**MODELOS Y BASES DE DATOS**

**XML y SQL**

**2022-02**

**Laboratorio 6/6 [: ) ]**

**PUNTO UNO. Sólo XML. CATALOGO DE CURSOS (courses.xml)**

El caso es uno de los propuestos en el curso de bases de datos de Stanford.

**A. Explorando**

1. Explore el archivo usando un navegador y un editor. ¿Cuáles son las ventajas de uno u otro?

* Al usar el navegador permite que sea más fácil de navegar por el archivo courses.xml, permitiendo que sea más sencillo en visualizar sus diferentes componentes, mientras que un editor permite que sea más fácil hacer los atributos del xml.

2. Describa la información que encuentra en el archivo.

* Se encuentra en una raíz llamada “Courses\_Catalog” la cual se compone de diferentes departamentos (Ramas) los cuales tienen un título y su respectivo código, un departamento tiene diferentes Cursos, estos poseen un código y numero además tiene instructores, estos pueden ser profesores o lectores que tiene como atributos un nombre y un apellido.

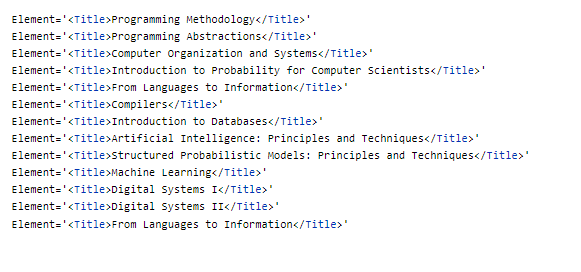
**B. Consultas iniciales Implemente las siguientes consultas:**

Para evaluar use la herramienta que aparece en moodle.

1. ¿De cuáles cursos se tiene información?

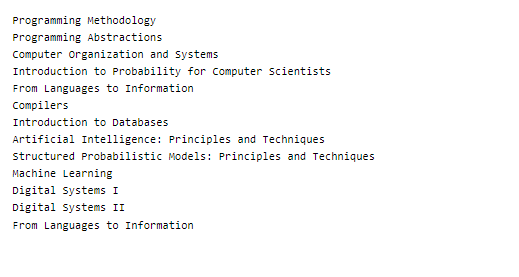
a. Nombre con etiqueta

/Course\_Catalog/Department/Course/Title



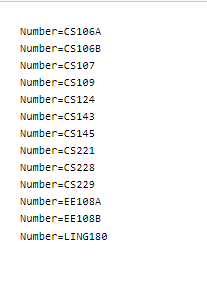
b. Nombre sin etiqueta

/Course\_Catalog/Department/Course/Title/text()



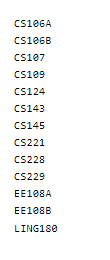
c. número con atributo

/Course\_Catalog/Department/Course/@Number



d. Número sin atributo

/Course\_Catalog/Department/Course/@Number/data()



e. Número total de cursos

/Course\_Catalog/Department/count(Course)

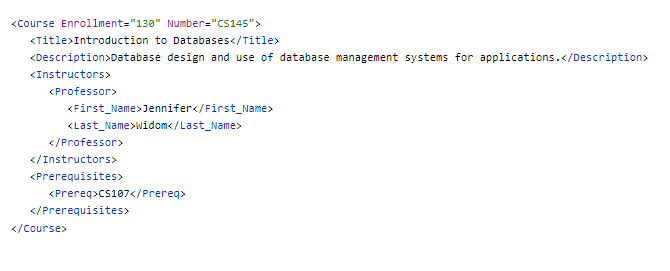
count(/Course\_Catalog/Department/Course)



2. ¿Cuál es la información del curso Introduction to Databases?

