

EVALUACIÓN	Obligatorio	GRUPOS	Todos	FECHA	Septiembre 2017
MATERIA	Bases de Datos 2				
CARRERA	Analista en Tecnologías de Información / Analista Programador				
CONDICIONES	<p>- Entrega: 11/12/2017</p> <p>- Puntos: <u>Máximo</u>: 40. <u>Mínimo</u>: 1.</p> <p>- <u>Horario de Bedelía para entregas</u>: de lunes a jueves de 8:00 a 12:00 y de 15:00 a 20:00 hs.</p> <p><u>IMPORTANTE</u></p> <p>- Los grupos deben estar conformados por hasta un máximo de dos personas.</p> <p>- Inscribirse (sacar la "<u>boleta de entrega</u>").</p> <p>- Entregar en carpeta con elástico.</p> <p>- Etiquetar documentación y CDs con nombre, n° estudiante, carrera, grupo, materia y docente.</p>				

La universidad tiene un sistema para registrar y gestionar los trabajos de investigación producidos por sus miembros (investigadores/estudiantes), modelado de la siguiente manera:

Universidad (nombre, pais, ciudad, telefono)

Investigador (idInvestigador, idUniverdidad, nombre, mail, telefono, carrera, nivelInvestig, cantTrabPub)

Trabajo (idTrab, nomTrabajo, descripTrab, tipoTrab, fechaInicio, linkTrab, lugarPublic)

Tags (idtag, palabra)

TTags (idTrab, idTaq)

TAutores (idTrab, idInvestigador, rolinvestig)

Referencias (idTrab, idTrabReferenciado)

Lugares (idLugar, nombre, tipoLugar, nivelLugar, año, mes, diaIni, diaFin, link, universidad)

Universidad

En esta tabla se guarda la información de las universidades registradas en el sistema, se identifican por su nombre. Todos los datos de la universidad son requeridos.

Investigador

En esta tabla se guarda la información de los Investigadores de las universidades. Estos se identifican por un autonumérico. No hay investigadores que no estén vinculados a una universidad, y esta es única, es decir no cambian de universidad.

Todo investigador tiene un nivel (nivelInvestig) que toma valores: estudiante grado (EGrado), estudiante maestria (EMaestria), estudiante doctorado (EDoctor), y Doctor.

El mail del investigador es una clave alterna. Y el teléfono se registra solo en caso de que el investigador quiera registrarlo.

También se registra la cantidad de trabajos publicados que tiene el investigador.

Trabajo

En esta tabla se persisten los datos de los trabajos producidos por los investigadores. Los trabajos se identifican por un alfanumérico cuyo primer carácter indica el tipo de Trabajo (P->poster, A->artículo, C->capítulo, O->Otro). Los demás caracteres de la clave corresponden a un número secuencial.

Todo trabajo tiene un nombre, una descripción de hasta 200 caracteres, la indicación del tipoTrab (poster, artículo, capítulo, otro), la fecha de inicio del mismo, y si el trabajo tiene página se puede registrar el link de la misma (linktrab).

Los trabajos publicados deben indicar el lugar de publicación.

Tags

En esta tabla se registran todos los posible tags (palabras claves) que indican los temas que tratan los trabajos. Las palabras claves se identifican con un número impar autonumérico.

TTags

En esta tabla registra para cada trabajo cuáles son sus palabras claves (tags).

TAutores

En esta tabla se registran los autores de los trabajos y el rol del autor en el trabajo. Los roles pueden ser: autor-ppal, autor-sec, autor-director.

Referencias

En esta tabla se registran las relaciones de referencias entre los trabajos, es decir las referencias usadas en cada trabajo de investigación. Un trabajo no puede tenerse a si mismo como referencia.

Se considera una referencia externa a aquella que vincula dos trabajos que no tienen autores en común.

Lugares

En esta tabla se guarda información de los lugares en los cuales se puede publicar un trabajo de investigación. Los lugares se identifican por un número, tienen un nombre que no puede repetirse y e de hasta 250 caracteres, y un link al sitio del evento que lo promueve.

Los tipos de lugares posibles son: Congresos, Revistas, Libros (TipoLugar solo puede tomar estos valores). Cada lugar tiene un nivel según la exigencia y especialización de los mismos, *nivelLugar* es un número que puede tomar valores entre 1 y 4 siendo 1 los básicos y los 4 los más especializados y exigentes.

También se guarda la universidad principal/afitritona, así como el año, mes y día en que se publicó la revista o libro (el día se registra en el campo dialni), y en caso de un congreso también se guarda día en que inicio y finalizo el congreso.

