HECTOR GARCIA EXAMEN

1-

USE exam;

alter table CLIENTE

add ch\_dni VArchar (9) CHECK (dni like '[0-9]','[0-9]','[0-9]','[0-9]','[0-9]','[0-9]','[0-9]','[0-9]','[a-z]'));

CORRECCION

:

USE exam;

alter table CLIENTE

add CONSTRAINT ch\_dni VArchar (9) CHECK (dni like ('[0-9] [0-9] [0-9] [0-9]','[0-9] [0-9] [0-9] [0-9] [a-z]'));

USE exam;

alter table CLIENTE with no check

add CONSTRAINT ch\_dni VArchar (9) CHECK (dni like ('[0-9] [0-9] [0-9] [0-9]','[0-9] [0-9] [0-9] [0-9] [a-z]'));

2-

USE exam;

ALTER TABLE habitacion

ADD c\_ocupada VARCHAR(2) CHECK (c\_ocupada IN ('si','no') );

CORRECCION

:

USE exam;

ALTER TABLE habitacion

ADD CONSTRAINT c\_ocupada VARCHAR(2) CHECK (c\_ocupada IN ('si','no') );

SE PODIA HACER TAMBIEN CON UPDATE SET CASE WHEN ELSE

3-

USE exam;

ALTER TABLE cliente

ADD CONSTRAINT u\_dni UNIQUE;

La solución es añadirle una restricción de tipo UNIQUE para que cada vez que se ingrese un nuevo cliente , no se pueda meter un dni duplicado o que cada vez que haya que operar con ese dni no se cometan errores por usar un dni duplicado.

CORRECCION

:

USE exam;

CREATE UNIQUE INDEX UX\_Dni ON cliente(dni);

4-

USE exam;

SELECT codhot, nomhot ,categoria, ciuhot, fechainaug INTO Hoteles\_Inauguracion

FROM hotel

WHERE fechainaug BETWEEN '1940-01-01' AND GETDATE();

CORRECCION

:

USE exam;

ALTER TABLE hotel

ADD CONSTRAINT ck\_fechainaug CHECK (fechainaug >= '01/01/1940' AND fechainaug >= getdate());

5- USE exam;

SELECT c.nombre, c.apellido , Tipo =

CASE

WHEN hab.superficie > 25 THEN 'Grande'

WHEN hab.superficie > 20 AND hab.superficie < 25 THEN 'Mediana'

WHEN hab.superficie < 20 THEN 'Pequeña'

ELSE 'ninguna'

END

FROM hotel AS h, estancia AS e, cliente AS c, habitacion as hab

CORRECCION

:

USE exam;

SELECT c.nombre, c.apellido , Tipo =

CASE

WHEN hab.superficie > 25 THEN 'Grande'

WHEN hab.superficie > 20 AND hab.superficie < 25 THEN 'Mediana'

ELSE 'Pequeña'

END AS 'TIPO'

FROM hotel AS h, estancia AS e, cliente AS c, habitacion as hab

where e.codcli = c.codcli AND numhab numhab fecha salida is null and cidudadclienete = 'logroño'

6-

USE exam;

UPDATE habitacion SET precionoche = (precionoche/100)\*16

WHERE habitacion.numhab = (SELECT estancia.numhab FROM estancia WHERE estancia.codhot = (SELECT hotel.codhot FROM hotel

WHERE hotel.categoria > 4 AND ciuhot LIKE 'Logroño') );

CORRECCION

:

USE exam;

UPDATE habitacion SET precionoche = (precionoche/100)\*16

WHERE codhot IN (SELECT hotel.codhot FROM hotel

WHERE hotel.categoria > 4 AND hotel.ciuhot LIKE 'Logroño') ;

7- USE exam;

CREATE VIEW La\_Cuenta AS SELECT c.nombre, c.apellido

,((DATEDIFF(d,fechaentrada, GETDATE())\*hab.precionoche) AS PrecioAcumulado

FROM hotel AS h, estancia AS e, cliente AS c, habitacion as hab

WHERE c.codcli IN (SELECT e.codcli FROM estancia WHERE fechasalida IS NULL

AND e.numhab IN (SELECT hab.numhab FROM habitacion));

O

USE exam;

CREATE VIEW La\_Cuenta AS SELECT cliente.nombre, cliente.apellido ,estancia.codcli

,((DATEDIFF(d,fechaentrada, GETDATE())\*hotel.precionoche) AS PrecioAcumulado

FROM cliente, hotel

INNER JOIN estancia ON estancia.codcli = cliente.codcli AND

estancia.codcli IN (SELECT estancia.codcli FROM estancia WHERE fechasalida IS NULL AND estancia.numhab IN (SELECT habitacion.numhab FROM habitacion));

CORRECCION

:

USE exam;

CREATE VIEW La\_Cuenta AS SELECT c.nombre, c.apellido

,hab.precionoche \* (DATEDIFF(d,estancia.fechaentrada, GETDATE()) AS Precio\_Estancia

FROM hotel AS h, estancia AS e, cliente AS c, habitacion as hab

WHERE c.codcli = estancia.codcli AND estancia.fechasalida IS NULL

AND e.numhab = hab.numhab;

8-

USE exam;

CREATE TABLE EMPLEADOS(

id\_empleado INT PRIMARY KEY NOT NULL ,

nombre\_empleado VARCHAR (50) NOT NULL,

apellido\_empleado VARCHAR (50) NOT NULL,

codhot SMALLINT NOT NULL

CONSTRAINT f\_codhot FOREIGN KEY REFERENCES hotel(codhot)

CONSTRAINT ch\_id\_empleado CHECK (id\_empleado > 3)

)

USE exam;

alter TABLE estancia

ADD Empleado\_habitacion INT CONSTRAINT f\_EMP\_HAB FOREIGN KEY REFERENCES EMPLEADOS(id\_empleado );

USE exam;

INSERT INTO EMPLEADOS (id\_empleado, nombre\_empleado, apellido\_empleado, codhot)

VALUES (4, 'Juan','Perez',3 );

CORRECCION

:

USE exam;

CREATE TABLE EMPLEADOS(

id\_empleado INT IDENTITY (3,1) PRIMARY KEY NOT NULL ,

nombre\_empleado VARCHAR (50) NOT NULL,

apellido\_empleado VARCHAR (50) NOT NULL,

codhot SMALLINT NOT NULL

)

Alter table estancia

Add constraint fk\_cod\_hotel fk cod empleado FOERIGN KEY on deletwe no action

USE exam;

INSERT INTO EMPLEADOS (id\_empleado, nombre\_empleado, apellido\_empleado, codhot)

VALUES (4, 'Juan','Perez',’ritz’ );

9-

USE exam;

go

CREATE LOGIN Maria WITH PASSWORD '12345'

GO

CREATE USER Maria FOR LOGIN Maria ON dbo.exam

GO

GRANT SELECT(numhab, fechaentrada, fechasalida) ON Estancia TO Maria

CORRECCION

:

USE exam;

go

CREATE USER Maria WITHOUT login

GRANT SELECT(numhab, fechaentrada, fechasalida) ON Estancia TO Maria

EXECUTE AS USER 'Maria'

GO

SELECT numhab, fechaentrada, fechasalida FROM estancia;

10-