

Proyecto Final de Base de datos

Profesor: Andrés Quintero Prada

Estudiante: Rafael Eduardo Valencia Buitrago

Universidad Tecnológica de Pereira

Facultad de Ingenierías

Programa de Ingeniería de Sistemas y computación

IS644-Base De Datos I



UNIVERSIDAD TECNOLÓGICA DE PEREIRA

Asignatura de Bases de Datos

ENUNCIADO.

Una empresa de construcciones desea realizar un sistema de información que le permita realizar un correcto seguimiento de las obras. Para ello nos dan la siguiente información:

El nuevo sistema debe ser capaz de guardar información de las obras ya terminadas, de las que actualmente están en desarrollo y de las que se piensa comenzar en un futuro breve.

Cada obra necesita para su realización de diversos tipos de planos: de ejecución, del sistema eléctrico y del sistema de desagüe (todos los planos disponen de un código que permite ordenarlos dentro de cada tipo de planos, una escala y una fecha de realización).

Es posible que haya planos que se utilicen en varias obras, pues a veces las casas a construir son idénticas. Dado que las obras se construirán bien en un solar o bien en un terreno, es necesario tener también el plano del solar o del terreno a construir, siendo posible en este último caso, que en un terreno se realicen varias obras. Tanto si es un solar o un terreno es necesario disponer de la referencia de su ubicación (dirección, coordenadas).

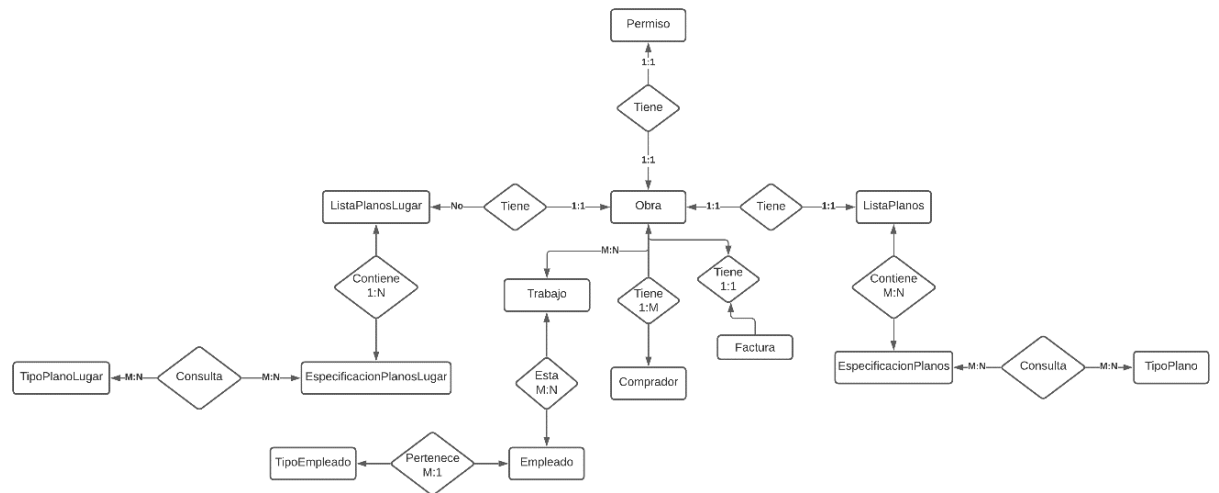
Toda obra para iniciar su construcción necesita disponer del permiso del ayuntamiento. Y una vez acabada necesita que la den de paso. Es necesario por lo tanto conocer si estos permisos han sido concedidos y cuando.

Toda obra necesita de personal para su realización, y dado que hay que pagarles, es necesario conocer en qué obras ha trabajado cada obrero y cuántos días trabajó en cada una (tanto en el mes actual como en los anteriores). Hay que tener en cuenta que toda obra necesita tener siempre al menos a tres obreros y un jefe de obra.