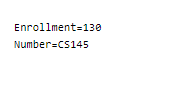
a. Toda la información

//Course[Title="Introduction to Databases"]



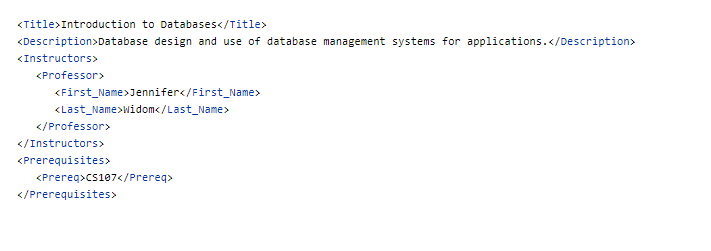
b. Todos los atributos

//Course[Title="Introduction to Databases"]/@\*



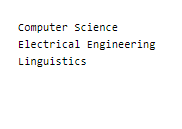
c. Todas las etiquetas

//Course[Title="Introduction to Databases"]/\*



3. ¿Cuántos departamentos tenemos? ¿Cuáles son?[Nombre]

/Course\_Catalog/Department/Title/text()



4. ¿A qué departemento pertenece el curso Introducción to Databases? [Codigo]

//Course[Title="Introduction to Databases"]/../@Code



5. ¿Cuáles cursos son iniciales (no tienen prerrequisitos)? [Nombre]

6. ¿Cuáles cursos son finales (no son prerrequisitos de ninguno)? [Nombre]

7. ¿Cuáles directores de departamento no colaboran en cursos?[Toda la información] (Los profesores se pueden identificar por su apellido)

8. ¿Cuál es el total de inscripciones de los cursos?

sum(/Course\_Catalog/Department/Course/ @Enrollment /data())



9. ¿De cuáles cursos no se conoce el número de inscritos? [Nombre]

/Course\_Catalog/Department/Course[not(@Enrollment)]/ Title/text()



10. ¿Cuál curso tiene el mayor número de inscritos? [Nombre]

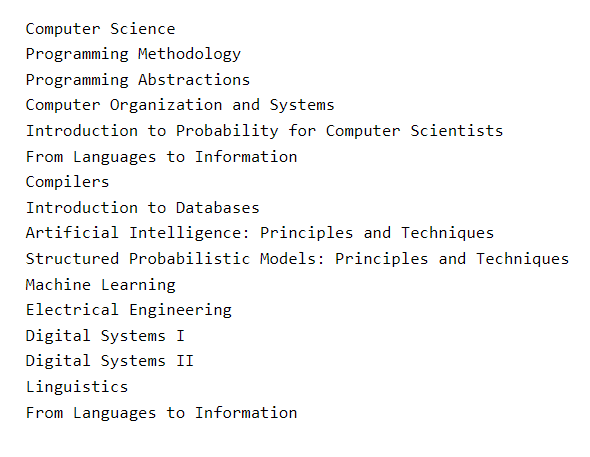
/Course\_Catalog/Department /Course [max(/Course\_Catalog/Department/ Course/@Enrollment) = @Enrollment]/Title/text()



**B. Consultas quiz Standford**

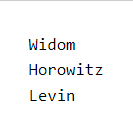
1. Return all Title elements (of both departments and courses). (16)

//Title/text()



2. Return last names of all department chairs. (3)

/Course\_Catalog/Department/Chair/Professor/Last\_Name/text()



3. Return titles of courses with enrollment greater than 500. (2)

//Course[@Enrollment >500]/Title/text()



4. Return titles of departments that have some course that takes "CS106B" as a prerequisite. (2)

//Department[Course/Prerequisites/Prereq = 'CS106B']/Title/text()



5. Return last names of all professors or lecturers who use a middle initial. Don't worry about eliminating duplicates. (5)

//(Professor|Lecturer)[Middle\_Initial]/Last\_Name/text()



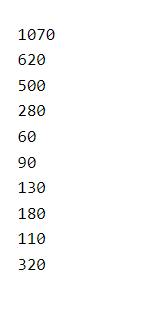
6. Return the title of courses that have a cross-listed course (i.e., that have "Cross-listed" in their description). (2)

count(//Course[contains(Description, 'Cross-listed')])



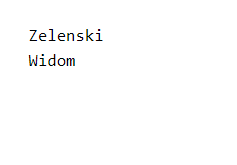
7. Return the enrollment of all courses in the CS department. (10)

//Department[@Code = 'CS']/Course/data(@Enrollment)



8. Return last names of instructors teaching at least one course that has "system" in its description and enrollment greater than 100. (2)

//Course[contains(Description, 'system') and @Enrollment > 100]/Instructors/\*/Last\_Name/text()



