

EVALUACION	Parcial	GRUPO	N2C y N2G	FECHA	28/11/2023
MATERIA	Bases de Datos 1				
CARRERA	Analista Programador / Analista en tecnologías de Información				
CONDICIONES	- Puntos: 40 - Duración: 2 horas - Sin material				

EJERCICIO 1

Cuatro de los mejores hombres de un comando militar sobreviven como mercenarios después de haber sido culpados de un delito que no habían cometido. El líder, teniente Gral. John "Hannibal" Smith se está por retirar y quiere crear una franquicia del servicio. Para eso lo contrata para modelar la base de datos según el siguiente detalle:



- Poseen distintos tipos de misiones que se adaptan a las necesidades del momento (ej: Rescate, Investigación, Secuestro, etc), y se identifican por un código alfanumérico. Posee una descripción general y un precio. Las misiones son diseñadas por el Sargento Bosco Alberto "B.A." Baracus.
- Los clientes que contratan las misiones poseen su dni que los identifican, nombre y apellido. Cuenta con los teléfonos y un apodo secreto que es único.
- Un cliente puede contratar distintos tipos de misiones.
- A cada contratación se le asocia los recursos materiales necesarios. Los tipos de recursos que pueden ser vehículos, armamento y vestimenta. Se los identifica con un id autonumérico, poseen una descripción.
- Cada vez que se asigna una vestimenta se registra el talle y color. En el caso del vehículo se registra, además, la matrícula del mismo.
- Se cuenta con los datos de los "malos", siendo el objetivo que tiene cada misión, variando según el cliente. De ellos se sabe el alias que los identifica, y la última ubicación que se los vio. Algunas veces la misión puede tener varios objetivos, pero no se repiten por cliente/misión

Se pide:

- Modelo Entidad-Relación. Se deben especificar claramente los atributos de cada entidad y/o relación, la participación y cardinalidad de asignación, agregaciones, generalizaciones, entidades débiles y toda restricción (no estructural y de dominio) que no se pueda deducir del modelo. **(10 puntos)**
- Realizar el Modelo Relacional en 3FN **(10 puntos)**

EJERCICIO 2

Dado el siguiente modelo relacional:

SUCURSAL (idsuc, descripcion)

Sucursales que posee la empresa

TIPODOCUMENTO (idtipo, descripcion)

Tipos de documentos/comprobantes. Ejemplo: Facturas, Remitos, Recibos, Cheques

FA: Facturas, RE: Remitos, CQ: Cheques, RC: Recibos

CLIENTE (idCli, nombre)

Registro de todos los clientes

DOCUMENTO (idsuc, idtipo, numdoc, fecha, idclie, montototal)

Transacciones realizadas como ser: ventas, remitos, pagos, etc, dependiendo del tipo de documento.

DOC_ITEM (idsuc, idtipo, numdoc, nroltem, idart, cantidad, preciounitario, %descuento, flagControl)

Lineas/Items que tiene cada documento, indicando el articulo asociado, donde el campo flagControl debe ser valores positivos

ARTICULO (idart, descripcion)

Artículos que vende la empresa

- A. Escribir el script para crear la tabla "doc_items" con todas las restricciones que surgen del modelo **(2 puntos)**
- B. Listar el monto total facturado por descripción de documento, en el año 2021. Listar únicamente aquellos documentos con total facturado superior a 500. **(3 puntos)**

Descripcion	MontoTotal
-------------	------------

- C. Listar las cinco sucursales que tuvieron menos unidades vendidas **(2 puntos)**

Descripcion

- D. Listado de artículos vendidos, indicando la cantidad de facturas y el promedio de \$ facturado vendido del item **(3 puntos)**

idArt	Descripcion	# Cant. Facturas	Promedio Facturado
-------	-------------	------------------	--------------------

EJERCICIO 3

Utilizando el modelo relacional del ejercicio 2, elija la opción correcta. Justificar la elección, escribiendo la consulta correcta de corresponder.

