**ESCUELA POLITÉCNICA NACIONAL**

**ESCUELA DE FORMACIÓN DE TECNÓLOGOS**

**TÓPICOS ESPECIALES**

**ING. EDWIN SALVADOR**

**TATIANA OLMEDO**

**“EJERCICIOS”**

**2015 – A**

* **EJERCICIO 2.1**

<xs:element name="ejercicioUno">

<xs:simpleType>

<xs:restriction base="xs:string">

<xs:minLength value="2"/>

<xs:maxLength value="30"/>

<xs:pattern value="[A-Za-z\-. ’]{2,30}"/>

</xs:restriction>

</xs:simpleType>

</xs:element>

* [A-Za-z\-. ’] Cualquier letra mayúscula o minúscula seguida de un guión (-).
* {2,30} Se puede repetir mínimo 2 veces o máximo 30 la expresión que está antes de las llaves.

Ejemplos:

|  |  |
| --- | --- |
| M-M-M | Válido |
| M | Inválido |

* **EJERCICIO 2.2**

<xs:element name="ejercicioDos">

<xs:simpleType>

<xs:restriction base="xs:string">

<xs:pattern value="\(?\d{3}\)?[ \-\.]?\d{3}[ \-\.]?\d{4}"/>

<xs:minLength value="10"/>

<xs:maxLength value="16"/>

</xs:restriction>

</xs:simpleType>

</xs:element>

* \d{3} Nos indica que son dígitos, en este caso 3 dígitos.
* \d{4} Nos indica dígitos, en este caso 4 dígitos.
* ? Busca cualquier elemento basado en la última expresión.
* \-\ Deben ir los dígitos seguidos de un guión.

Ejemplos:

|  |  |
| --- | --- |
| 568-489-5879 | Válido |
| 1234-256-987 | Inválido |

* **EJERCICIO 2.3**

<xs:element name="ejercicioTres">

<xs:simpleType>

<xs:restriction base="xs:string">

<xs:pattern value="(\w+\.)\*\w+@(\w+\.)+[A-Za-z]{2,9}"/>

<xs:minLength value="6"/>

<xs:maxLength value="255"/>

</xs:restriction>

</xs:simpleType>

</xs:element>

* \w Nos indica que debe ir cualquier caracter alfanumérico.
* + Pueden ir uno o más caracteres después de \w.
* [A-Za-z] Pueden ir letras mayúsculas o minúsculas de la “aA” a la “zZ”
* {2,9} Se puede repetir mínimo 2 veces o máximo 9 la expresión que está antes de las llaves.
* \. Debe ir punto después de la expresión.

Ejemplos:

|  |  |
| --- | --- |
| tatyolmedo@gmail.com | Válido |
| tatyolmedo@gmailcom | Inválido |

* **EJERCICIO 2.4**

<xs:element name="ejercicioCuatro">

<xs:simpleType>

<xs:restriction base="xs:string">

<xs:pattern value="\d{5}(-\d{4})?"/>

<xs:minLength value="5"/>

<xs:maxLength value="10"/>

</xs:restriction>

</xs:simpleType>

</xs:element>

* \d{5} Nos indica dígitos, en este caso 5 dígitos.
* \d{4} Nos indica dígitos, en este caso 4 dígitos.
* ? Busca cualquier elemento basado en la última expresión.

Ejemplos:

|  |  |
| --- | --- |
| 25879-1245 | Válido |
| 458615-1254 | Inválido |

* **EJERCICIO 2.5**

<xsd:simpleType name="ejercicioCinco">

<xsd:restriction base="xsd:string">

<xsd:pattern value="\d{3}(-)?\d{2}(-)?\d{4}"/>

<xsd:minLength value="9"/>

<xsd:maxLength value="11"/>

</xsd:restriction>

</xsd:simpleType>

* \d{3} Nos indica dígitos, en este caso 3 dígitos.
* \d{2} Nos indica dígitos, en este caso 2 dígitos.
* \d{4} Nos indica dígitos, en este caso 4 dígitos.
* ? Busca cualquier elemento basado en la última expresión.

Ejemplos:

|  |  |
| --- | --- |
| 456-12-4587 | Válido |
| 1235-78-3325 | Inválido |

* **EJERCICIO 2.6**

<xsd:simpleType name="ejercicioSeis">

<xsd:restriction base="xsd:string">

<xsd:length value="2" />

<xsd:enumeration value="AB"/>

<xsd:enumeration value="BC"/>

<xsd:enumeration value="MB"/>

<xsd:enumeration value="NB"/>

<xsd:enumeration value="NL"/>

<xsd:enumeration value="NS"/>

<xsd:enumeration value="NT"/>

<xsd:enumeration value="NU"/>

<xsd:enumeration value="ON"/>

<xsd:enumeration value="PE"/>

<xsd:enumeration value="QC"/>

<xsd:enumeration value="SK"/>

<xsd:enumeration value="YT"/>

</xsd:restriction>

</xsd:simpleType>

* **EJERCICIO 2.7**

<xs:element name="ejercicioSiete">

<xs:simpleType>

<xs:restriction base="xs:string">

<xs:pattern value="(https?://)?(www.)?[-\w]+(\.\w{2,3})+(:\d{2,5})?

(/([\w/\_.]\*)?)?"/>

</xs:restriction>

</xs:simpleType>

</xs:element>

* ? Busca cualquier elemento basado en la última expresión.
* \w Nos indica que debe ir cualquier caracter alfanumérico.
* {2,3} Se puede repetir mínimo 2 veces o máximo 3 la expresión que está antes de las llaves.
* {2,5} Se puede repetir mínimo 2 veces o máximo 5 la expresión que está antes de las llaves.
* \w/\_. Une caracteres como “b” a ‘b’ase o ‘b’alón.
* \* El grupo puede aparecer cero o muchas veces.

Ejemplos:

|  |  |
| --- | --- |
| https://www.semop.gob.ec:8081/Semop | Válido |
| https://www.semop.gob.ec:80/Semop | Inválido |

* **EJERCICIO 2.8**
* Edwi

<xs:element name="ejercicioOcho">

<xs:simpleType>

<xs:restriction base="xs:string">

<xs:pattern value="(((\d{0,2})|(1(\d){0,2})|(2[0-4]\d)|(25[

0-5]))\.){3} ((\d{0,2})|(1(\d){0,2})|(2[0-4]\d)|(25[0-5]))"/>

</xs:restriction>

</xs:simpleType>

</xs:element>

* | Compara dos expresiones, puede ser una u otra pero no ambas.