejercicio 1: Análisis de Dependencias Funcionales con Datos

Para los siguientes esquemas, considerando únicamente la extensión mostrada:

- Describir un ejemplo de anomalía de inserción.
- Describir un ejemplo de anomalía de modificación.
- Describir un ejemplo de anomalía de borrado.

1. Uso de Recursos:

RECURSOS <DNI_Prof, nombre_prof, oficina_prof, recurso, ubicación, fecha_inicio, fecha_fin>

Contiene información relacionada al uso de los recursos de la universidad por parte de los profesores en diferentes períodos de tiempo:

DNI_prof	nombre_prof	oficina_prof	recurso	descripción	fecha_inicio	fecha_fin
33.456.456	P. Martínez	Box 05	R122	Tablet PC	01/01/2016	15/01/2016
44.556.689	L. Moreno	Box 10	R100	Impresora	01/01/2016	15/01/2016
33.456.456	P. Martínez	Box 05	R200	Portátil	12/01/2016	22/01/2016
11.223.344	E. Lopez	Box 23	R122	Tablet PC	16/01/2016	31/01/2016
99.771.122	D. Rodríguez	Box 24	R200	Portátil	23/01/2016	26/01/2016
33.456.456	P. Martínez	Box 05	R300	Scanner	01/01/2016	15/01/2016
99.887.766	F. Sanchez	Box 11	R300	Scanner	01/02/2016	06/02/2016
33.456.456	P. Martínez	Box 05	R122	Tablet PC	01/02/2016	06/02/2016

2. Asignación de Turnos

ASIGNACION <DNI, nombre, codigo_sucursal, direccion_sucursal, turno, fecha>

Contiene información relativa a la asignación de los turnos de trabajo de los empleados de las distintas sucursales de un negocio de moda:

DNI	nombre	codigo_sucursal	direccion_sucursal	turno	fecha
22.140.780	P. Martín	100A	Av. Rivadavia 1430	M	02/09/2016
30.576.380	L. Sanz	100A	Av. Rivadavia 1430	M	02/09/2016
89.751.132	D. Díaz	100A	Av. Rivadavia 1430	T	02/09/2016
22.140.780	P. Martín	200B	Av. Corrientes 756	T	03/09/2016
11.223.344	E. López	300C	Reconquista 230	M	03/09/2016
29.787.726	F. Monte	200B	Av. Corrientes 756	M	03/09/2016
11.223.344	E. López	300C	Reconquista 230	M	04/09/2016
29.787.726	F. Monte	100A	Av. Rivadavia 1430	M	04/09/2016
30.576.380	L. Sanz	100A	Av. Rivadavia 1430	T	04/09/2016
22.140.780	P. Martín	200B	Av. Corrientes 756	M	05/09/2016

3. Actividades Deportivas

ACTIVIDADES <cod_act, nombre_act, DNI_profesor, nombre_profesor, sala, fecha, hora_i, hora_f>

Almacena información sobre la fecha y duración de las actividades deportivas que se organizan en una escuela.

cod_act	nombre_act	DNI_profesor	nombre_profesor	sala	fecha	hora_i	hora_f
01	Pilates	54.332.221	J. Luis Hernández	Pabellón polideportivo	09/09/2016	10:00	11:00
02	Fitness	65.434.527	P. García	Pabellón polideportivo	09/09/2016	10:00	11:00
01	Pilates	54.332.221	J. Luis Hernández	Pabellón polideportivo	11/09/2016	09:30	11:00
01	Pilates	54.332.221	J. Luis Hernández	Pabellón polideportivo	15/09/2016	12:00	13:00
03	Yoga	65.434.527	P. García	Sala Multiusos	15/09/2016	09:00	10:00
05	Elongación	45.673.214	Ana Sanz	Sala Multiusos	04/09/2016	11:00	12:00
04	Gimnasia	54.332.221	J. Luis Hernández	Sala Musculación	01/10/2016	12:00	13:00
03	Yoga	65.434.527	P. García	Sala Multiusos	15/09/2016	11:00	12:00
04	Gimnasia	45.673.214	Ana Sanz	Pabellón polideportivo	01/10/2016	13:00	14:00
02	Fitness	45.673.214	Ana Sanz	Pabellón polideportivo	09/09/2016	10:00	11:00
01	Pilates	65.434.527	P. García	Sala Multiusos	09/09/2016	11:00	12:00

