

# Transacciones y propiedades ACID

**Alex Di Genova**

**28/06/2022**

# Contenidos

- Repaso (Vistas, Indexación y Optimización)
- Propiedades ACID

# Notas y revisión control II

# Revisión control II

## Respuestas

The screenshot shows a web browser window displaying a university portal. The page is titled "Bases de Datos" and "COM3101-1 - Primer Semestre 2022". The user is logged in as "Alex Di Genova". The main content area shows a PDF document titled "Pauta Control II.pdf" with a download button. The PDF content includes three SQL exercises and their solutions:

6. **Base de datos Chinook:** Implemente un código en SQL que permita seleccionar el nombre (firstname), la edad, ciudad(city) y país(country) de todos los empleados (Employee) mayores a 60 años. (5 puntos) Hint : datetime("2022-06-14").

```
select firstname, datetime("2022-06-14")-BirthDate as edad, city, country
from employee
where edad > 60
limit 10;
```

7. **Base de datos Empresa:** seleccionar todos los productos (product) que son provistos (supplier) por la compañía (Companyname) "Exotic Liquids" (5 puntos). Hint: utilice una subconsulta.

```
select * from Product
where SupplierId in (select id from supplier where Companyname == "Exotic Liquids")
```

8. **Base de datos Empresa:** seleccione todos los productos cuyo precio de unidad (UnitPrice) esta sobre el promedio (7 puntos). Hint: utilice una subconsulta y función promedio - AVG.

```
select * from OrderDetail where UnitPrice > (select AVG(UnitPrice)
from OrderDetail) limit 10;
```

9. **Base de datos Empresa:** liste todos los empleados (Employee) que son de (Cityname) "Seattle".

Below the PDF, there is a description of the document:

**Descripción** Estimad@s,  
Adjunto la pauta del control II para su revisión.  
Saludos Cordiales  
Alex Di Genova

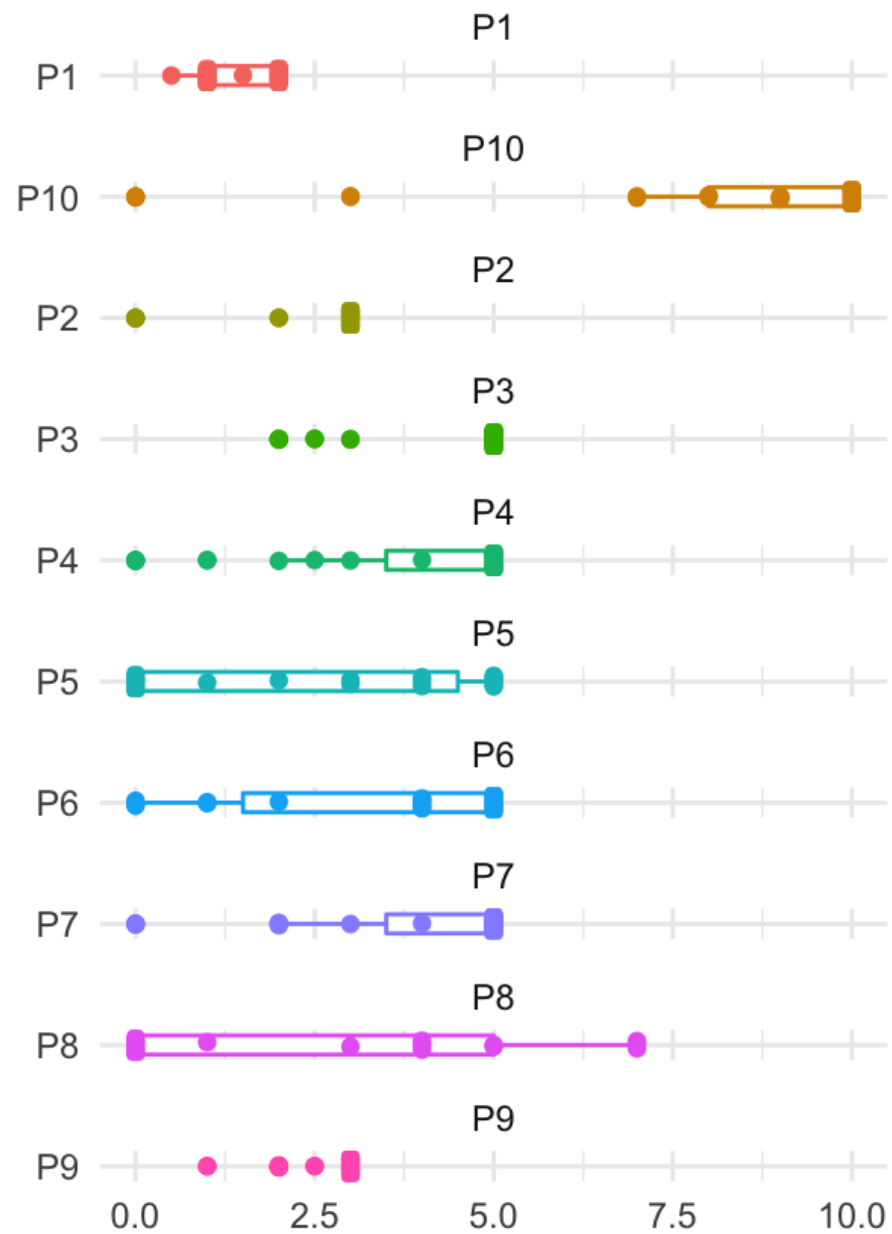
**Categoría** controles

**Año** 2022

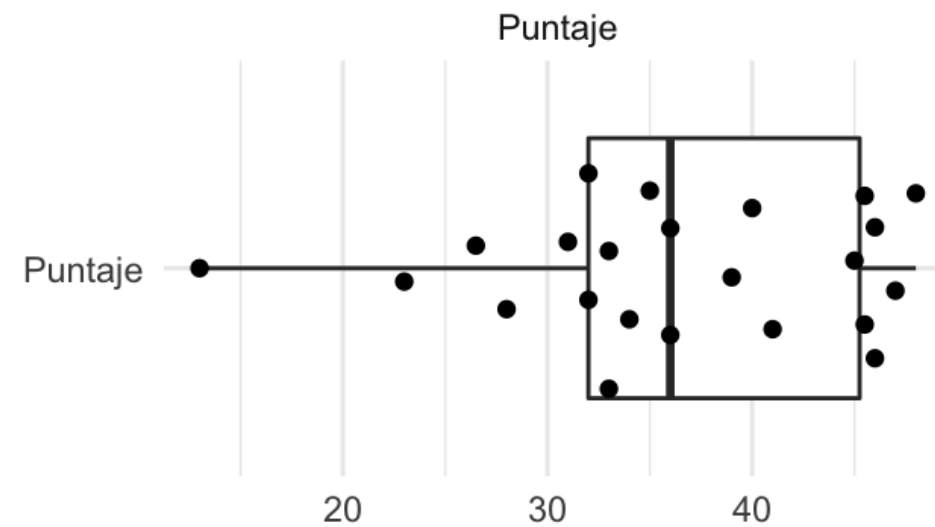
# Notas

## Control II

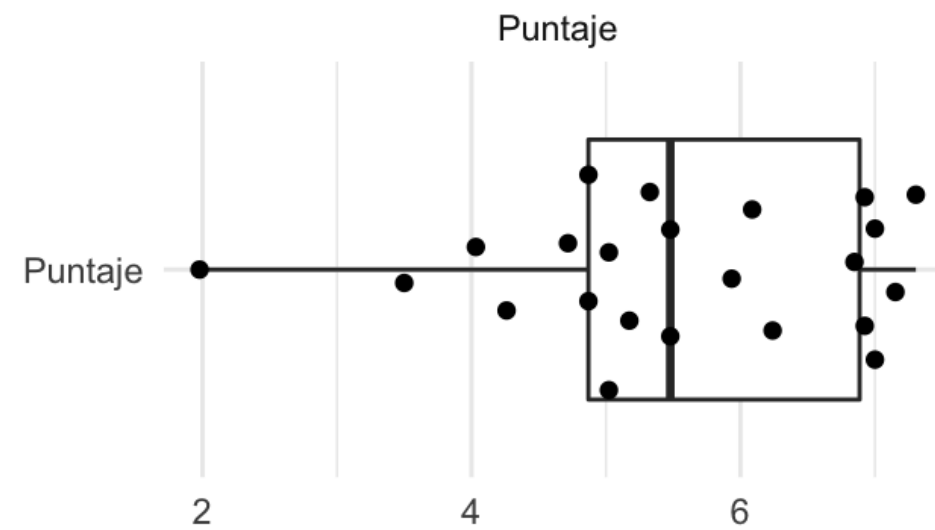
Preguntas



Puntaje



Notas



- $\text{Nota} = \text{Puntaje} / 46 * 7$
- Promedio = 5,5
- En promedio incremento de 0.6 Decimas.