Las cuentas de gastos han de ser independientes para cada obra, eso quiere decir que es necesario saber a qué obra se corresponden que facturas y cuales están pagadas o hay que pagar.

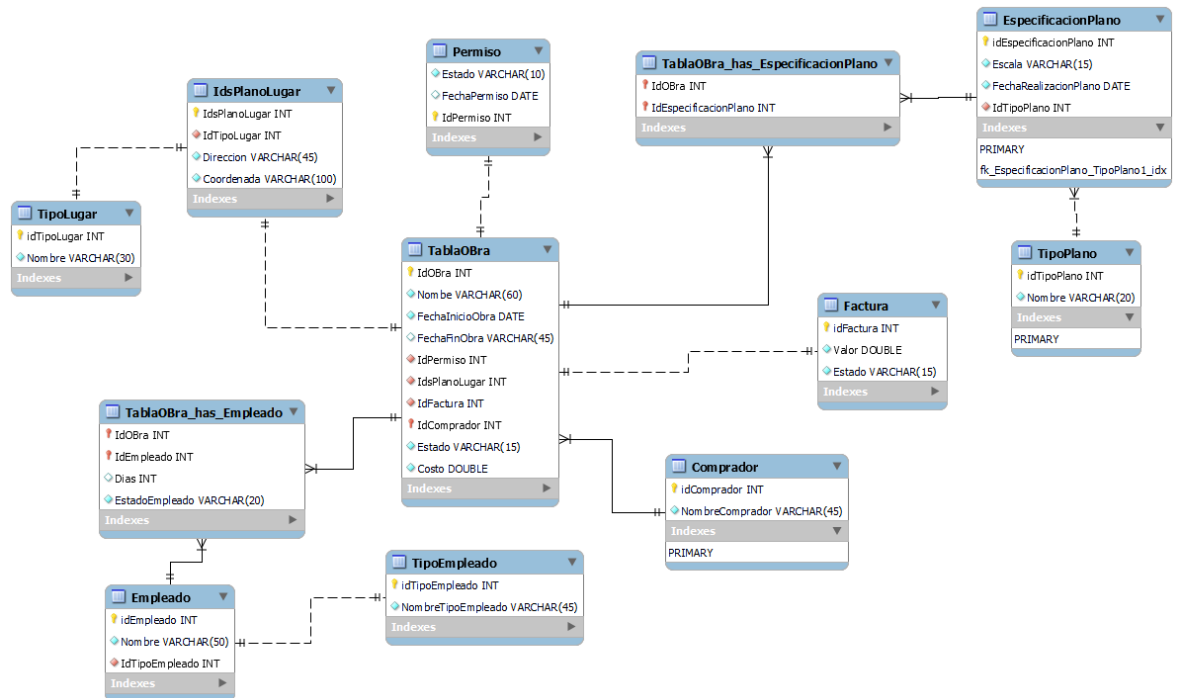
Finalmente, un dato muy importante es el hecho de que la obras son realizadas siempre por encargo (siempre se dispone de comprador) por lo que es necesario saber quién encargó la construcción (en el caso de una obra en terreno) o cada apartamento si es una obra en un edificio. En todos los casos hay que saber la cantidad que van a pagar.

1. Modelo Entidad Relación



Se encuentra de forma independiente esta imagen dentro de la carpeta compartida para una mejor visualización. La carpeta se llama “Modelos”

2. Modelo Relacional



Se encuentra de forma independiente esta imagen dentro de la carpeta compartida para una mejor visualización. La carpeta se llama “Modelos”

3. Aplicar Integridad referencial

Hecho

4. Normalización mínima

La normalización de las tablas se encuentra en un archivo formato Excel llamado *“EmpresaConstruccion”*

5. Modelo físico

Se encontrará el script dentro de la carpeta compartida donde, esta este documento, en la carpeta *“scripts y más/scripts de la base de datos”*

