

## Programa de Curso

Marzo de 2020

Profesores:	Adrián Soto Suárez assoto@uc.cl Juan Reutter jreutter@ing.puc.cl
Ayudante jefe cátedra	Tamara Cucumides tacucumides@uc.cl
Ayudante jefe proyecto	Kevin Jhonson kjjohnson@uc.cl
Clases:	miércoles, módulos 5 y 6. Sala A5 - B17
Horario de Ayudantías:	viernes, módulo 5. B12 - Sala B17

### Descripción

Las bases de datos forman parte del núcleo del desarrollo de aplicaciones comerciales modernas, y son indispensables para cualquier aplicación que requiera almacenar grandes volúmenes de datos, actualizarlos y consultarlos de manera eficiente. El propósito de este curso es introducir al alumno en el diseño y uso de los sistemas de bases de datos, específicamente bases de datos relacionales.

### Objetivo General

Durante el curso el alumno aprenderá la teoría detrás del modelo relacional, incluyendo el lenguaje SQL (hoy estándar para consultar bases de datos relacionales). Sabrá aplicar esta teoría al diseño y uso de bases de datos relacionales. Además, conocerá como interpretar y consultar bases de datos noSQL. Al final del curso el alumno podrá diseñar y manejar una base de datos relacional en un ambiente real. También conocerá los modelos teóricos usados en el estudio de las bases de datos, podrá analizar y comparar distintos esquemas y consultas de acuerdo a su utilidad práctica, e interpretar las ventajas y desventajas de los modelos de datos relacionales en comparación con modelos noSQL.

### Competencias e indicadores de desempeño

Competencia	Indicadores de Desempeño
1. Entender qué es un sistema de bases de datos, y conocer las ventajas y desventajas de los modelos de datos relacional y noSQL.	<ol style="list-style-type: none"><li>1. El alumno conoce la necesidad de implementar y usar sistemas de bases de datos.</li><li>2. Entiende la diferencia entre modelos de datos, esquema de datos e instancia de un esquema.</li><li>3. Distingue entre distintos modelos de datos, conoce que hay distintas alternativas, sus ventajas y desventajas.</li></ol>

2. Diseñar bases de datos relacionales para aplicaciones basadas en casos reales	<ol style="list-style-type: none"> <li>1. Crea un esquema de datos a partir de una descripción verbal.</li> <li>2. Abstrae y optimiza requerimientos mediante un modelo ER.</li> <li>3. Conoce como generar un esquema relacional a través de un modelo ER.</li> <li>4. Compara distintos esquemas de acuerdo a los requerimientos.</li> <li>5. Aplica la teoría de formas normales a la comparación y/o optimización de esquemas.</li> </ol>
3. Usar el lenguaje de consulta SQL para bases de datos relacionales	<ol style="list-style-type: none"> <li>1. Conoce la sintaxis y la semántica de SQL.</li> <li>2. Escribe consultas simples.</li> <li>3. Establece un plan lógico frente a una consulta, y lo expresa en SQL.</li> <li>4. Puede comprender el significado de una consulta compleja y establecer su plan lógico.</li> </ol>
4. Usar bases de datos para trabajar en problemas de ciencia de datos en entorno realista	<ol style="list-style-type: none"> <li>1. Importa y exporta datos desde y hacia distintos formatos.</li> <li>2. Usa consultas SQL para obtener transformaciones de datos que puedan responder hipótesis simples sobre datos.</li> <li>3. Usa consultas SQL para transformar datos de manera de poder usar técnicas estadísticas básicas.</li> </ol>
5. Comprender conceptos esenciales en teoría de bases de datos, especialmente el Álgebra Relacional (AR), y conoce como se usa para estudiar las limitaciones de las bases de datos	<ol style="list-style-type: none"> <li>1. Conoce la sintaxis y la semántica del Álgebra Relacional.</li> <li>2. Puede transformar consultas de SQL a AR, y vice-versa.</li> <li>3. Entiende el concepto consultas no expresables en SQL o AR, y puede dar ejemplos.</li> <li>4. Conoce el concepto de plan de consultas, sabe el papel del AR en la optimización de consultas.</li> </ol>
6. Conocer nociones básicas del paradigma noSQL, búsqueda de texto y herramientas de análisis de datos	<ol style="list-style-type: none"> <li>1. Conoce el paradigma key-value (JSON) el sistema MongoDB.</li> <li>2. Conoce los desafíos detrás de la búsqueda en textos.</li> <li>3. Realiza consulta simples a bases de datos orientadas a grafos.</li> <li>4. Conoce herramientas clásicas de análisis de datos.</li> </ol>

7. Comprender los desafíos y problemas más importantes de la industria de bases de datos actuales. Discutir y saber plantear soluciones de diseño e implementación de bases de datos.	<ol style="list-style-type: none"> <li>1. Describe la arquitectura de un sistema de bases de datos.</li> <li>2. Conoce y usa conceptos avanzados del lenguaje SQL, como vistas, transacciones, procedimientos almacenados, cursores y serialización.</li> <li>3. Implementa una plataforma Web para acceder a una base de datos, y mostrar el resultado de las consultas.</li> <li>4. Diseña soluciones para un caso real del problema de integración de datos.</li> <li>5. Entiende cómo un buen sistema de bases de datos puede ayudar a tomar decisiones estratégicas a nivel industrial.</li> </ol>
---	---

## Contenidos

### Por qué un sistema de bases de datos

1. Importancia de las bases de datos.
2. Sistemas de bases de datos versus almacenamiento ad-hoc.
3. Arquitectura básica de los sistemas de bases de datos.
4. Modelos de datos: XML, key-value, relacional.

