

Tutorial Validación Modelo con SHACL

Mauricio Espinoza

Crear ontología (I)

- Cambiar la IRI a: <http://www.example.org/#>

Active ontology × Entities × Object properties × Individuals by

Ontology header:

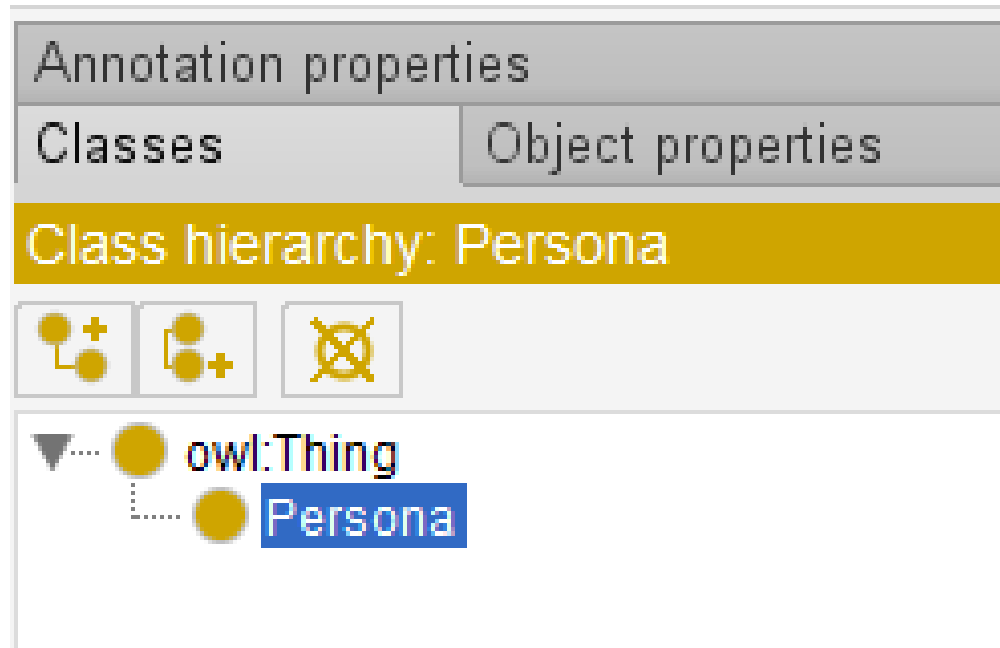
Ontology IRI

Ontology Version IRI

Annotations +

Crear ontología (II)

- Crear la clase Persona



Crear ontología (III)

- Crear la propiedad (objectProperty): tienePadre

The screenshot displays a web-based ontology editor interface. The top navigation bar includes tabs for 'Annotation properties', 'Datatypes', and 'Individuals'. Below this, there are sub-tabs for 'Classes', 'Object properties', and 'Data properties'. The main content area is titled 'Object property hierarchy: tienePadre' and shows a tree structure where 'tienePadre' is a sub-property of 'owl:topObjectProperty'. To the right, there is a panel for 'tienePadre' with a URL 'http://www.example.org/#tienePadre'. This panel has tabs for 'Annotations' and 'Usage'. The 'Annotations' tab is active, showing a list of annotations. Below this, there is a 'Characteristic' section with checkboxes for 'Functional', 'Inverse functional', 'Transitive', 'Symmetric', 'Asymmetric', 'Reflexive', and 'Irreflexive'. The 'Description: tienePadre' section shows a list of characteristics with plus signs for adding more. The 'Domains (intersection)' and 'Ranges (intersection)' sections both show 'Persona' as the domain and range.

