OWL 1

Lab 8 – INFO216

Python Example On GitHub

(A)Symmetric Properties

https://www.w3.org/TR/owl2-primer/#Advanced_Use_of_Properties

Symmetric

```
g.add((ex.Emma, ex.neighborTo, ex.Cade))
```


```
g.add((ex.neighborTo, RDF.type,
OWL.SymmetricProperty))
```

“You can swap the subject and object with each other”

Asymmetric

```
g.add((ex.Emma, ex.hasFather, ex.Tom))
```


```
g.add((ex.hasFather, RDF.type,
OWL.AsymmetricProperty))
```

“You cannot swap the subject and object with each other”

(Ir)Reflexive Properties

https://www.w3.org/TR/owl2-primer/#Advanced_Use_of_Properties

Reflexive

g.add((ex.livesWith, RDF.type, OWL.ReflexiveProperty))

“You are related to yourself”

Irreflexive

g.add((ex.Emma, ex.hasFather, ex.Tom))

g.add((ex.hasFather, RDF.type, OWL.IrreflexiveProperty))

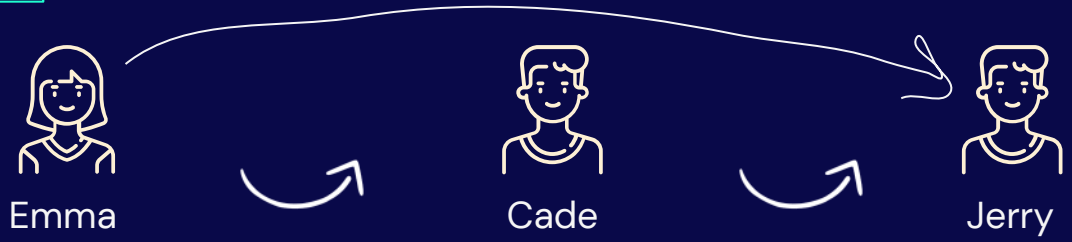
“You cannot be related to yourself”

Transitive Properties

https://www.w3.org/TR/owl2-primer/#Advanced_Use_of_Properties

*"Everything you can do in **two steps**, you can do **in one**"*

```
g.add((ex.Emma, ex.groupPartner, ex.Cade))
g.add((ex.Cade, ex.groupPartner, ex.Jerry))
```




```
g.add((ex.groupPartner, RDF.type, OWL.TransitiveProperty))
```

This gets added!

```
ex:Emma ex:groupPartner ex:Jerry
```

Functional Properties

https://www.w3.org/TR/owl2-primer/#Advanced_Use_of_Properties

Functional

```
g.add((ex.Emma, ex.birthdate,
Literal("1996-10-22", datatype=XSD.date)))

g.add((ex.birthdate, RDF.type,
OWL.FunctionalProperty))
```

"You can only have one instance of it, however it is not unique. Like a birthdate, you only have one, but several people share your birthdate."

InverseFunctional

```
g.add((ex.Emma, ex.socialSecurityNumber,
Literal("123456789", datatype=XSD.integer)))

g.add((ex.socialSecurityNumber, RDF.type,
OWL.InverseFunctionalProperty))
```

"You are the only one with this particular object, no other subject shares this object with you. Like a social security number that only you have, and no one shares it."

Inverse Of

```
g.add((ex.Emma, ex.hasFather, ex.Tom))
```

Emma Tom

TIP:
You can use the ^ to specify the inverse of in SPARQL

```
g.add((ex.hasFather, OWL.inverseOf, ex.fatherOf))
```

Tom Emma

This gets added!

```
ex:Tom ex:fatherOf ex:Emma
```

Differences & Equivalences

Individual	Predicate	Class
OWL.sameAs OWL.differentFrom **GROUPWISE** OWL.AllDifferent OWL.distinctMembers	OWL.equivalentProperty OWL.propertyDisjointWith **GROUPWISE** OWL.AllDisjointProperties OWL.members	OWL.equivalentClass OWL.disjointWith **GROUPWISE** OWL.AllDisjointClasses OWL.members
g.add((ex.Emma, OWL.differentFrom, ex.Cade))	g.add((FOAF.knows, OWL.equivalentProperty, schema.knows))	g.add((ex.Student, OWL.equivalentClass, dbpedia.Student))