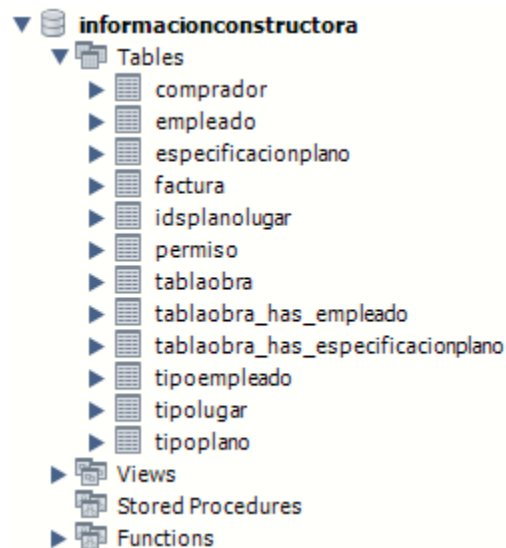
6. Diccionario de Datos

Tengo un problema para mostrar el diccionario, al buscarlo en internet por el localhost, no me aparece las tablas solo el código HTML, sin embargo, ese archivo HTML estará en la carpeta compartida, en la carpeta *“scripts y más/Diccionario de datos”*

7. Creación del modelo en la Base de Datos en el SGBD haciendo uso de DDL o importando los SCRIPT en la base de datos

Importé el script de la base de datos para su creación, con la herramienta que proporciona el Workbench, este script se encontrara en la carpeta compartida.

“scripts y más/scripts de la base de datos”



8. Realice las operaciones de INSERT necesarias para almacenar 10 registros en cada tabla, por lo menos 1 INSERT deben realizarse obteniendo la clave primaria de la tabla fuerte.

1)

```
insert into tipoplano(idTipoPlano,Nombre)
values(1,'Ejecucion'),(2,'Sistema Electrico'),(3,'Sistema Desague')
```

2)

```
insert into tipolugar(idTipoLugar,Nombre)
values(1,'Solar'),(2,'Terreno')
```

3)

```
insert into tipoempleado(idTipoEmpleado,NombreTipoEmpleado)
values(1,'Obrero'),(2,'Jefe de obra')
```

4)

```
insert into permiso(Estado,FechaPermiso)
values('Concedido','2020-07-12'),('Revicion',null),
('Concedido','2021-03-23'),
('Concedido','2021-02-14'),('Revicion',null),
('Concedido','2020-09-27'),('Concedido','2020-11-14'),
('Concedido','2021-06-09'),('Revicion',null),
('Concedido','2021-08-30')
```

5)

```

insert into idsplanolugar(IdsPlanoLugar,IdTipoLugar,Direccion,Coordenada)
values(333,1,'CARRERA 5 #743-31','32 08'59.96" N, 110 50'09.03"W'),
(313,2,'CALLE 76 #31-9','32°40'34.19"N 117° 9'27.58"W'),
(491,1,'Bis 12 calle 14 #88','4°17'21.49" S 31°23'46.46" E'),
(996,1,'Avenida Las Americas','33.747252, -112.633853'),
(12,2,'Cerritos entrada 5','33.927911, -118.38069'),
(231,1,'CARRERA 43 #42-11a','41.303921, -81.901693'),
(1,2,'La vaquita Km 2','-33.350534, -71.653268'),
(482,1,'Bis 28 calle 9a #66e','-18.529211, -70.249941'),
(309,2,'carrera 6 #12-91','51°19'18.13"N, 6°34'35.64"E'),
(669,1,'Carrera 11 #0-21','52°29'52.24"N 13°27'13.67"E')