Annotation properties | Datatypes | Individuals

Classes | Object properties | Data properties

Object property hierarchy: tienePadre

Annotations | Usage

Annotations: tienePadre

Annotations +

Characteristic + - = > <

Description: tienePadre

Equivalent To +

SubProperty Of +

Inverse Of +

Domains (intersection) +

Persona

Ranges (intersection) +

Persona

☐ Functional

☐ Inverse functional

☐ Transitive

☐ Symmetric

☐ Asymmetric

☐ Reflexive

☐ Irreflexive

Crear ontología (IV)

- Crear la propiedad (dataProperty): nss

The screenshot displays an ontology editor interface with the following components:

- Data property hierarchy: nss**: A tree view on the left showing the hierarchy. It includes `owl:topDataProperty` as the parent, with `fechaNacimiento` and `nss` as children. The `nss` property is highlighted with a blue border.
- Annotations: nss**: A panel on the right with an **Annotations** section containing a plus icon for adding annotations.
- Characteristics: nss**: A panel on the right with a **Functional** checkbox, which is currently unchecked.
- Description: nss**: A panel on the right with several sections for defining the property's domain and range:
 - Equivalent To**: Contains a plus icon.
 - SubProperty Of**: Contains a plus icon.
 - Domains (intersection)**: Contains a plus icon and a domain entry `Persona` (represented by a yellow circle).
 - Ranges**: Contains a plus icon and a range entry `xsd:string` (represented by a red circle).

Crear ontología (V)

- Crear la propiedad (dataProperty): fechaNacimiento

The screenshot displays an ontology editor interface with the following components:

- Data property hierarchy: fechaNacimiento:** A tree view on the left showing the hierarchy: `owl:topDataProperty` (expanded) containing `fechaNacimiento` (selected) and `nss`.
- Annotations: fechaNacimiento:** A panel on the top right showing a list of annotations with a "+" button to add new ones.
- Characteristics: fechaNacimiento:** A panel on the bottom left containing a checkbox for **Functional**, which is currently unchecked.
- Description: fechaNacimiento:** A panel on the bottom right showing the domain and range of the property:
 - Equivalent To:** A "+" button to add equivalent properties.
 - SubProperty Of:** A "+" button to add subproperties.
 - Domains (intersection):** A list containing **Persona** (indicated by a yellow dot).
 - Ranges:** A list containing **xsd:dateTime** (indicated by a red dot).

Crear ontología (V)

- Crear el individuo: Alicia

The screenshot displays an ontology editor interface with several panels:

- Individuals: Alicia**: A list of individuals including Alicia (selected), Boris, Carlos, and Roberto. Each name is preceded by a purple diamond icon.
- Annotations: Alicia**: A panel for adding annotations to the selected individual, currently empty.
- Description: Alicia**: A panel for adding descriptions to the selected individual. It includes a 'Types' section with a yellow circle icon and the label 'Persona', and sections for 'Same Individual As' and 'Different Individuals', each with a plus icon.
- Property assertions: Alicia**: A panel for adding property assertions to the selected individual. It includes sections for 'Object property assertions' and 'Data property assertions', each with a plus icon. Under 'Object property assertions', there are two entries: 'tienePadre Roberto' and 'tienePadre Boris', each preceded by a blue square icon. Under 'Data property assertions', there is one entry: 'nss "987-65-432A"', preceded by a green square icon.

Crear ontología (VI)

- Crear el individuo: Boris

The screenshot displays an ontology editor interface with four main panels:

- Individuals: Boris**: A list of individuals including Alicia, Boris (highlighted in blue), Carlos, and Roberto. It includes icons for adding (+) and deleting (X) individuals.
- Annotations: Boris**: A panel for adding annotations to the selected individual, featuring an "Annotations +" button.
- Description: Boris**: A panel for defining the individual's type, showing "Types +" and "Same Individual As +" buttons. The type "Persona" is listed with a yellow circle icon. To the right are icons for help (?), undo (@), redo (X), and refresh (O).
- Property assertions: Boris**: A panel for adding property assertions, with buttons for "Object property assertions +" and "Data property assertions +". A data assertion is shown: **nss "124-35-6789"**.

Crear ontología (VII)

- Crear el individuo: Carlos

The screenshot displays an ontology editor interface with several panels:

- Individuals: Carlos**: A list of individuals including Alicia, Boris, Carlos (highlighted in blue), and Roberto.
- Annotations: Carlos**: A panel for adding annotations to the selected individual.
- Description: Carlos**: A panel showing the type **Persona** and a "Same Individual As" option.
- Property assertions: Carlos**: A panel for adding property assertions. It includes sections for "Object property assertions" and "Data property assertions". Under "Data property assertions", the property **fechaNacimiento** is shown with the value **"1956-06-25T10:00:00"^^xsd:dateTime**.

