

Class: Mother

SubClassOf: Person

EquivalentTo: Woman **and** Parent

Class: ChildlessPerson

EquivalentTo: Person **and not** Parent

Class: MyBirthdayGuests

EquivalentTo: { Bill, John, Mary }

Class: Parent

EquivalentTo: Mother **or** Father

EquivalentTo: hasChild **some** Person

HasKey: hasSSN

Class: HappyPerson

EquivalentTo: hasChild **only** HappyPerson **and** hasChild **some** HappyPerson

Class: NarcisisticJohnsChildren

EquivalentTo: hasParent **value** John **and** loves **Self**

DisjointClasses: Father, Mother

ObjectProperty: hasGrandparent

SubPropertyChain: hasParent **o** hasParent

SubPropertyOf: hasRelative

Domain: Person

Range: Person

InverseOf: hasGrandchild

DataProperty: hasAge

Range: xsd:nonNegativeInteger

Individual: John

Types: hasChild **min** 3 Parent, hasChild **exactly** 4 Parent, hasChild **max** 5 Parent

Types: hasChild **min** 3, hasChild **exactly** 4, hasChild **max** 5

Class: Orphan

EquivalentTo: **inverse** hasChild **only** Dead

ObjectProperty: hasSpouse

Characteristics: Symmetric, Irreflexive, Functional, InverseFunctional

ObjectProperty: hasChild

Characteristics: Asymmetric, Irreflexive

ObjectProperty: hasRelative

Characteristics: Symmetric, Reflexive

ObjectProperty: hasAncestor

Characteristics: Transitive

SubPropertyOf: hasParent

EquivalentProperties: loves, ama

DisjointProperties: hasParent, hasSpouse

SameIndividual: Bill, John

DifferentIndividuals: Bill, Mary

ObjectPropertyAssertion(:hasWife :John :Mary)
ClassAssertion(:Person :Mary)
NegativeObjectPropertyAssertion(:hasWife :Bill :Mary)
DataPropertyAssertion(:hasAge :John "51"^^xsd:integer)
NegativeDataPropertyAssertion(:hasAge :Jack "53"^^xsd:integer)

SubClassOf(:Woman :Person)
EquivalentClasses(:Person :Human)
DisjointClasses(:Woman :Man)
SubObjectPropertyOf(:hasWife :hasSpouse)
ObjectPropertyDomain(:hasWife :Man)
ObjectPropertyRange(:hasWife :Woman)
DataPropertyDomain(:hasAge :Person)
DataPropertyRange(:hasAge xsd:nonNegativeInteger)

DifferentIndividuals(:John :Bill :Matt :Dave)
SameIndividual(:John :Bill)

EquivalentClasses(:Mother **ObjectIntersectionOf**(:Woman :Parent))
EquivalentClasses(:Parent **ObjectUnionOf**(:Mother :Father))
EquivalentClasses(:ChildlessPerson
ObjectIntersectionOf(:Person **ObjectComplementOf**(:Parent)))

SubClassOf(:Grandfather **ObjectIntersectionOf**(:Man :Parent))

EquivalentClasses(:Parent **ObjectSomeValuesFrom**(:hasChild :Person))
EquivalentClasses(:HappyPerson **ObjectAllValuesFrom**(:hasChild :HappyPerson))
EquivalentClasses(:JohnsChildren **ObjectHasValue**(:hasParent :John))
EquivalentClasses(:NarcisticPerson **ObjectHasSelf**(:loves))
ClassAssertion(**ObjectMaxCardinality**(4 :hasChild :Parent) :John)
EquivalentClasses(:MyBirthdayGuests **ObjectOneOf**(:Bill :John :Mary))

InverseObjectProperties(:hasParent :hasChild)

DisjointObjectProperties(:hasParent :hasSpouse)
SymmetricObjectProperty(:hasSpouse)
AsymmetricObjectProperty(:hasChild)
ReflexiveObjectProperty(:hasRelative)
IrreflexiveObjectProperty(:parentOf)
FunctionalObjectProperty(:hasHusband)
SubObjectPropertyOf(**ObjectPropertyChain**(:hasParent :hasParent) :hasGrandparent)

SubObjectPropertyOf(:hasParent :hasAncestor)
TransitiveObjectProperty(:hasAncestor)

HasKey(:Person () (:hasSSN))
Declaration(**NamedIndividual**(:John))
Declaration(**Class**(:Person))
Declaration(**ObjectProperty**(:hasWife))
Declaration(**DataProperty**(:hasAge))