### Introducción al modelo relacional

7. Conceptos básicos.
8. SQL básico.

### Modelamiento (diseño)

9. Creación de Tablas.
10. Lenguajes de modelamiento; diagramas E/R.
11. Principios de diseño.
12. Desde el diagrama E/R a esquemas relacionales.
13. Modelado de restricciones, formas normales.

### SQL avanzado

14. Outerjoins y valores nulos.

15. Vistas.

16. Procedimientos almacenados y cursores.

17. Transacciones y control de concurrencia.

### Conceptos teóricos

20. El Álgebra relacional.

21. Limitaciones de SQL: valores nulos, clausura transitiva, etc.

### Indexación y optimización

22. Procesamiento de operadores relacionales.

23. Optimización de consultas relacionales.

24. Índices en memoria secundaria.

25. Índices invertidos y búsqueda en texto

### Otros modelos de datos

25. Paradigma JSON

26. Análisis de datos con Pandas

27. Bases de datos orientadas a grafos

## Metodología

Las clases se dividen en submódulos expositivos, de discusión activa entre pares, o de desarrollo de guías, o de trabajo en el proyecto. Un proyecto semestral de desarrollo de una base de datos cuya información se revelará a medida que avanza el curso. Cada clase se estructura de la siguiente manera:

- **Lunes:** se sube el material complementario a la página.

- **Miércoles:** clase expositiva, con tiempo para trabajar en actividad, tarea, evaluación o proyecto.
- **Viernes:** se entrega la actividad en caso de que sea evaluada.

Calendario de semanas

Numero	Semana	Evaluación
1	Introducción	-
2	Álgebra relacional	Control evaluado 1
3	SQL	Guía de ejercicios
4	Modelamiento en BD	Proyecto
5	Restricciones de Integridad	Proyecto
6	SQL avanzado	Guía de ejercicios
7	Almacenamiento e índices	Control evaluado 2
8	Algoritmos en BD	Control evaluado 2
9	Procedimientos Almacenados	Proyecto
10	Transacciones y recuperación de fallas	Proyecto
11	NOSQL y JSON/MongoDB	Guía de ejercicios y proyecto
12	Búsqueda por texto	Guía de ejercicios y proyecto
13	SQL para data science 1	Control evaluado 3
14	SQL para data science 2	Control evaluado 3
15	Otros tópicos	Guía de ejercicios

## Evaluación

La evaluación se realizará en base a:

- 3 Actividades evaluadas, a realizarse principalmente en clases.
- Examen, con el objetivo de evaluar conocimientos teóricos.
- Un proyecto semestral con 5 evaluaciones parciales. El objetivo del proyecto es que el alumno desarrolle una base de datos para una aplicación basada en un caso real. Las etapas avanzadas del proyecto introducirán al alumno otras problemáticas reales de las bases de datos.

La **nota de controles y exámenes** (NCE) corresponde al promedio de las actividades y el examen, en donde el examen pondera el doble y además puede reemplazar a la peor nota en alguna actividad. En otras palabras,  $NCE = (A1 + A2 + A3 + Ex + 2 \cdot Ex - \min(A1, A2, A3, Ex))/5$ .

La **nota del proyecto** (NP) corresponde al promedio ponderado de todas las entregas del proyecto. La ponderación es: Entrega 1: 5 %, Entrega 2: 30 %, Entrega 3: 25 %, Entrega 4: 20 %, Entrega 5: 20 %,.

**Importante:** La nota de las entregas 2, 3, 4 y 5 estarán sujetas a una evaluación de pares que puede alterar la nota del alumno. Si el profesor lo considera necesario, citará a los alumnos cuyas evaluaciones de pares indican un comportamiento irregular en las entregas.

Para aprobar el ramo, el alumno debe cumplir que NCE y NP sean mayores a 4. En ese caso, la **nota final** se calcula como  $NF = (0,5NCE + 0,5NP)$ . En caso contrario,  $NF = \min\{NCE, NP\}$ .

La fecha y forma de entrega de las entregas del proyecto será informada a medida que avanza el curso. El plazo final siempre es a las 23:59 del día de entrega.

## Bibliografía

- Ramakrishnan, Gehrke. Database Management Systems (3ed).

- Garcia-Molina, Ullman, Widom. Database Systems: The complete book (2ed).
- Date, C. J.. Introducción a los sistemas de bases de datos (7ed).
- Elmasri, Ramez. Fundamentos de sistemas de bases de datos (5ed).
- Abiteboul, Hull, Vianu. Foundations of databases. (sobretudo para álgebra relacional y optimización).
- Phillip Greenspun. SQL for Web Nerds. <http://philip.greenspun.com/sql/>

## Otros

El Departamento de Ciencias de la Computación adopta una política de tolerancia-cero frente a copias o plagios. Se sugiere revisar las políticas y penalidades que el departamento establece ante estas acciones.

El curso tiene dos canales de comunicación oficiales: Las clases y la página Web (en sistema SIDING). Se asume que que toda la información que es entregada por ambos canales llega a todos los alumnos. Por lo mismo, se sugiere a los alumnos revisar la página Web constantemente.

Es responsabilidad del alumno ponerse al día con los contenidos en caso de inasistencia a clases.