Ejercicio 2: Primera Forma Normal

Para los siguientes esquemas:

- Identifique la clave de la relación.
 - ¿La relación está en Primera Forma Normal? Justifique su respuesta.
 - Si la relación no está en Primera Forma Normal, ¿cómo llevaría esta relación a esta forma normal? Explícite la división de la relación, si corresponde.
- La UNQ tiene una base de datos que registra la información de los alumnos que cursan sus materias de TPI en el año 2019.

Nro. Legajo	Nombre y Apellido	Ciudad	Materias Cursando
84749	Leonardo Gassman	Capital Federal	{Objetos I, Bases de Datos}
93672	Leonardo Volinier	Bernal	{Arquitectura de Software I, Estructuras de Datos}
54367	Pablo Suárez	La Plata	{Introducción a la Prog., Bases de Datos}
87461	Cristian López	La Plata	{Estructuras de Datos, Desarrollo de Software}
74820	Federico Pacheco	Capital Federal	{Introducción a la Prog., Organización de Computadores}
73833	Cristian López	Bernal	{Arquitectura de Software I, Estructuras de Datos}

- Nuevamente, la UNQ tiene una base de datos que registra la información de los alumnos que cursan sus materias de TPI en el año 2015.

Nro. Legajo	Nombre y Apellido	Ciudad	Materias Cursando
84749	Leonardo Orellana	{Capital Federal, Bernal}	{Objetos I, Bases de Datos}
93672	Lucio Quintana	{Bernal, Berazategui}	{Organización de Computadores, Estructuras de Datos}
54367	Juan Acosta	{La Plata, Quilmes}	{Introducción a la Prog., Bases de Datos}
87461	Mariano Varela	{La Plata, Wilde}	{Estructuras de Datos, Desarrollo de Software}
74820	Julián Calvento	{Capital Federal, La Plata}	{Introducción a la Prog., Organización de Computadores}

3. En la biblioteca de la UNQ, tenemos la siguiente relación de los libros (y sus respectivas copias) de referencia de la materia Bases de Datos, y las fechas de 2019 en las cuales cada copia ha sido prestada.

Cod.	Copia	Título	Autores	Año Edición	Fechas_Prestamos
11	A	Fundamentos de BD	{Elmasri, Navathe}	2006	{12/08, 22/09, 15/10}
11	B	Fundamentos de BD	{Elmasri, Navathe}	2006	{21/08, 28/10, 30/10}
11	C	Fundamentos de BD	{Elmasri, Navathe}	2006	{14/08, 19/09, 23/09}
22	A	Sistemas de Manejo de BD	{Ramakrishnan, Gerken}	2002	{10/09, 22/09, 02/10}
22	B	Sistemas de Manejo de BD	{Ramakrishnan, Gerken}	2002	{11/10, 23/10}
22	C	Sistemas de Manejo de BD	{Ramakrishnan, Gerken}	2002	{12/09, 18/09, 22/10}
33	A	Conceptos de BD	{Silberchatz, Korth, Sudarshan}	2010	{11/08, 25/10, 12/11}
33	B	Conceptos de BD	{Silberchatz, Korth, Sudarshan}	2010	{23/10}
33	C	Conceptos de BD	{Silberchatz, Korth, Sudarshan}	2010	{12/08, 18/09}

4. En la OSUNQ (Obra Social de la UNQ), tenemos la siguiente relación de los asociados junto con los médicos que pueden atenderlos y las fechas en que se han hecho consultas en la sede central de la obra social.