Se Pide:

1. Crear las restricciones de integridad que surjan del análisis de la letra, sobre el script de creación de tablas proporcionado (mediante ALTER de las estructuras dadas).
(6 puntos)
2. Creación de índices que considere puedan ser útiles para optimizar las consultas (según criterio establecido en el curso).
(2 puntos)
3. Ingreso de un juego completo de datos de prueba (será más valorada la calidad de los datos más que la cantidad. El mismo debería incluir ejemplos que deban ser rechazados por no cumplir con las restricciones implementadas).
(2 puntos)
4. Crear procedimientos o funciones según corresponda:
 - a. Crear una función que dada una universidad devuelva el último trabajo publicado por esta. Si hay más de uno devolver uno cualquiera.
 - b. Crear una función almacenada que reciba como parámetro un trabajo y devuelva la cantidad de referencias externas que tiene.
 - c. Crear una función que reciba dos investigadores y devuelva la cantidad de trabajos publicados en los cuales ambos investigadores fueron autores y alguno de los dos o los dos fueron autores principales.
 - d. Crear una función o procedimiento, según considere/corresponda, que dado un investigador actualice el campo cantidad de trabajos publicados registrados de la tabla Investigador, y devuelva una indicación de si la cantidad que estaba registrada (antes de la actualización) era correcta o no.
 - e. Crear una función que dado dos años (año1 y año2) y una palabra clave devuelva la cantidad de trabajos publicados en libros o revistas en el rango de años y que trata el tema indicado por la palabra clave.
 - f. Crear una función que reciba un tipo de trabajo y devuelva un nuevo identificador para dicho trabajo.
 - g. Crear un procedimiento que dado una universidad devuelva los datos del congreso en el cual público su ultimo Artículo, y algún tipo de indicación de si existe o no un artículo con esas condiciones. Suponemos que no hay congresos que traten los mismos temas y que se solapen en el tiempo.
 - h. Crear un procedimiento o función, según considere/corresponda, que dado un tipo de trabajo (Artículo, Poster, etc.) devuelva la cantidad de investigadores que tengan más de 5 trabajos del tipo indicado, y que tengan la maxima cantidad de publicaciones.

(8 puntos)

5. Crear disparadores necesarios para realizar las siguientes actividades:

- a.** Crear un trigger que al insertar un trabajo asegure las restricciones identificadas, y en caso de que no se asignó una fecha de inicio el trigger asigne el primero de enero del año actual como fecha de inicio del trabajo.
En el caso de considerarse insert de a un trabajo por vez, debe asegurarse de que la operación se haga solo en ese caso.
- b.** Crear un trigger que controle que solo puedan eliminarse trabajos no publicados que iniciaron hace más de 2 años. Eliminando todos los datos de la base de datos que considere necesarios asociados a dichos trabajos.
Debe considerarse eliminaciones múltiples.
- c.** Crear un trigger solo no permita insertar dos trabajos uno como referencia del otro si tienen algún tema (palabras claves) en común.
- d.** Crear un trigger que audite las inserciones y las actualizaciones de los lugares que tienen algún trabajo publicado, registrando en alguna tabla auxiliar el lugar que se quiere actualizar/insertar, la operación (INS o UPD), el usuario y la fecha hora de la operación.

(8 puntos)

6. Resolver mediante consultas SQL:

- a.** Mostrar, para cada trabajo publicado con mas de 3 autores, el identificador del trabajo, nombre del mismo, la cantidad de autores que tiene, y lugar donde se publico (id y nombre).
- b.** Obtener la lista de palabras claves que aparecen en trabajos sobre 'BASE DE DATOS' (palabra clave) publicados en revistas del año actual. La salida debe aparecer ordenado alfabéticamente, sin repetidos.
- c.** Mostrar los datos de las universidades y el link a los congresos, de las universidades que han realizado más de 2 congresos de nivel 4, en los ultimos 5 años, con mas de 20 trabajos publicados en esos congresos.
- d.** Obtener para cada investigador el ultimo trabajo que inicio en el cual fue/es autor principal.

- e. Para cada investigador mostrar, su identificación, nombre, nombre de la universidad a la que pertenece, y la cantidad de trabajos suyos publicados en lugares de nivel 1, de nivel 2, de nivel 3 y de nivel 4, en los últimos 5 años, en la carrera de Ingeniería.
- f. Para la universidad "ORT" mostrar el identificador de sus investigadores que tienen algún trabajo en el año actual y la cantidad de trabajos publicados en congresos de nivel 4. En caso de investigadores sin trabajos publicados en estos congresos pero con trabajos en proceso el año actual deben aparecer en el resultado.
- g. Mostrar para cada universidad que tiene trabajos publicados, los datos del último trabajo publicado. Solucionar usando la función a).
- h. Eliminar las palabras claves no usadas en los trabajos.

