Summary of Syntax Validity of SPARQL Queries

Concept	Code	Valid Syntax	Parser Output

rdfs:Resource	PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
	ASK {
	  SELECT (count(*) as ?count)
	  WHERE {
	  ?s a rdfs:Resource .
	  }
	  GROUP BY ?s
	  HAVING (COUNT(?s) = 1)
	}
		True	None

rdfs:Literal	PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
	SELECT (COUNT(?literal) AS ?literal_count)
	WHERE {
	  ?literal a rdfs:Literal .
	}
		True	None

rdfs:Class	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
	
	SELECT (COUNT(*) as ?count)
	WHERE {
	  ?class rdf:type rdfs:Class .
	}
	GROUP BY ?class
	HAVING (COUNT(*) = 1)
		True	None

rdfs:Datatype	PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
	SELECT (COUNT(?datatype) as ?datatypeCount)
	WHERE {
	  ?datatype a rdfs:Datatype .
	}
	HAVING (COUNT(?datatype) = 1)
		True	None

rdfs:Container	PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
	SELECT (COUNT(?container) AS ?containerCount)
	WHERE {
	  ?container rdf:type rdfs:Container .
	}
		False	QueryBadFormed: A bad request has been sent to the endpoint: probably the SPARQL query is badly formed.
			
			b'Error 400: Parse error: \
			nPREFIX rdfs:
			<http://www.w3.org/2000/01/rdf-schema#>\nSELECT (COUNT(?container) AS ?containerCount) \
			nWHERE {\n ?container rdf:type rdfs:Container .\
			n}\n\rLine 4, column 14: Unresolved prefixed name: rdf:type\n'

rdfs:label	PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
	SELECT DISTINCT ?s
	WHERE {
	  ?s rdfs:label ?label.
	  FILTER NOT EXISTS {
	  ?s rdfs:label ?otherLabel.
	  FILTER (?otherLabel != ?label)
	  }
	}
		True	None

rdfs:comment	PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
	
	SELECT ?resource
	WHERE {
	  ?resource rdfs:comment ?comment .
	  FILTER NOT EXISTS {
	  ?resource rdfs:comment ?otherComment .
	  FILTER (?otherComment != ?comment)
	  }
	}
		True	None

rdfs:domain	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> PREFIX owl: <http://www.w3.org/2002/07/owl#> SELECT ?property (COUNT(?domain) AS ?domainCount) WHERE {	True	None

	?property rdf:type owl:ObjectProperty . ?property rdfs:domain ?domain . } GROUP BY ?property HAVING (COUNT(?domain) = 1)		

rdfs:range	PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
	SELECT (count(*) as ?count)
	WHERE {
	?property rdfs:range ?range .
	}
	GROUP BY ?property
	HAVING (count(*) = 1)
		True	None

rdfs:subClassOf	PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
	PREFIX owl: <http://www.w3.org/2002/07/owl#>
	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	
	SELECT ?class (COUNT(?parent) AS ?count)
	WHERE {
	?class rdfs:subClassOf ?parent .
	}
	GROUP BY ?class
	HAVING (COUNT(?parent) = 1)
		True	None

skos:Concept	PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
	
	ASK {
	?c a skos:Concept .
	FILTER NOT EXISTS {
	?c a skos:Concept .
	FILTER (?c != ?c2 || ?c2 != ?c)
	}
	}
		True	None

skos:Collection	PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
	
	SELECT (COUNT(*) AS ?count)
	WHERE {
	?s a skos:Collection .
	}
	HAVING (COUNT(*) = 1)
		True	None

skos:ConceptScheme	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
	
	ASK {
	{
	SELECT (COUNT(?scheme) AS ?count)
	WHERE {
	?scheme rdf:type skos:ConceptScheme .
	}
	}
	FILTER(?count = 1)
	}
		True	None