Crear ontología (VIII)

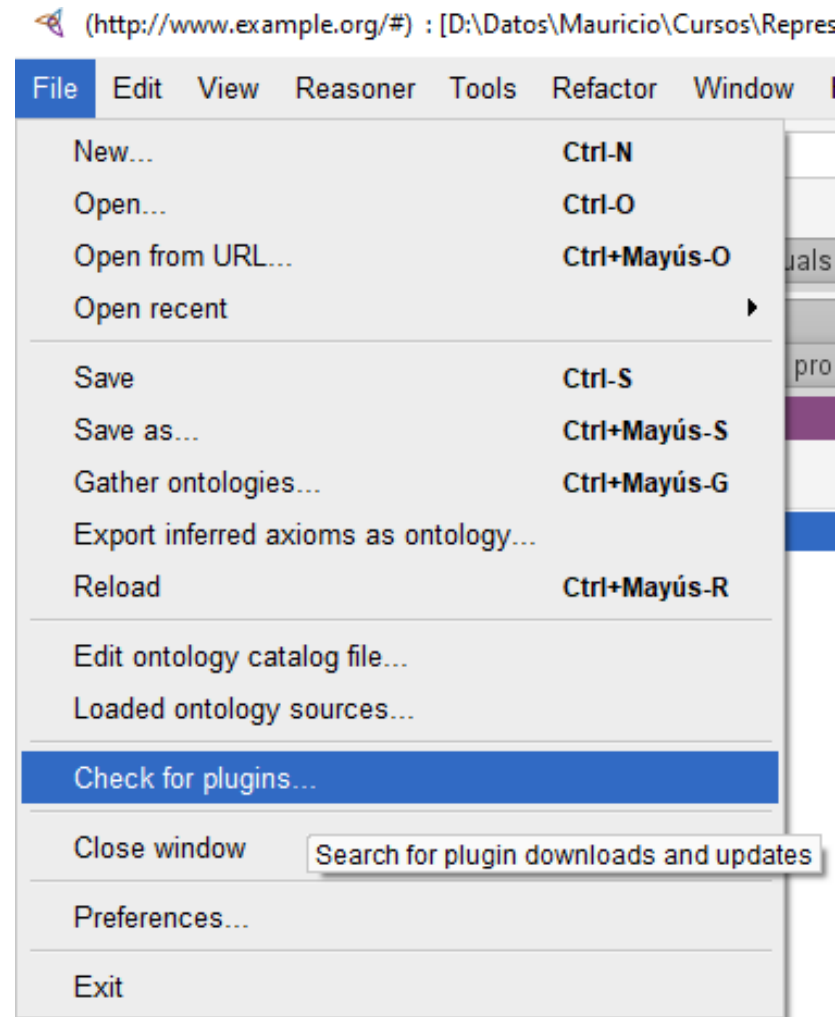
- Crear el individuo: Roberto

The screenshot displays an ontology editor interface with several panels:

- Individuals: Roberto**: A panel on the left showing a list of individuals: Alicia, Boris, Carlos, and Roberto. Roberto is selected and highlighted in blue.
- Annotations: Roberto**: A panel on the right showing a list of annotations for the selected individual Roberto, currently empty.
- Description: Roberto**: A panel on the right showing the description of the selected individual Roberto. It includes a "Types" section with a plus sign and a list of types: **Persona**. There are also icons for adding, removing, and other operations.
- Property assertions: Roberto**: A panel on the right showing property assertions for the selected individual Roberto. It includes sections for "Object property assertions" and "Data property assertions". Under "Data property assertions", there is an assertion: **nss "123-45-6789"**.