# Repaso

# Vistas

## Ejemplo Join Anidado

- Usos:
  - Para almacenar consultas frecuentes o complejas.
  - Para crear versiones de tablas mas amigables al usuario (tiempo y fechas).

- Crear una vista de un join anidado

```
create view join_album_artista_track as select * from
artist natural join album join track using (albumid)
```

- Como desplegamos los elementos de la lista?

```
select * from join_album_artista_track limit 10;
```

- Como eliminamos la vista?

```
drop view join_album_artista_track;
```

```
[7] 1 %%sql
2 create view join_album_artista_track as select * from artist natural join album join track using (albumid)

* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook_Sqlite.sqlite
Done.
[]
```

```
[8] 1 %%sql
2 select * from join_album_artista_track limit 10;
```

\* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook\_Sqlite.sqlite  
Done.

ArtistId	Name	AlbumId	Title	TrackId	Name:1	MediaTypeId	GenreId	Composer
1	AC/DC	1	For Those About To Rock We Salute You	1	For Those About To Rock (We Salute You)	1	1	Angus Young, Malcolm Young, I
2	Accept	2	Balls to the Wall	2	Balls to the Wall	2	1	None
2	Accept	3	Restless and Wild	3	Fast As a Shark	2	1	F. Baltes, S. Kaufman, U. Dirks Hoffman
2	Accept	3	Restless and Wild	4	Restless and Wild	2	1	F. Baltes, R.A. Smith-Diesel, S. Dirkschneider & W. Hoffman
2	Accept	3	Restless and Wild	5	Princess of the Dawn	2	1	Deaffy & R.A. Smith-Diesel
1	AC/DC	1	For Those About To Rock We Salute You	6	Put The Finger On You	1	1	Angus Young, Malcolm Young, I
1	AC/DC	1	For Those About To Rock We Salute You	7	Let's Get It Up	1	1	Angus Young, Malcolm Young, I
1	AC/DC	1	For Those About To Rock We Salute You	8	Inject The Venom	1	1	Angus Young, Malcolm Young, I
1	AC/DC	1	For Those About To Rock We Salute You	9	Snowballed	1	1	Angus Young, Malcolm Young, I
1	AC/DC	1	For Those About To Rock We Salute You	10	Evil Walks	1	1	Angus Young, Malcolm Young, I

```
1 #drop view
2 %%sql
3 drop view join_album_artista_track;
```

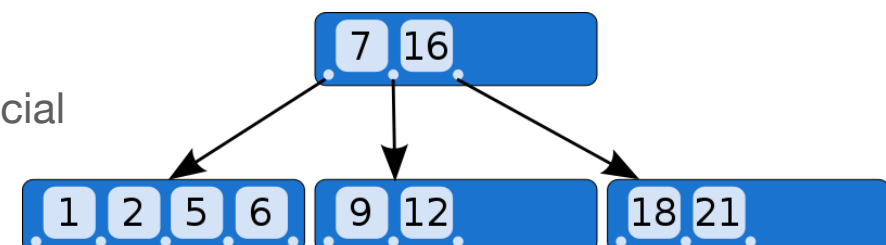
\* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook\_Sqlite.sqlite  
Done.  
[]

# Indices

## Optimización de consultas

- Sin índices, el motor de base de datos está obligado de revisar las tablas completas en cada consulta.
  - Los JOINS son costosos computacionalmente.
  - La idea es encontrar filas sin la necesidad de revisar toda la tabla.
- Cada índice agrega una carga adicional a INSERT, UPDATE y DELETE.
  - Debemos evaluar donde colocar los índices.
- Los índices no se pueden crear en Vistas.
- Como almacena los datos el motor SQL?
- Por defecto, cada tabla es almacenada utilizando una estructura indexada (B-Tree SQLite).
  - A medida que se insertan filas en el B-Tree, las filas se ordenan, organizan y optimizan, de modo que una fila con un ROWID específico y conocido se puede recuperar de manera relativamente directa y rápida.

Permite: búsquedas, inserciones, deletaciones y acceso secuencial  
 $\text{Time}(n) = O(\log n)$



- Cuando creamos un índice, el sistema de base de datos crea otro B-Tree para almacenar los datos del índice.
- El nuevo B-Tree se ordena y organiza usando la columna o columnas que se especifican en la definición del índice (! ROWID).