```

6)

```

insert into empleado(idEmpleado,Nombre,IdTipoEmpleado)
values(1,'Rafael',2),(2,'Manuela',2),
(3,'Richard',2),(4,'Oscar',1),
(5,'Eduardo',1),(6,'Penelope',1),
(7,'Lucas',1),(8,'Marcos',1),
(9,'Mauricio',1),(10,'Luis',2),
(11,'Fabian',1),(12,'Juan',2),
(13,'Antonio',1),(14,'Antonela',1),
(15,'Lorena',1),(16,'Felipe',1),
(17,'Jina',1),(18,'Sergio',1),
(19,'Pablo',1),(20,'Camilo',2)

```

7)

```

insert into especificacionplano(idEspecificacionPlano,Escala,FechaRealizacionPlano,IdTipoPlano)
values (1,'1:10','2020-06-20',1),
(2,'1:10','2020-06-21',2),
(3,'1:10','2020-06-21',3),
(4,'1:10','2021-03-10',1),
(5,'1:10','2021-03-10',2),
(6,'1:10','2021-03-11',3),
(7,'1:100','2021-01-27',1),
(8,'1:100','2021-01-28',2),
(9,'1:100','2021-01-27',3),
(10,'1:1000','2020-09-10',1),
(11,'1:100','2020-09-12',2),
(12,'1:100','2020-09-10',3),
(13,'1:100','2020-10-29',1),
(14,'1:10','2020-10-29',2),
(15,'1:10','2020-10-29',3),
(16,'1:1000','2021-05-28',1),
(17,'1:10','2021-05-29',2),
(18,'1:10','2021-05-29',3),
(19,'1:100','2021-08-24',1),
(20,'1:10','2021-08-24',2),
(21,'1:10','2021-08-25',3),
(22,'1:10','2020-09-19',1),
(23,'1:10','2020-09-21',2),
(24,'1:10','2020-09-21',3)

```

8)

```

insert into comprador(idComprador,NombreComprador)
values(1,'Luis carlos'),(2,'Hermanos Shelby'),
(3,'Carlos slim'),(4,'Montgomery Burns'),(5,'Industria metarlugica caldas'),
(6,'Javier Miliei'),(7,'Bernando fact'),(8,'Finn el Humano')

```

9)

```

insert into factura(idFactura,Valor,Estado)
values(1,500.000,'Pagado'),(2,300.000,'No pagado')
,(3,150.000,'Pagado'),(4,800.000,'Pagado'),(5,100.000,'Pagado')
,(6,1000.000,'Pagado'),(7,290.000,'Pagado'),(8,920.000,'No pagado')
,(9,450.000,'No pagado'),(10,560.000,'Pagado')

```

10)

```

insert into tablaobra(Nombre,FechaInicioObra,FechaFinObra,IdPermiso,IdsPlanoLugar,IdFactura,IdComprador,Estado,Costo)
values('Casa 278a','2020-07-30','2020-11-20',1,313,5,8,'Terminada',200.000),
('X100','2022-01-25',null,2,313,8,1,'Planificacion',1100.000),
('Edificio 23B','2021-04-02','2021-11-30',4,491,4,5,'Terminada',950.000),
('Fabrica de carros I','2021-02-26','2021-10-23',5,12,10,2,'Terminada',700.000),
('Fabrica de carros II','2022-05-13',null,6,12,9,2,'Planificacion',600.000),
('Parque Buenos aires','2020-12-15','2021-03-19',7,231,7,6,'Terminada',340.000),
('Factory B','2020-12-10','2021-06-09',8,482,6,7,'Terminada',1200.000),
('Casa 13d','2021-07-12','2021-12-03',9,313,3,8,'Terminada',250.000),
('Almacen IB','2022-03-25',null,10,309,2,3,'Planificacion',450.000),
('Iglesia de Marco','2021-08-30','2022-02-09',11,669,1,4,'Inactiva',550.000)