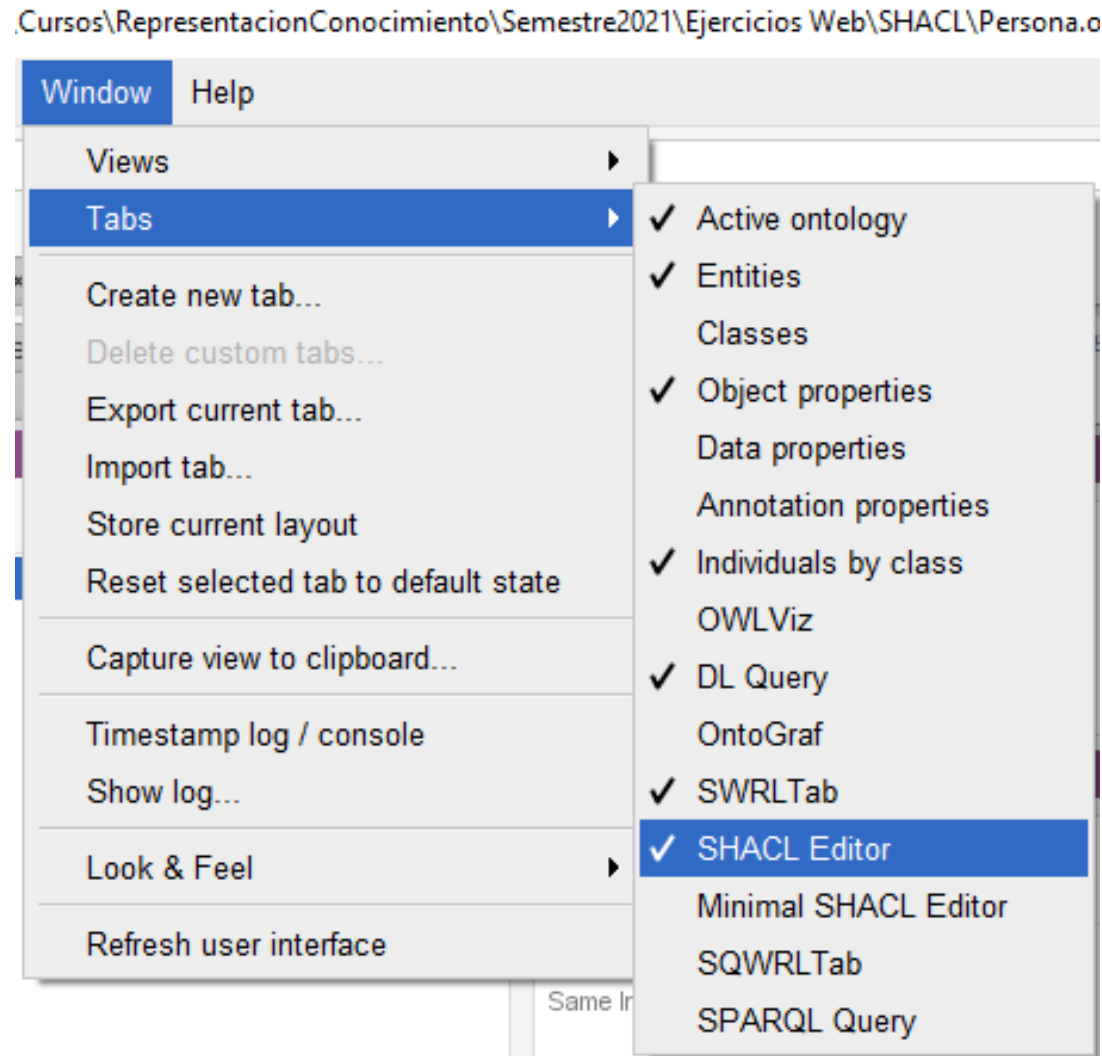
Instalar plugin

- SHACL4Protege



Usar Plugin

- Abrir editor



Crear archivo de validación

Objetivo

- Validar instancias de la clase `ex:Persona` usando SHACL, asegurando que el atributo `ex:nss`:
 - Aparezca a lo sumo una vez
 - Sea una cadena con el formato "999-99-9999"
 - Marcar advertencia si el formato no es correcto
 - Rechazar propiedades no declaradas explícitamente

1. Declarar los prefijos

- Agrega los siguientes prefijos al inicio del archivo de validacion

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

@prefix sh: <http://www.w3.org/ns/shacl#> .