9. Return the title of the course with more than 2 instructors (\*)(1)

/Course\_Catalog/Department /Course[count(Instructors/\*)>2]/Title



10. Return course numbers of courses that have the same title as some other course. (Hint: You might want to use the "preceding" and "following" navigation axes for this query, which were not covered in the video or our demo script; they match any preceding or following node, not just siblings.) (2)

11. Return the courses numbers of courses that have no lecturers as instructors. (\*)(10)

count(//Course[count(Instructors/Lecturer) = 0])



12. Return titles of courses taught by the chair of a department. For this question, you may assume that all professors have distinct last names. (1)

//Course[Instructors/Professor/Last\_Name = parent::Department/Chair/Professor/Last\_Name]/Title/text()



13. Return titles of courses taught by a professor with the last name "Ng" and by a professor with the last name "Thrun". (\*)(1)

//Course[Instructors/Professor[Last\_Name = 'Ng'] and Title != //Course[Instructors/Professor[Last\_Name = 'Thrun']]/Title]/Title/text()



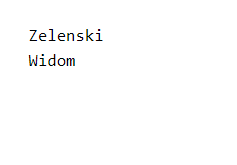
14. Return course numbers of courses that have a course taught by Eric Roberts as a prerequisite. (4)

//Course[Prerequisites/Prereq = //Course[Instructors/\*[First\_Name = 'Eric' and Last\_Name = 'Roberts']]/data(@Number)]/data(@Number)



Return last names of instructors teachiing at least one course that has system in its description and enrollment greater than 100.(2)

//Course[contains(Description, 'system') and @Enrollment > 100]/Instructors/\*/Last\_Name/text()

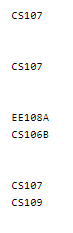


**C. Consultas propias**

Propongan e implementen cinco consultas propias

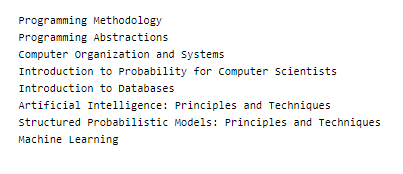
1.Los requeriditos de los cursos[código]

//Prerequisites/data()

2. Retornar los títulos del curso con enrollement más de 100

//Course[@Enrollment > 100]/Title/text()



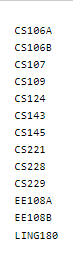
3. ¿Cuál curso tiene el menor número de inscritos? [Nombre]

/Course\_Catalog/Department /Course [min(/Course\_Catalog/Department/ Course/@Enrollment) = @Enrollment]/Title/text()



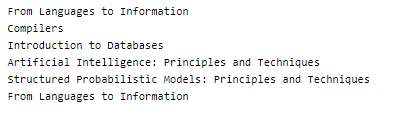
4.Codigo de los cursos

/Course\_Catalog/Department /Course/@Number/data()



5. Retornar los títulos del curso con enrollement menores de 200

//Course[@Enrollment < 200]/Title/text()



**D. Esquema**

Proponga un DTD para estos datos. Para verificar use la herramienta que aparece en moodle.

<?xml version = "1.0"?>

<!DOCTYPE Course\_Catalog[

<!ELEMENT Course\_Catalog (Department\*)>

<!ELEMENT Department (Title\*, Chair\*, Course\*)>

<!ELEMENT Title (#PCDATA)>

<!ELEMENT Chair (Professor\*)>

<!ELEMENT Professor (First\_Name+,Middle\_Initial?, Last\_Name+)>

<!ELEMENT First\_Name (#PCDATA)>

<!ELEMENT Middle\_Initial (#PCDATA)>

<!ELEMENT Last\_Name (#PCDATA)>

<!ELEMENT Course (Title\*, Description?, Instructors\*,Prerequisites\*)>

<!ELEMENT Description (#PCDATA)>

<!ELEMENT Instructors (Professor,Lecture\*)>

<!ELEMENT Lecture (First\_Name+,Middle\_Initial+,Last\_Name+)>

<!ELEMENT Prerequisites (Prereq+)>

<!ELEMENT Prereq (#PCDATA)>

<!ATTLIST Course Number CDATA #REQUIRED>

<!ATTLIST Course Enrollment CDATA #REQUIRED>

<!ATTLIST Department Code CDATA #REQUIRED>

]>

**E. Nuevos datos**

Incluya en el archivo la información de tres cursos que no existan en el archivo. Verifique que cumple el esquema definido anteriormente.