Cod. Fam.	Cod. Ind.	Nombre y Apellido	Medicos Habilitados	Año Ingreso OS	Fechas_Consultas
111	01	José Rodriguez	{Rossi, Scarpino}	2002	{10/04, 22/04, 02/05}
111	02	María Rodriguez	{Rossi, Scarpino}	2004	{11/05, 23/05}
111	03	Lucas Rodriguez	{Rossi, Scarpino}	2006	{12/04, 18/05, 22/06}
222	01	Luis Martinez	{García, López, Jara}	2002	{12/03, 22/04, 15/06}
222	02	Lucio Martinez	{García, López, Jara}	2005	{21/05, 28/06, 30/06}
222	03	Roberto Martinez	{García, López, Jara}	2006	{14/04, 19/05, 23/05}
333	01	Ricardo Avila	{López, Jara, Rossi}	2006	{11/02, 25/04, 12/05}
333	02	Tatiana Avila	{López, Jara, Rossi}	2010	{23/06}
333	03	Gisela Avila	{López, Jara, Rossi}	2012	{12/06, 18/06}

5. La UNQ tiene una base de datos que registra los alumnos y los cursos de extensión que cada uno ha realizado en el año 2019.

Nro. Legajo	Apellido	Cursos	Teléfonos
64852	Martínez	{Word, Excel, Powerpoint}	{4365-7100, 4361-7010}
97423	Hernández	{Word, Electrónica}	{4128-5100, 4182-5300, 4128-5600}
87898	Alvarez	{Electricidad, Excel}	{5778-1526, 5778-1545}
96352	Ortega	{Powerpoint, Electrónica, Redes}	{4631-8989, 4584-8963, 4859-6978}
56478	Sanchez	{Electricidad, Bases de Datos, Word}	{4380-7120, 4351-7014}

Ejercicio 3: Ejercicio de Conceptos Teóricos

Supongamos que queremos registrar para una facultad los datos personales de los alumnos, las materias en las que se inscribieron y los exámenes que rindieron. Para esto definimos el esquema de relación FACULTAD y su conjunto de dependencias funcionales:

FACULTAD<nroAlumno, nombre, materia, fechaInscripcion, fechaExamen, nota>

- nroAlumno → nombre (no puede haber dos alumnos con el mismo NroAlumno)
- nroAlumno, materia → fechaInscripcion (se puede inscribir una sola vez en cada materia)
- nroAlumno, materia, fechaExamen → nota (hay una sola nota por examen)

Nota: Bajo las condiciones que muestran las dependencias funcionales, observamos que un alumno podría rendir varias veces la misma materia, porque no está la DF NroAlumno, Materia → FechaExamen.

- Identifique la clave primaria de la relación.
- ¿Cuál de los siguientes grupo de relaciones no tiene pérdida de información? Justifique su respuesta

1. ALUMNO<nroAlumno, nombre>
INSCRIPTO<nroAlumno, materia, fechaInscripcion>
EXAMEN<materia, fechaExamen>
RESULTADO<nroAlumno, materia, nota>
2. ALUMNO<nroAlumno, nombre>
INSCRIPTO<nroAlumno, materia, fechaInscripcion>
EXAMEN<nroAlumno, materia, fechaExamen, nota>

Ejercicio 4: Musimundo

Partiendo de este esquema inicial de una sola relación y sus restricciones

CD<codMaster, tituloGrabacion, cantPistas, nroDePista, duraciónDePista, nombreCancion, autorCancion, añoComposicion>

1. Cada codMaster identifica en forma unívoca a cada CD.
2. Cada CD tiene un número fijo de pistas.
3. Cada tema del CD tiene asignado un número de pista. Obviamente los números de pistas pueden repetirse en diferentes en diferentes CDs, pero no se repiten en el mismo CD.
4. Cada pista en un CD tiene una duración determinada, tiene asociada una canción y un año de composición.

se desprenden las siguientes dependencias funcionales:

- codMaster → tituloGrabacion, cantPistas
- codMaster, nroPista → duracionPista, nombreCancion, añoComposicion

a) Indique la clave primaria de la relación CD.

b) Dadas las relaciones

CD1 <codMaster, tituloGrabacion, cantPistas>
CD2 <codMaster, nroPista, duracionPista, nombreCancion, autorCancion, añoComposicion>

- I. ¿En qué relaciones siguen valiendo las DFs de CD?
- II. Indique las claves primarias.