Listar los IDs de sucursales que únicamente vendieron artículos cuya descripción termina con la letra "X" **(5 puntos)**

- A)

```
SELECT DI.IDSUC
FROM DOC_ITEM DI
INNER JOIN ARTICULO A A.IDART = DI.IDART
WHERE A.DESCRIPCION LIKE '%X'
```
- B)

```
SELECT DI.IDSUC
FROM DOC_ITEM DI
INNER JOIN ARTICULO A A.IDART = DI.IDART
WHERE A.DESCRIPCION LIKE '%X'
EXCEPT
SELECT DI.IDSUC
FROM DOC_ITEM DI
INNER JOIN ARTICULO A A.IDART = DI.IDART
WHERE A.DESCRIPCION NOT LIKE '%X'
```
- C)

```
SELECT S.IDSUC
FROM SUCURSALES S
INNER JOIN DOC_ITEM DI ON DI.IDSUC = S.IDSUC
INNER JOIN ARTICULOS A ON A.IDART = DI.IDART
WHERE A.DESCRIPCION LIKE '%X'
```
- D) NINGUNA DE LAS ANTERIORES

Listar los artículos que nunca tuvieron ventas **(5 puntos)**

- A)

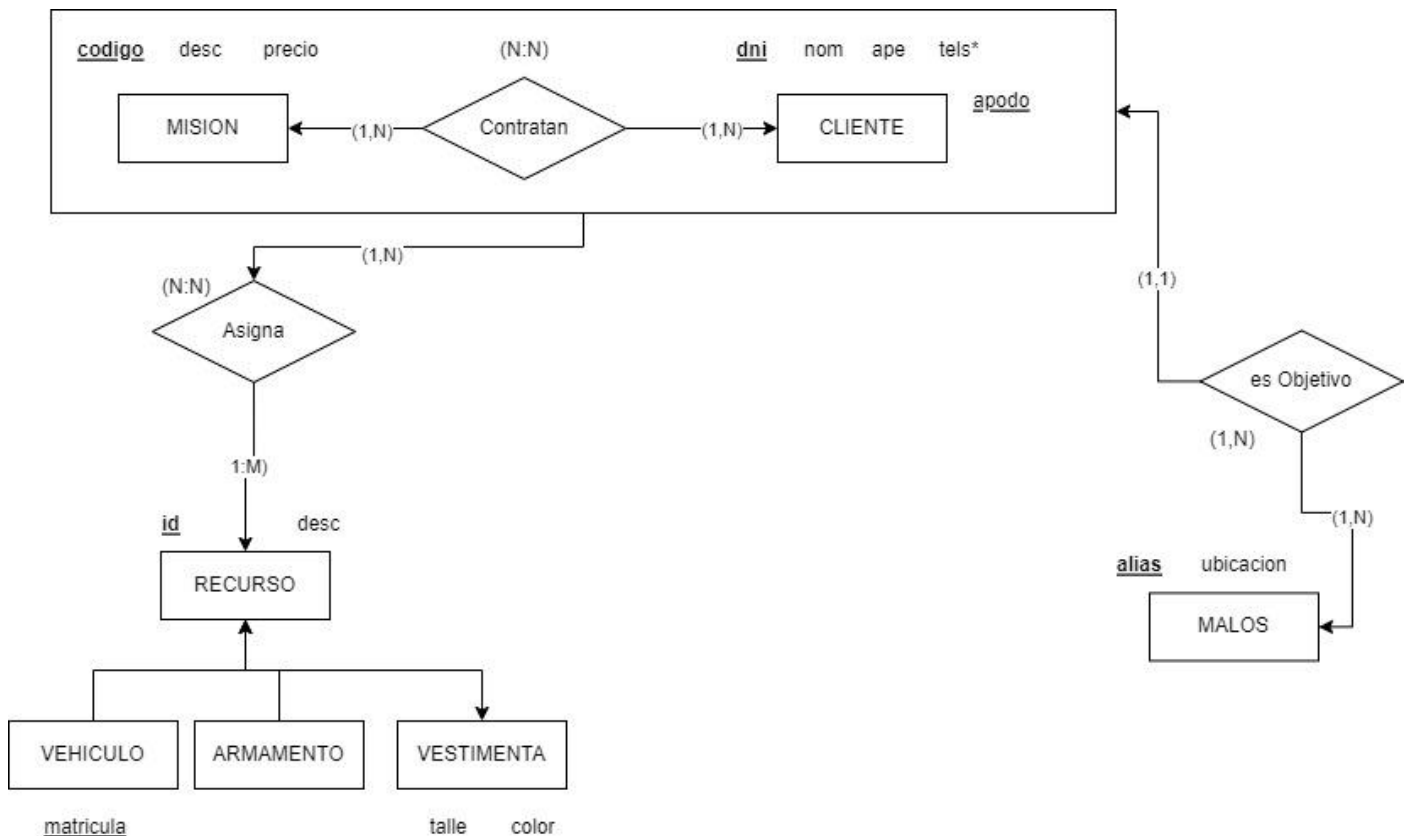
```
SELECT A.IDART, A.DESCRIPCION
FROM ARTICULOS A
INNER JOIN DOC_ITEMS DI ON DI.IDART = A.IDART
WHERE SUM(CANTIDAD) = 0
```
- B)

```
SELECT A.IDART, A.DESCRIPCION
FROM ARTICULOS A
INNER JOIN DOC_ITEMS DI ON DI.IDART = A.IDART
GROUP BY A.IDART, A.DESCRIPCION
HAVING SUM(CANTIDAD) = 0
```
- C)

```
SELECT A.IDART, A.DESCRIPCION, COUNT(CANTIDAD) AS CANT
FROM ARTICULOS A
INNER JOIN DOC_ITEMS DI ON DI.IDART = A.IDART
GROUP BY A.IDART, A.DESCRIPCION
ORDER BY CANTIDAD DESC
```
- D) Opciones B y C
- E) Ninguna de las anteriores