skos:OrderedCollection	PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
	
	SELECT (COUNT(?oc) as ?ocCount)
	WHERE {
	?oc a skos:OrderedCollection .
	}
	HAVING (COUNT(?oc) = 1)
		True	None

skos:CollectableProperty	PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
	
	SELECT (COUNT(?cp) AS ?numCollectableProperty)
	WHERE {
	GRAPH <mygraph> {
	?cp a skos:CollectableProperty .
	}
	}
	HAVING (?numCollectableProperty = 1)
		True	None

skos:topConceptOf	PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
	
	SELECT ?concept (COUNT(?topConcept) AS ?count)
	WHERE {
	?concept skos:topConceptOf ?topConcept .
	}
	GROUP BY ?concept
	HAVING (COUNT(?topConcept) = 1)
		True	None

skos:semanticRelation	PREFIX skos: <http://www.w3.org/2004/02/skos/core#> SELECT ?x WHERE { ?x skos:semanticRelation ?y .	True	None

	FILTER NOT EXISTS {
	?x skos:semanticRelation ?z .
	FILTER (?z != ?y)
	}
	}		

skos:memberList	SPARQL
	PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
	
	ASK {
	?c skos:memberList ?list .
	NOT EXISTS {
	?c skos:memberList ?otherList .
	FILTER (?list != ?otherList)
	}
	}
		False	QueryBadFormed: A bad request has been sent to the endpoint: probably the SPARQL query is badly formed.
			
			b'Error 400: Parse error: \nSPARQL\nPREFIX skos: <http://www.w3.org/2004/02/skos/core#>\n\nASK {\n ?c skos:memberList ?list .\n NOT EXISTS { \n ?c skos:memberList ?otherList .\n FILTER (?list != ?otherList)\n }\n}\n\rLexical error at line 1, column 7. Encountered: "\\n" (10), after : "SPARQL"\n'

skos:member	PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
	
	SELECT ?concept (COUNT(?member) AS ?count)
	WHERE {
	?concept skos:member ?member .
	}
	GROUP BY ?concept
	HAVING (COUNT(?member) = 1)
		True	None

skos:hasTopConcept	PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
	
	SELECT (COUNT(?topConcept) AS ?count)
	WHERE {
	?scheme skos:hasTopConcept ?topConcept .
	}
	GROUP BY ?scheme
	HAVING (COUNT(?topConcept) = 1)
		True	None

foaf:Agent	PREFIX foaf: <http://xmlns.com/foaf/0.1/>
	
	ASK {
	?agent a foaf:Agent .
	FILTER NOT EXISTS {
	?agent a [a foaf:Agent ; !sameTerm(?agent)] .
	}
	}
		False	QueryBadFormed: A bad request has been sent to the endpoint: probably the SPARQL query is badly formed.
			
			b'Error 400: Parse error: \nPREFIX foaf: <http://xmlns.com/foaf/0.1/>\n\nASK {\n ?agent a foaf:Agent .\n FILTER NOT EXISTS {\n ?agent a [a foaf:Agent ; !sameTerm(?agent)] .\n }\n}\n\rEncountered "
			"sameTerm" "sameTerm
			"" at line 6, column 32.\nWas expecting one of:\n <IRIref> ...\n <PNAME_NS> ...\n <PNAME_LN> ...\n "a" ...\n "(" ...\n "^" ...\n\n'

foaf:Person	PREFIX foaf: <http://xmlns.com/foaf/0.1/>
	SELECT (COUNT(?person) AS ?count)
	WHERE {
	?person rdf:type foaf:Person .
	}
		False	QueryBadFormed: A bad request has been sent to the endpoint: probably the SPARQL query is badly formed.