Ejercicio 5: Padrones electorales

Dada la relación

PADRON<DNI, nombre, direccionCalleNro, localidad, codPostal, nombreHijo, edadHijo, nroEscuelaDondeVota, dirEscuela>

con las restricciones:

1. Cada persona se identifica en forma unívoca con el DNI.
2. Cada persona tiene un nombre, una dirección, vive en una localidad y vota en una sola escuela.
3. Cada localidad tiene un sólo código postal.
4. Cada escuela se identifica con un número en la localidad en la que se localiza. En una localidad hay varias escuelas, pero no hay dos escuelas con el mismo número. En diferentes localidades, puede haber dos escuelas con el mismo número.
5. Cada escuela de una localidad se ubica en una dirección de esa localidad.

6. Cada persona puede tener varios hijos. Una persona no tiene dos hijos con el mismo nombre, pero dos personas podrían tener hijos con el mismo nombre. De cada hijo, se sabe solamente la edad.

se desprenden las siguientes dependencias funcionales:

- $\text{DNI} \rightarrow \text{nombre}, \text{direccionCalleNro}, \text{localidad}, \text{nroEscuelaDondeVota}$
- $\text{localidad} \rightarrow \text{codPostal}$
- $\text{nroEscuelaDondeVota}, \text{localidad} \rightarrow \text{dirEscuela}$
- $\text{DNI}, \text{nombreHijo} \rightarrow \text{edadHijo}$

a) Identificar y justificar una clave primaria para la relación.

Ejercicio 6: Dependencias Funcionales

Para cada esquema dado:

1. Suponer que las relaciones ya están en Primera Forma Normal (es decir, que no hay atributos multivaluados).
2. Encontrar las dependencias funcionales.
3. Determinar la/s clave/s candidata/s.
4. Indicar la clave primaria.

a) **AEROPUERTO**

AEROPUERTO <cod-aerop, provincia, pais, #pista, longitud_pista>

donde:

- El cod-aerop no se repite en diferentes países ni en diferentes provincias. Es decir, el cod-aerop es único por aeropuerto en esta base de datos.
- Cada aeropuerto se ubica en una sola provincia en un solo país.
- En un país puede haber varios aeropuertos. Por ejemplo, en Argentina tenemos el *Aeroparque Jorge Newbery (Buenos Aires)* y el *Aeropuerto El Plumerillo (Mendoza)*.
- Para provincia puede haber varios aeropuertos. Por ejemplo, en la provincia de Buenos Aires tenemos el *Aeroparque Jorge Newbery (C.A.B.A.)* y el *Aeropuerto Astor Piazzolla (Mar del Plata)*.
- En una ciudad de un país puede haber varios aeropuertos. Por ejemplo, en Buenos Aires tenemos el *Aeroparque Jorge Newbery (Buenos Aires)* y el *Aeropuerto Ministro Pistarini (Ezeiza)*.
- Un aeropuerto tiene varias pistas.
- El #pista se puede repetir para distintos aeropuertos.
- La longitud de pista se puede repetir para diferentes pistas dentro del mismo aeropuerto. También se puede repetir para pistas de diferentes aeropuertos.

b) **EMPLEADOS (1)**

EMPLEADO <DNI, nyAp, dir, nro_empleado, departamento>

donde:

- Los números de empleados no se repiten dentro del mismo departamento.
- Cada empleado tiene solamente un DNI.
- El empleado trabaja en un solo departamento.
- Una misma persona no puede tener más de un número de empleado asignado.
- En un domicilio pueden vivir más de un empleado, pero un empleado tiene un único domicilio.