```

11)

```

insert into tablaobra_has_empleado(IdOBra,IdEmpleado,Dias,EstadoEmpleado)
values(1,1,113,'Inactivo'),(1,4,113,'Inactivo'),(1,5,113,'Inactivo'),(1,6,113,'Inactivo'),
(2,2,0,'Inactivo'),(2,7,0,'Inactivo'),(2,8,0,'Inactivo'),(2,9,0,'Inactivo'),
(3,3,242,'Inactivo'),(3,4,242,'Inactivo'),(3,5,242,'Inactivo'),(3,9,242,'Inactivo'),
(4,20,239,'Inactivo'),(4,18,239,'Inactivo'),(4,19,239,'Inactivo'),(4,17,239,'Inactivo'),
(5,12,0,'Inactivo'),(5,13,0,'Inactivo'),(5,15,0,'Inactivo'),(5,14,0,'Inactivo'),
(6,1,94,'Inactivo'),(6,9,94,'Inactivo'),(6,4,94,'Inactivo'),(6,7,94,'Inactivo'),
(7,2,181,'Inactivo'),(7,5,181,'Inactivo'),(7,13,181,'Inactivo'),(7,11,181,'Inactivo'),
(8,12,144,'Inactivo'),(8,15,144,'Inactivo'),(8,14,144,'Inactivo'),(8,6,144,'Inactivo'),
(9,2,0,'Inactivo'),(9,17,0,'Inactivo'),(9,12,0,'Inactivo'),(9,5,0,'Inactivo'),
(10,1,104,'Activo'),(10,7,104,'Activo'),(10,10,104,'Activo')

```

12)

```

INSERT into tablaobra_has_especificacionplano(IdOBra,IdEspecificacionPlano)
VALUES(1,1),(1,2),(1,3),
(2,19),(2,20),(2,21),
(3,7),(3,8),(3,9),
(4,10),(4,11),(4,12),
(5,10),(5,11),(5,12),
(6,22),(6,23),(6,24),
(7,13),(7,14),(7,15),
(8,1),(8,2),(8,3),
(9,16),(9,17),(9,18),
(10,4),(10,5),(10,6)

```


9. Modifique 5 registros de la tabla que contiene los registros del personal.

Todas las instrucciones de INSERT, UPDATE realizadas debe estar bajo transacciones

Antes:

idEmpleado	Nombre	IdTipoEmpleado
1	Rafael	2
2	Manuela	2
3	Richard	2
11	Fabian	1
18	Sergio	1

```
use informacionconstructora;  
start transaction;  
  
update `informacionconstructora`.`empleado` set `Nombre` = 'Rafael Eduardo Valencia' where (`idEmpleado` = 1);  
update `informacionconstructora`.`empleado` set `Nombre` = 'Manuela Arroyo Gomez' where (`idEmpleado` = 2);  
update `informacionconstructora`.`empleado` set `Nombre` = 'Jonnathan Quintero' where (`idEmpleado` = 3);  
update `informacionconstructora`.`empleado` set `Nombre` = 'Malefica Buitrago' where (`idEmpleado` = 18);  
update `informacionconstructora`.`empleado` set `Nombre` = 'Brayan andres' where (`idEmpleado` = 11);  
  
commit;
```

Después del update:

idEmpleado	Nombre	IdTipoEmpleado
1	Rafael Eduardo Valencia	2
2	Manuela Arroyo Gomez	2
3	Jonnathan Quintero	2
11	Brayan andres	1
18	Malefica Buitrago	1

10. Realizar las siguientes consultas haciendo uso de WHERE, INNER o UNION.

1. Listar las obras con estado (Terminadas, En Desarrollo, En Planificación).

```
select tablaobra.Nombre as NombreObra,estado from tablaobra
```

NombreObra	estado
Casa 278a	Terminada
X100	Planificacion
Edificio 23B	Terminada
Fabrica de carros I	Terminada
Fabrica de carros II	Planificacion
Parque Buenos aires	Terminada
Factory B	Terminada
Casa 13d	Terminada
Almacen IB	Planificacion
Iglesia de Marco	Inactivo