			b'Error 400: Parse error: \ nPREFIX foaf: <http://xmlns.com/foaf/0.1/ >\nSELECT (COUNT(? person) AS ?count)\ nWHERE {\n ?person rdf:type foaf:Person .\n}\n\ rLine 4, column 11: Unresolved prefixed name: rdf:type\n'

foaf:Document	PREFIX foaf: <http://xmlns.com/foaf/0.1/>
	
	SELECT (COUNT(*) as ?count)
	WHERE {
	?document a foaf:Document .
	}
	HAVING (?count = 1)
		True	None

foaf:Group	PREFIX foaf: <http://xmlns.com/foaf/0.1/>
	
	ASK {
	SELECT (COUNT(?group) AS ?count)
	WHERE {
	?group rdf:type foaf:Group .
	}
	GROUP BY ?group
	HAVING (COUNT(?group) = 1)
	}
		False	QueryBadFormed: A bad request has been sent to the endpoint: probably the SPARQL query is badly formed.
			
			b'Error 400: Parse error: \ nPREFIX foaf: <http://xmlns.com/foaf/0.1/ >\n\nASK {\n SELECT (COUNT(?group) AS ? count)\n WHERE {\n ? group rdf:type foaf:Group .\n }\n GROUP BY ?group\n HAVING (COUNT(?group) = 1)\n}\n\rLine 6, column 12: Unresolved prefixed name: rdf:type\n'

foaf:OnlineAccount	PREFIX foaf: <http://xmlns.com/foaf/0.1/>
	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	ASK {
	?person foaf:onlineAccount ?account1 .
	{ SELECT (COUNT(?account2) AS ?count)
	WHERE { ?person foaf:onlineAccount ?account2 . }
	}
	FILTER (?count = 1)
	}
		True	None

foaf:name	PREFIX foaf: <http://xmlns.com/foaf/0.1/>
	SELECT ?s WHERE {
	?s foaf:name ?name .
	FILTER NOT EXISTS {
	?s foaf:name ?otherName .
	FILTER (?name != ?otherName)
	}
	}
		True	None

foaf:knows	PREFIX foaf: <http://xmlns.com/foaf/0.1/>
	SELECT ?person
	WHERE {
	?person foaf:knows ?friend .
	FILTER NOT EXISTS {
	?person foaf:knows ?otherFriend .
	FILTER (?friend != ?otherFriend)
	}
	}
		True	None

foaf:based_near	PREFIX foaf: <http://xmlns.com/foaf/0.1/>
	
	SELECT (COUNT(*) AS ?count)
	WHERE {
	?person foaf:based_near ?location .
	}
	GROUP BY ?person
	HAVING (COUNT(*) = 1)
		True	None

foaf:accountName	PREFIX foaf: <http://xmlns.com/foaf/0.1/> ASK { ?person foaf:accountName ?name1 .	True	None

	?person foaf:accountName ?name2 . FILTER (?name1 != ?name2) }		

foaf:holdsAccount	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	PREFIX foaf: <http://xmlns.com/foaf/0.1/>
	
	ASK {
	?person foaf:holdsAccount ?account1 .
	?person foaf:holdsAccount ?account2 .
	FILTER (?account1 != ?account2)
	}
		True	None

prov:Agent	PREFIX prov: <http://www.w3.org/ns/prov#>
	
	SELECT (COUNT(?agent) AS ?count)
	WHERE {
	?agent a prov:Agent .
	}
		True	None

prov:Activity	PREFIX prov: <http://www.w3.org/ns/prov#>
	
	SELECT (COUNT(?activity) AS ?activityCount)
	WHERE {
	?activity a prov:Activity .
	}
	GROUP BY ?activity
	HAVING (COUNT(?activity) = 1)
		True	None

prov:Entity	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	PREFIX prov: <http://www.w3.org/ns/prov#>
	
