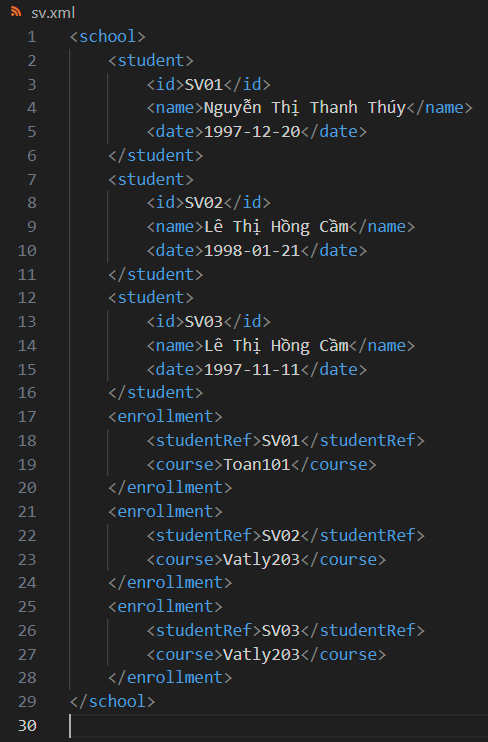
# **Buổi 5: XPATH và Lập trình Python với Xpath**

|  |  |
| --- | --- |
| **STT:** |  |
| **Mã sinh viên:** | KTPM2211045 |
| **Họ và tên:** | Nguyễn Trần Tuấn Khôi |

Cho file XML **sinhvien.xml**



Dựa vào file XML trên, xây dựng các biểu thức Xpath sau:

* Lấy tất cả sinh viên

//student

* Liệt kê tên tất cả sinh viên

//student/name/text()

* Lấy tất cả id của sinh viên

//student/id/text()

* Lấy ngày sinh của sinh viên có id = "SV01"

//student[id="SV01"]/date/text()

* Lấy các khóa học

//enrollment/course/text()

* Lấy toàn bộ thông tin của sinh viên đầu tiên

//student)[1]

* Lấy mã sinh viên đăng ký khóa học "Vatly203"

//enrollment[course="Vatly203"]/studentRef/text()

* Lấy tên sinh viên học môn "Toan101"

//student[id=//enrollment[course="Toan101"]/studentRef]/name/text()

* Lấy tên sinh viên học môn "Vatly203"

//student[id=//enrollment[course="Vatly203"]/studentRef]/name/text()

* Lấy ngày sinh của sinh viên có id="SV01"

//student[id="SV01"]/date/text()

* Lấy **tên và ngày sinh** của mọi sinh viên sinh năm 1997

//student[starts-with(date,"1997")]

* Lấy **tên** của các sinh viên có ngày sinh **trước năm 1998**

//student[date < "1998-01-01"]/name/text()

* Đếm tổng số sinh viên

count(//student)

* Thêm vào file XML thông tin 2 sinh viên và viết câu xpath: Lấy tất cả sinh viên chưa đăng ký môn nào.

//student[not(id = //enrollment/studentRef)]

* Lấy phần tử <date> anh em ngay sau <name> của SV01

//student[id="SV01"]/name/following-sibling::date

* Lấy phần tử <id> anh em ngay trước <name> của SV02

//student[id="SV02"]/name/preceding-sibling::id

* Lấy toàn bộ node <course> trong cùng một <enrollment> với studentRef='SV03'

//enrollment[studentRef="SV03"]/course

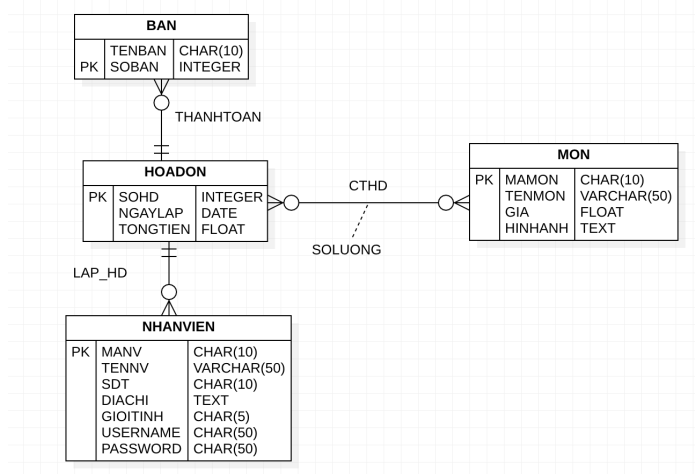
* Lấy sinh viên có họ là “Trần”

//student[contains(name,"Trần")]

* Lấy năm sinh của sinh viên SV01

substring(//student[id="SV01"]/date/text(), 1, 4)

Dùng thư viện lxml để kiểm tra kết quả truy vấn trên.



Dựa mô hình ERD buổi 4, cho file XML **quanlybanan.xml**, viết xpath theo yêu cầu sau:

* Lấy tất cả bàn

//BAN

* Lấy tất cả nhân viên

//NHANVIEN

* Lấy tất cả tên món

//MON/TENMON/text()

* Lấy tên nhân viên có mã NV02

//NHANVIEN[MANV='NV02']/TENV/text()

* Lấy tên và số điện thoại của nhân viên NV03

//NHANVIEN[MANV='NV03']/TENV/text() | //NHANVIEN[MANV='NV03']/SDT/text()

* Lấy tên món có giá > 50,000

 //MON[GIA > 50000]/TENMON/text()

* Lấy số bàn của hóa đơn HD03

//HOADON[SOHD='HD03']/SOBAN/text()

* Lấy tên món có mã M02

//MON[MAMON='M02']/TENMON/text()

* Lấy ngày lập của hóa đơn HD03

//HOADON[SOHD='HD03']/NGAYLAP/text()

* Lấy tất cả mã món trong hóa đơn HD01

//HOADON[SOHD='HD01']//CTHD/MAMON/text()

* Lấy tên món trong hóa đơn HD01

//MON[MAMON=//HOADON[SOHD='HD01']//CTHD/MAMON]/TENMON/text()

* Lấy tên nhân viên lập hóa đơn HD02

//NHANVIEN[MANV=//HOADON[SOHD='HD02']/MANV]/TENV/text()

* Đếm số bàn

count(//BAN)

* Đếm số hóa đơn lập bởi NV01

count(//HOADON[MANV='NV01'])

* Lấy tên tất cả món có trong hóa đơn của bàn số 2

//MON[MAMON=//HOADON[SOBAN='2']//CTHD/MAMON]/TENMON/text()

* Lấy tất cả nhân viên từng lập hóa đơn cho bàn số 3

//NHANVIEN[MANV=//HOADON[SOBAN='3']/MANV]/TENV/text()

* Lấy tất cả hóa đơn mà nhân viên nữ lập

//HOADON[MANV=//NHANVIEN[GIOITINH='Nữ']/MANV]

* Lấy tất cả nhân viên từng phục vụ bàn số 1

//NHANVIEN[MANV=//HOADON[SOBAN='1']/MANV]/TENV/text()

* Lấy tất cả món được gọi nhiều hơn 1 lần trong các hóa đơn

//MON[MAMON='M02']/TENMON/text()

* Lấy tên bàn + ngày lập hóa đơn tương ứng SOHD='HD02'

//BAN[SOBAN=//HOADON[SOHD='HD02']/SOBAN]/TENBAN/text() | //HOADON[SOHD='HD02']/NGAYLAP/text()

Dùng thư viện lxml để kiểm tra kết quả truy vấn trên.