2. Listar las obras y Planos asociados.

```
select tablaobra.Nombre as NombreObra1,  
tablaobra.IdsPlanoLugar as IdPlanoDelLugar,  
tablaobra_has_especificacionplano.IdEspecificacionPlano as IdPlano  
from tablaobra, tablaobra_has_especificacionplano  
where tablaobra.IdOBra = tablaobra_has_especificacionplano.IdOBra
```

La cosa de porque se repite es por la forma en la cual hice la lógica del modelo y es que como una obra debe de tener cuatro tipos de planos diferentes, donde 1 es del lugar pero los otros son de la construcción como tal, entonces la tablaobra_has_especificacionplano tiene los id relacionados al plano y al id de la obra que lo usara, la cosa es que el idplanoLugar no pertenece a la tabla anteriormente nombrada entonces por lo menos en la base de datos no se genera esa repetición de información, además de que solo guarda ids no el nombre del proyecto.

NombreObral	IdPlanoDelLugar	IdPlano
Casa 278a	313	1
Casa 278a	313	2
Casa 278a	313	3
X100	313	19
X100	313	20
X100	313	21
Edificio 23B	491	7
Edificio 23B	491	8
Edificio 23B	491	9
Fabrica de carros I	12	10
Fabrica de carros I	12	11
Fabrica de carros I	12	12
Fabrica de carros II	12	10

Fabrica de carros II	12	11
Fabrica de carros II	12	12
Parque Buenos aires	231	22
Parque Buenos aires	231	23
Parque Buenos aires	231	231
Factory B	482	13
Factory B	482	14
Factory B	482	15
Casa 13d	313	1
Casa 13d	313	2
Casa 13d	313	3
Almacen IB	309	16

Almacen IB	309	17
Almacen IB	309	18
Iglesia de Marco	669	4
Iglesia de Marco	669	5
Iglesia de Marco	669	6

3. Listar las obras que tienen permisos para iniciar labores.

```
select tablaobra.Nombre as NombreObral,  
permiso.Estado as EstadoDelPermiso  
from tablaobra, permiso  
where tablaobra.IdPermiso = permiso.IdPermiso  
and permiso.Estado = 'Concedido'
```

NombreObral	EstadoDelPermiso
Casa 278a	Concedido
Edificio 23B	Concedido
Fabrica de carros I	Concedido
Parque Buenos aires	Concedido
Factory B	Concedido
Casa 13d	Concedido
Iglesia de Marco	Concedido

4. Listar las obras que no cumplen con el personal requerido.

```
select tablaobra.IdOBra as IdObra, count(tablaobra_has_empleado.IdEmpleado) as trabajadores  
from tablaobra,tablaobra_has_empleado,empleado  
where tablaobra.IdOBra = tablaobra_has_empleado.IdOBra  
and empleado.idEmpleado = tablaobra_has_empleado.IdEmpleado  
and tablaobra.FechaFinObra > now()  
group by tablaobra.IdOBra  
having trabajadores < 4
```

IdObra	trabajadores
10	3

La especificación de la fecha es porque puede haber proyectos que no tengan el número de trabajadores mínimos, pero ya se termino el proyecto y/o aun no se ha dado los permisos adecuados para iniciar la obra, entonces no se cuenta los que aun no tiene fechas de finalización.