- En un departamento pueden trabajar muchos empleados. Entre diferentes departamentos, pueden repetirse los números de empleados (identificando a diferentes empleados en diferentes departamentos).

c) EMPLEADOS (2)

EMPLEADO <DNI, nyAp, dir, nroEmpleado, departamento>

donde:

- Cada empleado tiene solamente un DNI.
- nroEmpleado puede repetirse en distintos departamentos para distintos empleados.
- Un empleado tiene una sola dirección.
- Una persona puede tener asignado mas de un número de empleado y dentro de un mismo departamento una persona no puede tener asignado mas de un nroEmpleado.

d) GUARDERIA “YBSA”

En una Guardería de zona sur, su dueña lleva una Hoja de Cálculo tipo Excel y nos pidió que le ayudemos a mejorar la forma en que está diseñada. La misma cuenta con las siguientes columnas:

GUARDERIA<dniPadre, nombrePadre, domicilioPadre, nombreNiño, fnacNiño, grupoNiño, comidaGrupo, fechaPago, montoPago>

donde:

- De los padres registramos su dni, nombre y domicilio
- Para cada padre almacenamos todos los niños que nos mandan a cuidar.
- Los nombres de los niños son únicos por cada padre pero se pueden repetir entre padres.
- De los niños conocemos su nombre, su fecha de nacimiento y a qué grupo pertenece (bebé, preinfante, infante).
- El grupo que se le asigna a un niño no depende exclusivamente de su edad (podrían haber dos niños nacidos el mismo día en diferentes grupos).
- A un grupo determinado siempre se le da la misma comida (mamadera, té con galletitas, etc).
- Finalmente registramos los pagos mensuales que hacen los padres.
- Para cada pago mensual por niño anotamos la fecha y el monto.

e) FIESTA

FIESTA <#salon, direccion, capacidad, fecha_fiesta, nom_contratante, cant_invitados, nombre_invitado, cant_mesas, mesa_invitado, dir_contratante, dni_invitado>

donde:

- El atributo #salon es un código único de salón en esta base de datos, es decir no existen dos salones con el mismo código. Cada salón se ubica en una dirección única y tiene una capacidad máxima. En una misma dirección puede haber más de un salón, y la capacidad puede repetirse entre diferentes salones.
- En cada salón se realiza una sola fiesta por día.
- En un día puede haber varias fiestas en diferentes salones.
- Para cada fiesta puede figurar más de un contratante. Cada contratante declara una sola dirección por fiesta.
- Un contratante puede hacer distintas fiestas. En distintas fiestas el mismo contratante puede figurar con diferentes direcciones.
- Cada fiesta tiene asociado una cantidad específica de invitados.
- Cada invitado tiene asociado un número de mesa que depende de cada fiesta.
- La cantidad de mesas del salón varía para cada fiesta.
- Una persona puede ir a más de una fiesta en el mismo salón, en la misma o en diferentes fechas.

f) BUQUE

BUQUE <Nom_buque, NyAp_dueño, DNI_Dueño, tipo_buque, tonelaje, tipo_casco, #viaje, #Puerto_Origen, #Puerto_Destino, Nom_País_Puerto_Destino, Nombre_País_Puerto_Origen, Posicion_Actual, Fecha_Posicion_Actual, NyAp_Pasajero, DNI_Pasajero, Dir_Pasajero, #Puerto_Inicio_Pasajero, #Puerto_Final_Pasajero>

donde:

- El #viaje es un número consecutivo que identifica cada partida de cada buque.
- Un buque hace varios viajes.
- El #viaje se repite para distintos buques
- Un buque puede tener varios dueños.
- El nombre del buque es único.
- Cada buque es de un solo tipo de buque.
- El tonelaje y el casco están determinados por el tipo de buque.
- Un buque reporta su posición una vez por día independientemente del viaje.
- Cada viaje de un buque tiene un puerto origen, un puerto destino y varios puertos intermedios.
- Los buques transportan varios pasajeros.
- Un pasajero tiene un solo DNI y una única dirección independientemente del viaje.
- Un buque en su viaje puede pasar por varios puertos intermedios donde los pasajeros pueden abordar el buque.
- Cada pasajero tiene un puerto de inicio y un puerto final en su viaje. Los puertos donde embarcan o desembarcan los pasajeros no son necesariamente los mismos que el de inicio o final del viaje.
- Cada puerto (sea origen o destino de viaje o pasajero) tiene un identificador único, y un nombre único asociado a cada identificador. Sin embargo, los nombres pueden repetirse entre diferentes puertos.