# Indices

## Ejemplo

- Creamos una tabla con 4 x 1000 numeros.

```
create table tbl (a,b,c,d);
INSERT INTO tbl (a, b,c,d)
VALUES
(84,39,78,79),
(182,39,67,153),
(83,166,143,188),
(145,205,380,366),
(317,358,70,303), ...
```

```
1 %%time
2 %%sql
3 select * from tbl where a=29238;
```

```
* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook_Sqlite.sqlite
Done.
CPU times: user 5.2 ms, sys: 0 ns, total: 5.2 ms
Wall time: 7.23 ms
  a      b      c      d
29238 3171  23996 48794
29238 297097 765679 213680
```

```
[12] 1 %%time
      2 %%sql
      3 create index idx_tbl_a_b ON tbl(a,b)
```

```
* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook_Sqlite.sqlite
Done.
CPU times: user 9.09 ms, sys: 32 µs, total: 9.13 ms
Wall time: 22.4 ms
[]
```

```
1 %%time
2 %%sql
3 select * from tbl where a=29238;
```

```
[>] * sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook_Sqlite.sqlite
Done.
CPU times: user 3.88 ms, sys: 0 ns, total: 3.88 ms
Wall time: 3.76 ms
  a      b      c      d
29238 3171  23996 48794
29238 297097 765679 213680
```

# Transacciones

# Transacciones

- DDL (Data Definition Language)
  - CREATE TABLE...
- DML (Data Manipulation Language)
  - SELECT ...
- TCL (Transaction Control Language)
  - Las transacciones son una parte fundamental de cómo las bases de datos relacionales protegen la integridad y confiabilidad de los datos que contienen.
  - Las transacciones se utilizan automáticamente en todos los comandos DDL y DML.
- Transacción?
  - Una transacción puede involucrar la actualización de un solo valor hasta un procedimiento que puede insertar múltiples valores en múltiples tablas (**update**).
    - Es la unidad básica de cambio en BD (Operaciones parciales no son permitidas)
  - Si dos clientes cambian el mismo registro al mismo tiempo?
    - (Control de concurrencia)
  - Si estamos realizando una transferencia entre bancos y se corta la luz. ¿Cuál es el estado de la base de datos?
    - (Durabilidad, recuperación, consistencia)

# Transacciones

## Realizar una transferencia de dinero

- Como podemos tranferir fondos de manera segura a otra cuenta bancaria.

Pasos	Cliente1	Cliente2
Chequear	X	-
Subtraer	X	-
Agregar	-	X



- Todos los pasos deberían tener éxito por completo, dando como resultado que el saldo se transfiera correctamente, o los tres pasos deberían fallar por completo, dando como resultado que ambas cuentas no se modifiquen.
- Cualquier otro resultado, donde un paso tiene éxito y el otro falla, conlleva a un error.

