-- 1. Haz una consulta que haga que para campo DNI da la tabla CLIENTE, que es un campo de tipo cadena de caracteres, se verifique que está formado por 8 números y una letra minúscula (0,5 ptos.) (0.25)

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]'));

Si existiera datos en ese campo que no cumplieran la restricción que voy a crear pero necesito seguir almacenándolos. ¿Qué harías para que permaneciesen en ese campo de la base de datos? Justifica la respuesta. (0,5 ptos.)

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]'));

Nota: no va la coma en el medio

--2. Añade el campo con nombre ocupada a la tabla HABITACION, que será un campo de tipo cadena de caracteres con una longitud de 2 caracteres y con la única posibilidad de ser sí o no. (0,5 pto)

Actualiza los valores de ese campo mediante una consulta. (1 pto)

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 .Es frecuente consultar información de los clientes introduciendo su dni. Indica de qué forma se podría mejorar la base de datos y escribe la instrucción correspondiente. Justifica la respuesta. Con la solución que adoptes haz que el dni de cada cliente sea único en la tabla cliente. (0,5 pto)

USE exam;

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

-- 4. Realiza una consulta que haga que la fecha de inauguración de todos los hoteles este comprendida entre el 01/01/1940 y la fecha actual. (0,5 pto)

USE exam;

ALTER TABLE hotel

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

-- 5. Realiza una consulta que muestre una lista de los clientes de "Logroño" que estén actualmente alojados en algún hotel, se tiene que mostrar el nombre del cliente, apellido del cliente y un tercer campo llamado Tipo cuyo contenido atienda a la siguiente clasificación: "Grande" si está alojado en una habitación de superficie superior a 25 metros, "mediana" si la superficie esta entre 20 y 25 metros y "pequeña" si la superficie de la habitación es menor de 20 metros. (1 pto)

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 c.numhab = h.numhab AND fecha\_salida is null and c.ciudad\_cliente = 'logroño';

-- 6. Realiza una consulta que añada el impuesto IVA (16%) al campo ***precionoche*** de las habitaciones cuyo hotel sea de categoría 4 estrellas o estén situadas en Logroño. (0,75 pto.)

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. Crea una vista llamada La\_Cuenta que muestre el nombre y apellido del cliente y precio que lleva acumulado, de aquellos clientes que están actualmente alojados en algún hotel. (1,25 ptos.)

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. Crea una tabla empleados, que contenga tres campos: id\_empleado, nombrempleado, apellempleado, hotel (código del hotel en qué está contratado). El id\_empleado se irá asignando a medida que se dan de alta los empleados en el sistema de manera consecutiva. Empezarán a partir del 3. El hotel al que está asignado no puede estar vacío. Elige el tipo de datos más conveniente.

Añade en la tabla ESTANCIA un campo que indique el empleado que atendió la habitación durante la estancia. Establece las restricciones que consideres que son necesarias para mantener la integridad de la base de datos tras la creación de la nueva tabla.

Insertar el empleado Juan Pérez que trabaja en el hotel Rittz (1,5 pto.)

Nota: las vistas no se pueden crear si no hay campos.

USE exam;

CREATE TABLE EMPLEADOS(

id\_empleado INT IDENTITY 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',’ritz’ );

Cuidado con este, las FOREIGN KEYS EN SU CORRECCION ME LAS PUSO COMO BUENAS PERO ELLA LAS HIZO ASÍ:

USE exam;

alter TABLE estancia

ADD CONSTRAINT ch\_id\_empleado CHECK (id\_empleado > 3) ON DELETE NO ACTION;

USE exam;

alter TABLE estancia

ADD CONSTRAINT f\_codhot FOREIGN KEY REFERENCES hotel(codhot) ON DELETE NO ACTION;

--9. Crea un usuario María asociado a la base de datos exam. Otorga a ese usuario permisos para poder ver solo el número de habitación, fecha de entrada y fecha de salida de una estancia. (0,5 ptos.)

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. Describe qué es una transacción en SQL. Cita y explica qué cuatro propiedades debe cumplir.

¿Tiene algo que ver con los niveles de aislamiento en una base de datos? Explícalo (1,5 ptos.)