@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

@prefix ex: <http://www.example.org/#> .

@prefix owl: <http://www.w3.org/2002/07/owl#> .

2. Definir el node Shape

- Definir un nodo que representa las restricciones que aplican a los individuos de tipo `ex:Persona`

```
ex:PersonaShape a sh:NodeShape ;  
  sh:targetClass ex:Persona ;
```

Esto indica que el shape se aplica a todos los recursos de tipo `ex:Persona`.

3a. Agregar restricciones sobre ex:nss

- Agrega dos restricciones sobre la propiedad ex:nss usando sh:property:
 1. Solo un valor permitido (sh:maxCount)

```
sh:property [  
    sh:path ex:nss ;  
    sh:maxCount 1 ;  
];
```


3b. Agregar restricciones sobre ex:nss

- Agrega dos restricciones sobre la propiedad ex:nss usando sh:property:
 2. Debe ser string con patrón específico (sh:datatype, sh:pattern)

```
sh:property [  
  sh:path ex:nss ;  
  sh:datatype xsd:string ;  
  sh:pattern "^\\d{3}-\\d{2}-\\d{4}$" ;  
  sh:severity sh:Warning ;  
];
```

El patrón requiere doble barra invertida para escapar: \\d{3} significa tres dígitos.

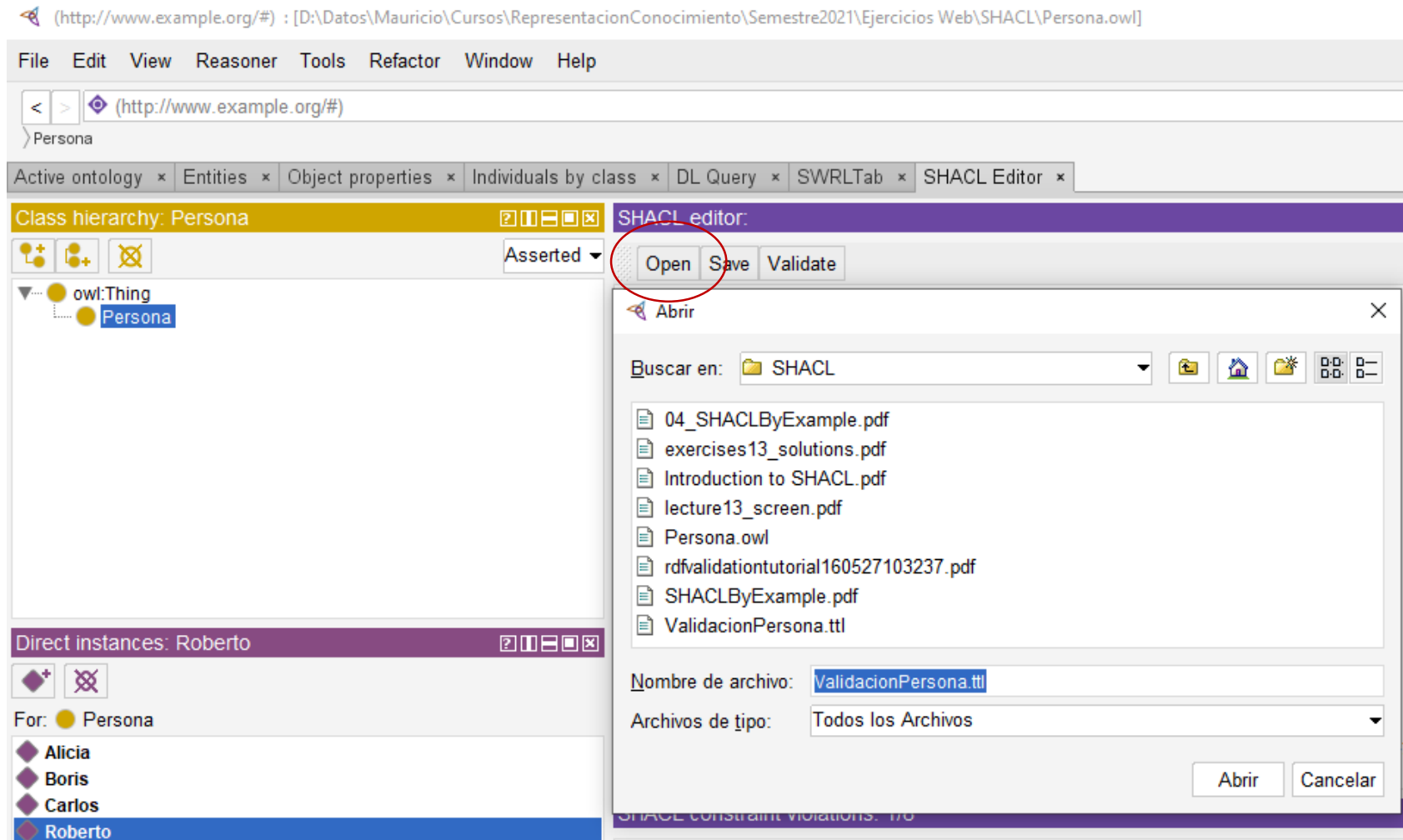
4. Otras validaciones

- Cerrar el shape (sh:closed)
 - Indica que no se permitirán otras propiedades diferentes a las indicadas en el shape:
sh:closed true ;
- Ignorar propiedades genéricas (rdf:type, owl:topDataProperty, etc.)
 - Estas propiedades se permiten sin validación explícita:
sh:ignoredProperties (
 rdf:type
 owl:topDataProperty