5. Listar los compradores y obras solicitadas por encargo.

```
select tablaobra.Nombre as Proyecto, comprador.NombreComprador
from tablaobra,comprador
where tablaobra.IdComprador = comprador.idComprador
```

Proyecto	NombreComprador
X100	Luis carlos
Fabrica de carros I	Hermanos Shelby
Fabrica de carros II	Hermanos Shelby
Almacen IB	Carlos slim
Iglesia de Marco	Montgomery Burns
Edificio 23B	Industria metalurgica caldas
Parque Buenos aires	Javier Miliei
Factory B	Bernando fact
Casa 278a	Finn el Humano
Casa 13d	Finn el Humano

6. Listar los promedios de gastos por obra.

```
select tablaobra.IdOBra,tablaobra.Nombre , factura.Valor as PromedioGastos
from tablaobra,factura
where tablaobra.IdFactura = factura.idFactura
order by tablaobra.IdOBra asc;
```

IdOBra	Nombre	PromedioGastos
1	Casa 278a	100
2	X100	920
3	Edificio 23B	800
4	Fabrica de carros I	560
5	Fabrica de carros II	450
6	Parque Buenos aires	290
7	Factory B	1000
8	Casa 13d	150
9	Almacen IB	300
10	Iglesia de Marco	500

7. Listar la cantidad de personas por obra indicando el cargo.

```
select tablaobra.IdOBra as Idobra,  
       tablaobra.Nombe as Obra,  
       count(tablaobra_has_empleado.IdEmpleado) as Empleados,  
       tipoempleado.NombreTipoEmpleado as TipoEmpleado  
  from tablaobra,tablaobra_has_empleado,tipoempleado,empleado  
 where tablaobra.IdOBra = tablaobra_has_empleado.IdOBra  
 and tablaobra_has_empleado.IdEmpleado = empleado.idEmpleado  
 and empleado.IdTipoEmpleado = tipoempleado.idTipoEmpleado  
 and tipoempleado.NombreTipoEmpleado = 'Obrero'  
 group by tablaobra.IdOBra;
```

Idobra	Obra	Empleados	TipoEmpleado
1	Casa 278a	3	Obrero
3	Edificio 23B	3	Obrero
6	Parque Buenos aires	3	Obrero
7	Factory B	3	Obrero
9	Almacen IB	2	Obrero
8	Casa 13d	3	Obrero
2	X100	3	Obrero
10	Iglesia de Marco	1	Obrero
5	Fabrica de carros II	3	Obrero
4	Fabrica de carros I	3	Obrero

```
select tablaobra.IdOBra as Idobra,  
       tablaobra.Nombe as Obra,  
       count(tablaobra_has_empleado.IdEmpleado) as Empleados,  
       tipoempleado.NombreTipoEmpleado as TipoEmpleado  
  from tablaobra,tablaobra_has_empleado,tipoempleado,empleado  
 where tablaobra.IdOBra = tablaobra_has_empleado.IdOBra  
 and tablaobra_has_empleado.IdEmpleado = empleado.idEmpleado  
 and empleado.IdTipoEmpleado = tipoempleado.idTipoEmpleado  
 and tipoempleado.NombreTipoEmpleado = 'Jefe de obra'  
 group by tablaobra.IdOBra;
```

Idobra	Obra	Empleados	TipoEmpleado
1	Casa 278a	1	Jefe de obra
6	Parque Buenos aires	1	Jefe de obra
10	Iglesia de Marco	2	Jefe de obra
2	X100	1	Jefe de obra
7	Factory B	1	Jefe de obra
9	Almacen IB	2	Jefe de obra
3	Edificio 23B	1	Jefe de obra
5	Fabrica de carros II	1	Jefe de obra
8	Casa 13d	1	Jefe de obra
4	Fabrica de carros I	1	Jefe de obra

No logre generar una consulta que tuviera ambas en una misma tabla.

