

Partiendo de universidad.xml obtener utilizando AXIS XPATH las siguientes consultas:

```
<?xml version="1.0" encoding="UTF-8"?>
<universidad>
  <departamento telefono="112233" tipo="A">
    <codigo>IFC1</codigo>
    <nombre>Informatica</nombre>
    <empleado salario="2000">
      <puesto>Asociado</puesto>
      <nombre>Juan Alvarez</nombre>
    </empleado>
    <empleado salario="2300">
      <puesto>Profesor</puesto>
      <nombre>Antonio Heras</nombre>
    </empleado>
  </departamento>
  <departamento telefono="990033" tipo="A">
    <codigo>MAT1</codigo>
    <nombre>Matematicas</nombre>
    <empleado salario="1900">
      <puesto>Tecnico</puesto>
      <nombre>Ana Solevita</nombre>
    </empleado>
    <empleado salario="2100">
      <puesto>Profesor</puesto>
      <nombre>Pedro Menor</nombre>
    </empleado>
    <empleado salario="2300">
      <puesto>Profesor</puesto>
      <nombre>Marta Paniagua</nombre>
    </empleado>
    <empleado salario="2500">
      <puesto>Tutor</puesto>
      <nombre>Alicia Alonso</nombre>
    </empleado>
  </departamento>
  <departamento telefono="886633" tipo="A">
    <codigo>MAT2</codigo>
    <nombre>Análisis</nombre>
    <empleado salario="1900">
      <puesto>Asociado</puesto>
      <nombre>Laura Ruiz</nombre>
    </empleado>
    <empleado salario="2200">
      <puesto>Asociado</puesto>
      <nombre>Mario Martin</nombre>
    </empleado>
  </departamento>
</universidad>
```

1. Devolver todos los nodos hijos de universidad.

```
/universidad/child::*
/child::universidad/child::element()
```

2. Devolver los descendientes del nodo departamento

```
/universidad/departamento/descendant::*
/child::universidad/child::departamento/descendant::element()
```

3. Devolver los nodos empleado descendiente de los nodos departamento

```
/universidad/departamento/descendant::empleado
```

4. Devolver todos los elementos nombre descendientes de universidad, tanto nombres de departamento como de empleado.

```
/universidad/descendant::nombre
```

```
data(/universidad/descendant::nombre) solo devuelve el nombre
```

5. Selecciona todos los hermanos de departamento a partir del primero, siguiendo el orden en el documento.

```
/universidad/departamento/following-sibling::*
```

```
/universidad/departamento[2]/following-sibling::*
```

 hace lo mismo pero desde el 2

6. Selecciona todos los hermanos de los elementos empleado que tienen salario superior a 2000

```
//empleado/following-sibling::empleado[@salario>2000]
```

7. Selecciona los nodos hermanos de Ana Solevita.

```
//empleado[nombre="Ana Solevita"]/following-sibling::*
```

8. Selecciona los nombres de los empleados hermanos de Ana Solevita que son profesores

```
//empleado[nombre="Ana Solevita"]/following-sibling::empleado[puesto="Profesor"]/nombre/text()
```

9. Selecciona el nombre de los padres de los elementos empleado.

```
//empleado/parent::departamento/nombre
```

10. Selecciona el nombre del padre del empleado Ana Solevita

```
//empleado[nombre="Ana Solevita"]/parent::departamento/nombre
```

11. Selecciona los descendientes del departamento que ocupa la posición 3 en el documento.

```
/descendant::departamento[1]
```

12. Obtener los departamentos con más de 3 empleados

```
/child::universidad/child::departamento[count(child::empleado)>3]
```

```
/universidad/departamento[count(empleado)>3]
```

13. Obtener las etiquetas de los nombres de los departamentos

```
/child::universidad/child::departamento/child::nombre
```

```
/universidad/departamento/nombre
```

```
data(/universidad/departamento/nombre)
```

14. Obtener los nombres de los departamentos.

```
/child::universidad/child::departamento/child::nombre/child::text()
```

15. Devolver el nombre de los departamentos de tipo B y con 2 o más empleados

```
/child::universidad/child::departamento[attribute::tipo="B"][count(child::empleado)>=2]/child::nombre/child::text()
```

```
/universidad/departamento[@tipo="B" and count(empleado)>=2]/nombre/text()
```