 owl:topObjectProperty
);

5. Grabar el archivo de validación

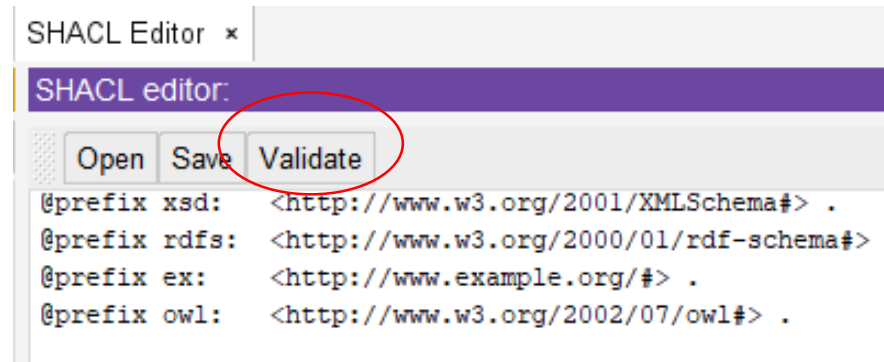
- Al terminar todo el bloque, no olvides el punto final (.):
.
- Guarda todo el archivo de validación como 'validacion.ttl'

Abrir archivo de validación



Validar el modelo

- Active el razonador antes de validar el modelo
- Vaya al SHACL editor y presione el botón Validate



Responder Preguntas

- ¿Cuántos errores encontró el proceso de validación?
- ¿Cuántos errores les corresponde a cada individuo?
- ¿Es cierto que en el individuo Alicia hay dos errores porque esta persona tiene dos padres?
- ¿Por qué se obtiene un error al validar el individuo Carlos?

Crear una regla

- Usando SWRL crea una regla para definir que:
- Una persona que tiene dos padres entonces es la misma persona

Validar el modelo nuevamente

- Activar el razonador y validar el modelo

Direct instances: Roberto

For: Persona

Alicia

Boris

Carlos

Roberto

```
sh:pattern "^\\d{3}-\\d{2}-\\d{4}$" ;
sh:severity sh:Warning ;
] ;
sh:closed true ;
sh:ignoredProperties ( rdf:type owl:topDataProperty owl:topObjectProperty ) ;
.
```

SHACL constraint violations: 1/6

Severity	SourceShape	Message
http://www.w3.org/ns/shacl...	5a690b1fc733b51db9439f...	Property may only have 1 value, but found 2

Responder Preguntas

- ¿Cuántos errores encontró el proceso de validación ahora?
- ¿Cuántos errores les corresponde a cada individuo?
- ¿Porqué aumentaron los errores al validar el modelo?