SOLUCION

EJERCICIO 1



MR

MISION (codigo, desc, precio)

CLIENTE (dni, nom, ape, apodo)

AK: apodo

CONTRATAN (codMision, dni)

FK: codMision → MISION (codigo)

FK: dni → CLIENTE (dni)

RECURSO (id, desc)

ASIGNA (codMision, dni, idRecurso)

FK: codMision, dni → CONTRATAN (codMision, dni)

FK: idRecurso → RECURSO (id)

VEHICULO (id, matricula)

AK: matricula

FK: id → RECURSO (id)

ARMAMENTO (id)

FK: id → RECURSO (id)

VESTIMENTA (id, talla, color)

FK: id → RECURSO (id)

MALOS (alias, ubicación, codMision, dni)
FK: codMision, dni → CONTARTAN (codMision, dni)

EJERCICIO 2

A

```
CREATE TABLE doc_items (  
  idSuc int,  
  idTipo VARCHAR(2),  
  numdoc int,  
  nroltem int,  
  idart int,  
  cantidad int,  
  preciounitario numeric (8,2),  
  %descuento numeric (8,2),  
  flagControl int CHECK (flagControl > 0),  
  CONSTRAINT PK PRIMARY KEY(idSuc,idTipo,numdoc,nroltem),  
  CONSTRAINT FK_DOC FOREIGN KEY(idSuc, idTipo, numDoc) REFERENCES DOCUMENTO (idSuc,  
  idTipo, numDoc)  
  CONSTRAINT FK_Art FOREIGN KEY(idart) REFERENCES Articulos (idart))
```

B

```
SELECT td.DESCRIPCION, SUM(di.cantidad*di.precioUnitario)  
FROM DOC_ITEM di  
INNER JOIN TIPODOCUMENTO td ON di.idTipo = td.idTipo  
WHERE YEAR(D.FECHA) = 2021  
GROUP BY td.idTipo, td.descripcion  
HAVING SUM(di.cantidad*di.precioUnitario) > 500
```

C

```
SELECT TOP 5 S.DESCRIPCION  
FROM SUCURSALES S  
INNER JOIN DOC_ITEMS I ON I.IDSUC = S.IDSUC  
GROUP BY S.DESCRIPCION  
ORDER BY SUM(I.UNID_VEND)
```

D

```
SELECT a.idART, a.DESCRIPCION, COUNT (DISTINCT D.NRODOC), AVG(I.cantidad *  
  I.PRECIOUNITARIO)  
FROM ARTICULOS A  
INNER JOIN DOC_ITEMS I ON I.idART = A.idART  
GROUP BY A.CODART,A.DESCRIPCION
```

EJERCICIO 3

OPCION B

OPCION D

CONSULTA SOLUCIÓN:

```
SELECT A.IDART, A.DESCRIPCION
FROM ARTICULOS A
EXCEPT
SELECT DISTINCT A.IDART,D.DESCRIPCION
FROM ARTICULOS A
INNER JOIN DOC_ITEMS DI ON DI.IDART = A.IDART
```