	SELECT (COUNT(?entity) AS ?count)
	WHERE {
	?entity rdf:type prov:Entity .
	}
	HAVING (?count = 1)
		True	None

prov:Collection	PREFIX prov: <http://www.w3.org/ns/prov#>
	
	ASK {
	{
	SELECT (COUNT(*) AS ?count)
	WHERE {
	?s a prov:Collection .
	}
	}
	FILTER (?count = 1)
	}
		True	None

prov:Location	PREFIX prov: <http://www.w3.org/ns/prov#>
	SELECT ?s WHERE {
	?s prov:Location ?o1 .
	FILTER NOT EXISTS {
	?s prov:Location ?o2 .
	FILTER (?o1 != ?o2)
	}
	}
		True	None

prov:wasGeneratedBy	PREFIX prov: <http://www.w3.org/ns/prov#>
	
	SELECT ?subject
	WHERE {
	?subject prov:wasGeneratedBy ?value .
	FILTER NOT EXISTS {
	?subject prov:wasGeneratedBy ?otherValue .
	FILTER (?value != ?otherValue)
	}
	}
		True	None

prov:generatedAtTime	PREFIX prov: <http://www.w3.org/ns/prov#>
	PREFIX ex: <http://example.com/>
	
	SELECT DISTINCT ?s WHERE {
	?s prov:generatedAtTime ?t .
	FILTER NOT EXISTS {
	?s prov:generatedAtTime ?t2 .
	FILTER (?t != ?t2)
	}
	}
		True	None

prov:atLocation	PREFIX prov: <http://www.w3.org/ns/prov#>
	SELECT ?instance
	WHERE {
	?instance prov:atLocation ?location1 .
	FILTER NOT EXISTS {
	?instance prov:atLocation ?location2 FILTER (?
	location1 != ?location2)
		True	None

	} }		

prov:hadMember	PREFIX prov: <http://www.w3.org/ns/prov#>
	
	SELECT ?group (COUNT(?member) AS ?count)
	WHERE {
	?group prov:hadMember ?member .
	}
	GROUP BY ?group
	HAVING (COUNT(?member) = 1)
		True	None

prov:used	PREFIX prov: <http://www.w3.org/ns/prov#>
	
	SELECT ?entity
	WHERE {
	?activity prov:used ?entity .
	}
	GROUP BY ?activity
	HAVING (COUNT(?entity) = 1)
		False	QueryBadFormed: A bad request has been sent to the endpoint: probably the SPARQL query is badly formed.
			
			b'Error 400: Parse error: \nPREFIX prov: <http://www.w3.org/ns/prov#>\n\nSELECT ?entity\nWHERE {\n ?activity prov:used ?entity .\n}\nGROUP BY ?activity\nHAVING (COUNT(?entity) = 1)\n\rNon-group key variable in SELECT: ?entity\n'

rdf:List	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	
	SELECT (COUNT(*) as ?count)
	WHERE {
	?list a rdf:List .
	}
	HAVING (COUNT(*) = 1)
		True	None

rdf:Bag	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	
	SELECT (COUNT(?bag) AS ?count)
	WHERE {
	?bag rdf:type rdf:Bag .
	}
		True	None

rdf:Statement	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	SELECT (COUNT(*) as ?count)
	WHERE {
	?subject ?predicate ?object .
	FILTER(isURI(?subject) && isURI(?predicate) && isURI(?object))
	FILTER(!isBlank(?subject) && !isBlank(?predicate) && !isBlank(?object))
	FILTER(?predicate = rdf:subject || ?predicate = rdf:predicate || ?predicate = rdf:object)
	}
	GROUP BY ?predicate
	HAVING (COUNT(*) = 1 && ?predicate = rdf:Statement)
		True	None

rdf:Property	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	
	SELECT (COUNT(*) as ?count) WHERE {
	?s rdf:type rdf:Property.
	}
	
	HAVING(?count = 1)
		True	None

rdf:HTML	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	PREFIX owl: <http://www.w3.org/2002/07/owl#>
	
	ASK {
	?s rdf:type owl:Class .
	?s owl:equivalentClass ?c .
	?c owl:onProperty rdf:HTML .
	?c owl:cardinality '1'^^xsd:nonNegativeInteger .
	}
		False	QueryBadFormed: A bad request has been sent to the endpoint: probably the SPARQL query is badly formed.
			