12. Cree una vista con el punto 10.4.

```
create view ObrasSinEmpleados AS
select tablaobra.IdOBra as IdOBra, count(tablaobra_has_empleado.IdEmpleado) as trabajadores
from tablaobra,tablaobra_has_empleado,empleado
where tablaobra.IdOBra = tablaobra_has_empleado.IdOBra
and empleado.idEmpleado = tablaobra_has_empleado.IdEmpleado
and tablaobra.FechaFinObra > now()
group by tablaobra.IdOBra
having trabajadores < 4;

show create view obrassinempleados;

select * from obrassinempleados
```

IdOBra	trabajadores
10	3

13. Cree una función que permita contar la cantidad de empleados por obra.

```

use informacionconstructora;
DELIMITER \
Drop function if exists ContadorTrabajadores\
create function ContadorTrabajadores (id int)
    Returns int deterministic
begin
    declare TotalTrabajadores int;
    select count(tablaobra_has_empleado.IdEmpleado) into TotalTrabajadores
    from tablaobra,tablaobra_has_empleado,empleado
    where tablaobra.IdOBra = id
    and id = tablaobra_has_empleado.IdOBra
    and empleado.idEmpleado = tablaobra_has_empleado.IdEmpleado;

    return TotalTrabajadores;
end\
DELIMITER ;

select informacionconstructora.ContadorTrabajadores(10) as NumeroTrabajadores

```

NumeroTrabajadores
3

```

select informacionconstructora.ContadorTrabajadores(8) as NumeroTrabajadores

```

NumeroTrabajadores
4

14. Cree un procedimiento con el punto 10.7


```

use informacionconstructora;

DELIMITER \\\
create procedure pa_ListaPersona_por_cargo(tipoempleo varchar(20))
begin
    select tablaobra.IdOBra as Idobra,
           tablaobra.Nombe as Obra,
           count(tablaobra_has_empleado.IdEmpleado) as Empleados,
           tipoempleo.NombreTipoEmpleado as TipoEmpleado
    from tablaobra,tablaobra_has_empleado,tipoempleado,empleado
    where tablaobra.IdOBra = tablaobra_has_empleado.IdOBra
    and tablaobra_has_empleado.IdEmpleado = empleado.idEmpleado
    and empleado.IdTipoEmpleado = tipoempleado.idTipoEmpleado
    and tipoempleado.NombreTipoEmpleado = tipoempleo
    group by tablaobra.IdOBra;

end\\
DELIMITER ;

call pa_ListaPersona_por_cargo('jefe de obra')

```

Idobra	Obra	Empleados	TipoEmpleado
1	Casa 278a	1	Jefe de obra
6	Parque Buenos aires	1	Jefe de obra
10	Iglesia de Marco	2	Jefe de obra
2	X100	1	Jefe de obra
7	Factory B	1	Jefe de obra
9	Almacen IB	2	Jefe de obra
3	Edificio 23B	1	Jefe de obra
5	Fabrica de carros II	1	Jefe de obra
8	Casa 13d	1	Jefe de obra
4	Fabrica de carros I	1	Jefe de obra

```

call pa_ListaPersona_por_cargo('obrero')

```

Idobra	Obra	Empleados	TipoEmpleado
1	Casa 278a	3	Obrero
3	Edificio 23B	3	Obrero
6	Parque Buenos aires	3	Obrero
7	Factory B	3	Obrero
9	Almacen IB	2	Obrero
8	Casa 13d	3	Obrero
2	X100	3	Obrero
10	Iglesia de Marco	1	Obrero
5	Fabrica de carros II	3	Obrero
4	Fabrica de carros I	3	Obrero

Con este procedimiento se facilita mucho la consulta, ya no necesito repetir el mismo código para consultar lo pedido en el punto 7.

15. Cree un disparador que cuente la cantidad de personal asignado a una obra según el estado del empleado (activo e inactivo) y cambie automáticamente el estado de la obra para que sea posible continuar su ejecución.

16. Todos los puntos anteriores deben quedar registrados en logs sobre Logs de la Base de Datos creados por un trigger y automáticos por los log de la base de datos.

```
set global general_log=ON;
```

El archivo con el historial que genera este comando de creación de logs se encuentra en la carpeta compartida, donde se encuentra este mismo archivo, esto debido a que la tabla generada es de aproximadamente 1000 filas. Se encuentra como: *"historial logs"* Esta en formato de CVS.