(8 puntos)

7. Vistas:

- a. Realizar una vista que muestre lista de Congresos y para cada uno la cantidad de trabajos publicados que no tienen autores que sean investigadores de la universidad anfitriona del congreso.
- b. Realizar una vista que muestre para cada Investigador, para cada tipo de trabajo la fecha de inicio del primer y último trabajo. Todos los investigadores deben aparecer en el resultado, aunque no tengan trabajos que cumplan las condiciones.

(4 puntos)

8. Informe que describa la solución entregada, justificándola. Si existieran casos en los que no se pudiera cumplir con la letra, explicar por qué y si es posible plantear una solución alternativa. (NOTA: Se valorará la prolijidad y cumplimiento del documento 302 y 303 para entrega de trabajos finales).

(2 puntos)

Se debe entregar:

- 1. En CD:
 - a. Script con las restricciones de integridad creadas sobre el script de creación de tablas, índices, y el ingreso de datos de prueba.
 - b. Script con la creación de disparadores.
 - c. Script con la creación de funciones y procedimientos.
 - d. Script con la resolución de las consultas y vista.
 - e. Versión digital del informe.
- 2. Impresión de los scripts antes mencionados junto con el informe.

Consideraciones generales:

1. Los docentes de la materia cumplirán el rol de usuario final del producto a los efectos de evacuar las dudas que puedan surgir a los estudiantes en detalles que no estén incluidos explícitamente en la letra. Independientemente de esto, los alumnos podrán investigar sobre sistemas existentes, así como aportes basados en su propia experiencia o relevamiento con terceros para enriquecer la solución a los problemas planteados siempre que no contradiga lo explicitado en la letra. Cualquier agregado deberá documentarse claramente en la solución y será considerado positivamente en la evaluación. Modificaciones de la letra que puedan surgir durante el curso, serán publicadas en aulas y deberán considerarse en la entrega final.
2. La corrección del obligatorio se hará en base a la estructura entregada junto con la letra del mismo, por lo que los puntos desarrollados deben ser testeados sobre esta estructura. Soluciones a los puntos del obligatorio que no ejecuten correctamente sobre la estructura proporcionada serán evaluados como incorrectos. Esta consideración también aplica para los procedimientos y funciones, que deben tener el mismo nombre, cantidad, tipo y orden de parámetros que se define en esta letra.
3. Durante la última semana (4 al 10 de diciembre) los docentes no contestarán dudas del Obligatorio por ningún medio. Esta consideración intenta evitar que los alumnos dejen la implementación del obligatorio para último momento. Se insta a los estudiantes a desarrollar el obligatorio durante el transcurso del semestre para entregar un trabajo de calidad.

Anexo 1 - Script de creación de tablas

```
create table Universidad (  
    nombre varchar(100),  
    pais varchar(50) not null,  
    ciudad varchar(50) not null,  
  
    )  
go  
  
create table Investigador (  
    idInvestigador int,  
    nombre varchar(100) not null,  
    mail varchar(200),  
    telefono varchar(20),  
    carrera varchar(50),  
    nivelInvestig varchar(20) not null,  
    cantTrabPub int )  
go  
  
create Table Trabajo (  
    idTrab int not null,  
    nomTrab varchar(100) not null,  
    descripTrab varchar(10) not null,  
    tipoTrab varchar(20) not null,  
    fechaInicio date not null,  
    linkTrab varchar(200),  
    lugarPublic int)  
go  
  
create table Tags (  
    idtag int,  
    palabra varchar(50))  
go  
  
create Table TTags (  
    idTrab varchar(10) not null,  
    idTag int not null  
    )  
go  
  
create Table TAutores (  
    idTrab varchar(10) not null,  
    idInvestigador int not null,  
    rolinvestig varchar(3))  
go
```



```
create Table Referencias (  
    idTrab varchar(10) not null,  
    idTrabReferenciado varchar(10) not null  
)  
go  
  
create Table Lugares (  
    idLugar int primary key,  
    nombre varchar(100) not null,  
    nivelLugar int ,  
    año int not null,  
    mes int not null,  
    diaIni int,  
    diaFin int,  
    link varchar(200),  
    universidad varchar(50) )  
go
```