			b"Error 400: Parse error: \nPREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>\nPREFIX owl: <http://www.w3.org/2002/07/owl#>\n\nASK {\n ?s

			rdf:type owl:Class .\n ?s owl:equivalentClass ?c .\n ?c owl:onProperty rdf:HTML .\n ?c owl:cardinality '1'^^xsd:nonNegativeInteger .\n}\n\rLine 8, column 27: Unresolved prefixed name: xsd:nonNegativeInteger\n"

rdf:first	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
	
	ASK {
	  ?s rdf:first ?first .
	  FILTER NOT EXISTS { ?s rdf:first ?other . FILTER (?first != ?other) }
	}
		True	None

rdf:type	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	SELECT (COUNT(?type) AS ?count)
	WHERE {
	  ?s rdf:type ?type .
	}
	HAVING (COUNT(?type) = 1)
		True	None

rdf:subject	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	PREFIX example: <http://example.com/>
	
	SELECT ?s (COUNT(?s) AS ?count)
	WHERE {
	  ?s rdf:subject example:exampleConcept .
	}
	GROUP BY ?s
	HAVING (COUNT(?s) = 1)
		True	None

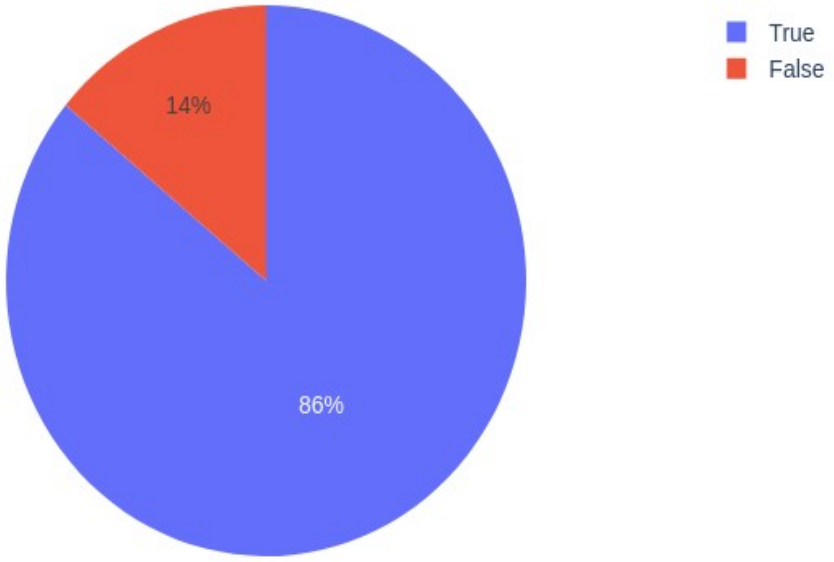
rdf:value	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	PREFIX owl: <http://www.w3.org/2002/07/owl#>
	
	SELECT ?subject
	WHERE {
	  ?subject rdf:value ?value .
	  FILTER NOT EXISTS {
	  ?subject rdf:value ?otherValue .
	  FILTER (?value != ?otherValue)
	  }
	}
		True	None

rdf:rest	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
	
	SELECT (COUNT(?rest) AS ?count)
	WHERE {
	  ?s rdf:rest ?rest .
	}
		True	None

The table below shows the occurrences of each category for the tested ontologies.

Ontology Prefix	True Count	False Count	No Response Count
rdfs	9	1	0
skos	9	1	0
foaf	7	3	0
prov	9	1	0
rdf	9	1	0

Figure 1: Occurences of each category of response ('True', 'False', 'None')



The table below shows an overview of the results for each category of tested code.

Code Category	True Count	False Count	No Response Count
Instance Graphs	43	7	0
Shacl Shapes	46	4	0
Sparql Queries	43	7	0