# Transacciones

## Realizar una transferencia de dinero

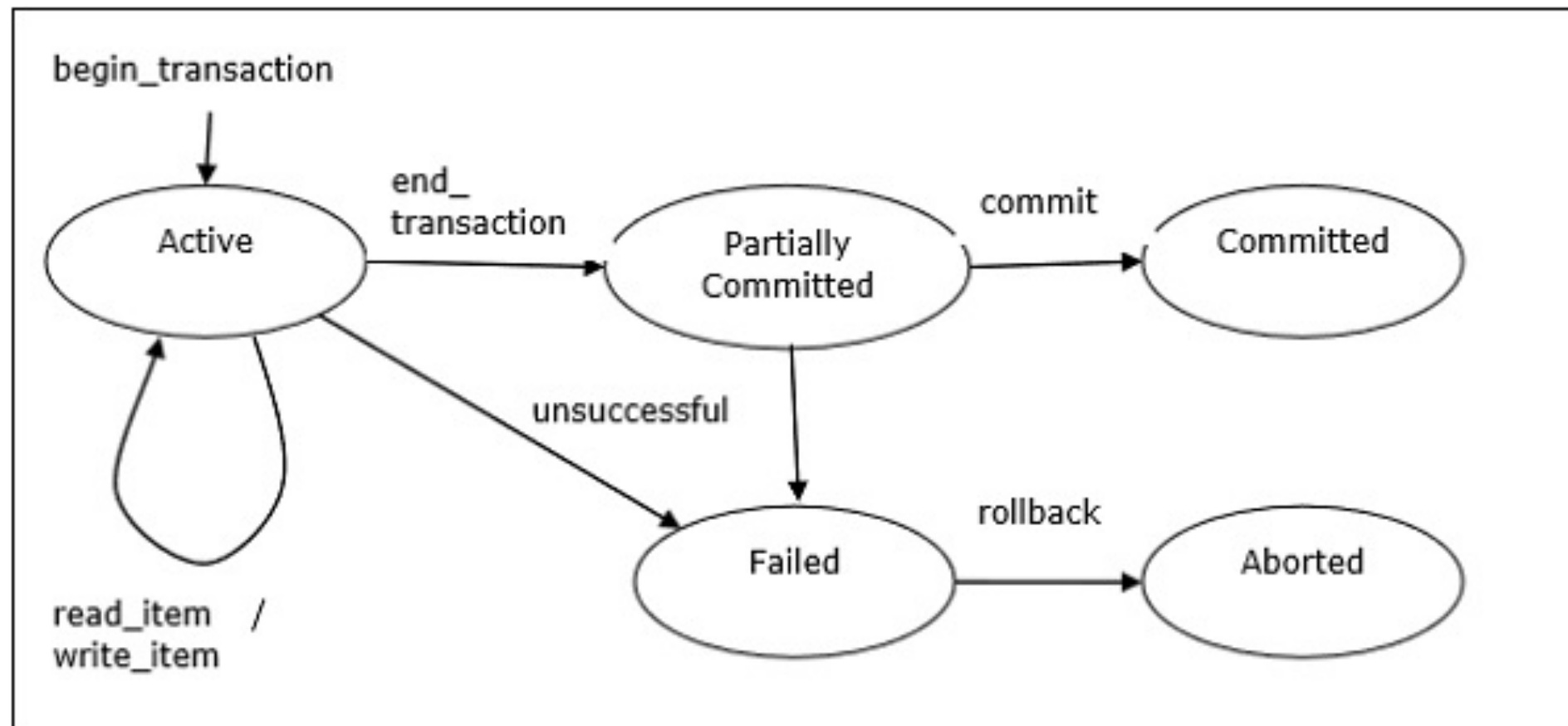
Pasos	Cliente1	Cliente2
Chequear	X	-
Subtraer	X	-
Agregar	-	X



- Como garantizar la transacción?
- Ejecutar las transacciones en orden serial:
  - Antes transacción : 1)copiamos DB; 2)cambiamos la copia; 3) si transaccion se ejecuta correctamente? 4) remplazamos la DB original o eliminamos la copia.
- Permitir concurrencia de transacciones independientes (mejor alternativa)
  - Que pasa si cliente1 intenta tranferir 2 veces al mismo tiempo?.
  - Concurrencia: problemas temporales(ok) y permanentes(bad) de consistencia en la DB.

# Transacciones

## Etapas de una transacción



- **BEGIN**
  - **COMMIT** (transacción ok) -> DBMS almacena todos los cambios
  - **ABORT** (transacción falla) -> todos los cambios se deshacen de modo que es como si el txn nunca se ejecutó en absoluto.

# Propiedades ACID

# Propiedades ACID

- El estándar para transacciones confiables y sólidas es la prueba ACID.
- ACID significa **Atómica, Consistente, Aislada y Duradera**
- **Atómica:** Una transacción debe ser atómica, en el sentido de que el cambio no se puede dividir en partes más pequeñas. **“Todo o nada”**
- **Consistente:** Suponiendo que una base de datos se inicia en un estado coherente, la aplicación de una transacción debe mantener la base de datos coherente. **“Me parece correcto”**
- **Aislada:** Los cambios no deben ser visibles para ningún otro sistema que acceda a la base de datos, ni deben integrarse en el registro permanente de la base de datos hasta que se confirme toda la transacción. **“Como si estuviera solo”**
- **Duradera:** Una vez que se devuelve un estado de éxito, no debería importar si el proceso se cancela, el sistema pierde energía o el sistema de archivos de la base de datos desaparece; al reiniciar, los cambios confirmados deben estar presentes en la base de datos. **“Sobrevivir a fallas”**
- **Las cuatro propiedades deben cumplirse para garantizar la integridad general de la base de datos.**



# Informaciones

- Control 3
  - Jueves 14 Julio.
- Control Recuperativo
  - Lunes 18 de Julio
    - Horario por definir en conjunto (encuesta ucampus)
    - Sala por informar.

# Consultas?

Consultas o comentarios?

Muchas gracias