<Course Number="CS129B" Enrollment="100">

<Title>Programacion Orientada a Objetos</Title>

<Description>crear un modelo del problema de destino en sus programas.</Description>

<Instructors>

<Professor>

<First\_Name>Sebastian</First\_Name>

<Last\_Name>Blanco</Last\_Name>

</Professor>

<Lecturer>

<First\_Name>Jose</First\_Name>

<Middle\_Initial>A.</Middle\_Initial>

<Last\_Name>Correa</Last\_Name>

</Lecturer>

</Instructors>

</Course>

<Course Number="CS130B" Enrollment="80">

<Title>Modelos y base de datos</Title>

<Description>la estructura lógica que adopta la base de base datos, incluyendo las relaciones y limitaciones que determinan cómo se almacenan y organizan y cómo se accede a los datos.</Description>

<Instructors>

<Professor>

<First\_Name>Pepito</First\_Name>

<Last\_Name>Perez</Last\_Name>

</Professor>

<Lecturer>

<First\_Name>Frank</First\_Name>

<Last\_Name>Rodriguez</Last\_Name>

</Lecturer>

</Instructors>

</Course>

<Course Number="CS131B" Enrollment="300">

<Title>Teoria de sistemas y organizaciones</Title>

<Instructors>

<Professor>

<First\_Name>Camila</First\_Name>

<Last\_Name>Poveda</Last\_Name>

</Professor>

<Professor>

<First\_Name>Daniela</First\_Name>

<Last\_Name>Jimenez</Last\_Name>

</Professor>

<Lecturer>

<First\_Name>Daniel</First\_Name>

<Last\_Name>Ramirez</Last\_Name>

</Lecturer>

</Instructors>

</Course>

**PUNTO DOS. (XML – Oracle) laHause. Oferta. Anexos**

Este punto vamos a perfeccionar uno de los atributos de la base de datos.

1) Proponga la estructura XML necesaria para tener la información de este atributo. DTD y ejemplos XML OK y XML NoOK. Explique.

// un tipo de vivienda tiene descripción, tamaño y que tipo es(casa, apartamento,bodega,etc)

<XMLOK>

<Tvivienda>

<Description>

Una casa de tres pisos con 3 habitacines y un parqueadero

</Description>

<Tipo>

Casa

</Tipo>

<size>

100m^2

</size>

</Tvivienda>

<Tvivienda>

<Description>

Una apartamento con 2 habitacines y un baño

</Description>

<Tipo>

apartamento

</Tipo>

<size>

60m^2

</size>

</Tvivienda>

<Tvivienda>

<Description>

Una bodega ubicada en las afueras de la ciudad

</Description>

<Tipo>

Bodega

</Tipo>

<size>

400m^2

</size>

</Tvivienda>

XMLNoOK

no tiene tamaño cuando este es requerido

<Tvivienda>

<Description>

Una casa de dos pisos

</Description>

<Tipo>

casa

</Tipo>

</Tvivienda>

2) Actualice la tabla y los datos adicionados a la base de datos. (PoblandoOK, ProblandoNoOK)

3) Implemente la consulta Consultar ofertas sin certificado de libertad en pdf

4) Proponga otra nueva consulta que use Anexos (Diseño e implementación)

5) Extienda la información de este atributo (Anexos-DTD) y proponga una nueva consulta que ilustre la pertinencia de la nueva información registrada en XML. (Diseño e implementación)

**RETROSPECTIVA**

1. ¿Cuál fue el tiempo total invertido en el laboratorio por cada uno de ustedes? (Horas/Hombre)

* 8 horas por persona

2. ¿Cuál es el estado actual del laboratorio? ¿Por qué?

* 70% debido a que no pudimos entender algunas consultas

3. ¿Cuál consideran fue el mayor logro? ¿Por qué?

* Aprender a hacer las consultas en XML

4. ¿Cuál consideran que fue el mayor problema técnico? ¿Qué hicieron para resolverlo?

* La realización de las consultas del quiz porque nos demoramos entendiendo como se implementaban

5. ¿Qué hicieron bien como equipo? ¿Qué se comprometen a hacer para mejorar los resultados?

* Cada uno investigo lo necesario para desarrollar el lab.