g) INTERNACIONES

INTERNACION <cod_hospital, cantidad_habitaciones, dirección_internacion_paciente, telefono_internacion_paciente, dni_paciente, domicilio_paciente, nombre_apellido_paciente, domicilio_hospital, ciudad_hospital, director_hospital, fecha_inicio_internacion, cant_dias_internacion>

donde:

- cantidad_habitaciones es la cantidad de habitaciones que hay en cada hospital.
- direccion_internacion_paciente y telefono_internacion_paciente, indican la dirección y el teléfono que deja un paciente cuando se interna.
- domicilio_paciente es el domicilio que figura en el dni del paciente.
- El código de hospital (cod_hospital) es único.
- Existe un único director por hospital. Un director podría dirigir mas de un hospital.
- Un paciente en la misma fecha no puede estar internado en diferentes hospitales.
- En un domicilio_hospital de una ciudad existe un único hospital.
- Cada paciente tiene un solo nombre y un solo DNI.

Observaciones: En este ejercicio, a partir de los atributos `cod_hospital` y `{domicilio_hospital, ciudad_hospital}`, se pueden generar dos claves candidatas. Esto se debe a que se podrían definir dos dependencias funcionales:

a) `cod_hospital` → `domicilio_hospital`, `ciudad_hospital`

b) `domicilio_hospital`, `ciudad_hospital` → `cod_hospital`

La sugerencia para la solución es elegir cuál de las dos dependencias funcionales vale y plantear el resto de las DFs en función de la DF elegida. Por ejemplo,

- a) Si elijo $\text{cod_hospital} \rightarrow \text{domicilio_hospital}, \text{ciudad_hospital}$, entonces puedo definir que $\text{cod_hospital} \rightarrow \text{director_hospital}$.
- b) Si elijo $\text{domicilio_hospital}, \text{ciudad_hospital} \rightarrow \text{cod_hospital}$, entonces puedo definir que $\text{domicilio_hospital}, \text{ciudad_hospital} \rightarrow \text{director_hospital}$.

h) EVALUACIONES

EVALUACIONES <dni_alumno, #legajo_alumno, cod_materia, nombre_materia, fecha_evaluacion_materia, nombre_apellido_alumno, dirección_alumno, ciudad_alumno, cod_profesor_responsable_examen, nombre_profesor, nota_evaluacion>

donde:

- cod_materia representa el código de la materia que se rindió
- $\text{fecha_evaluacion_materia}$ es la fecha en la que el alumno se presenta a rendir una materia. Un alumno puede presentarse a rendir mas de una materia en una fecha, sin embargo, no puede presentarse en la misma fecha a rendir más de una vez la misma materia
- $\text{cod_profesor_responsable_examen}$, es el código del profesor que le tomo examen a un alumno, en una fecha determinada, para una materia dada. Este código es único, sin embargo el nombre de los profesores se puede repetir para diferentes códigos de profesores
- nota_evaluacion es la nota que saca un alumno al presentarse en una evaluación de una materia en una fecha determinada
- Un profesor puede ser responsable de varias evaluaciones en la misma fecha y cada evaluación tiene un profesor responsable (en la misma fecha para la misma materia, a diferentes alumnos podrían tomarle diferentes profesores)
- En la evaluación de una materia en una fecha, para un alumno, solo puede haber un profesor asignado y un profesor puede participar de diferentes evaluaciones en la misma fecha de distintas materias
- el código de materia es único, sin embargo, diferentes códigos de materias podrían llevar el mismo nombre de materia
- El dni_alumno es único por alumno
- El $\text{\#legajo}$ es único por alumno