

RDF数据:



SELECT ?givenName

WHERE

```
{ ?y <http://www.w3.org/2001/vcard-rdf/3.0#Family> "Smith" .  
  ?y <http://www.w3.org/2001/vcard-rdf/3.0#Given> ?givenName .  
}
```

Prefix

PREFIX vcard: <<http://www.w3.org/2001/vcard-rdf/3.0#>>

SELECT ?givenName

WHERE

```
{ ?y vcard:Family "Smith" .  
  ?y vcard:Given ?givenName .  
}
```

	givenName
1	"Rebecca"
2	"John"

Blank Nodes

空节点：以 "_" 开头

PREFIX vcard: <http://www.w3.org/2001/vcard-rdf/3.0#>

```
SELECT ?y ?givenName
WHERE
{ ?y vcard:Family "Smith" .
  ?y vcard:Given ?givenName .
}
```

	y	givenName
1	_:b0	"John"
2	_:b1	"Rebecca"

FILTER

过滤掉符合模式但是不想要的结果，选择符合特定标准的数据，运算符如下：

逻辑：&&, ||, !

数学：+, -, *, /

比较：=, !=, <, >, <=, >=

SPARQL测试：isURI, isBlank, isLiteral, isNumeric, bound

SPARQL访问器：str, lang, datatype

字符串匹配：

FILTER regex(?x, "pattern" [, "flags"]): flags 是 "i" , 忽略大小写

查询名字中包含 "R" 或 "r" 的项：

PREFIX vcard: <http://www.w3.org/2001/vcard-rdf/3.0#>

```
SELECT ?g
WHERE
{ ?y vcard:Given ?g .
  FILTER regex(?g, "r", "i") }
```

givenName

1 "Rebecca"

2 "Sarah"

值匹配:

PREFIX info: <http://somewhere/peopleInfo#>

```
SELECT ?resource
WHERE
{
  ?resource info:age ?age .
  FILTER (?age >= 24)
}
```

OPTIONAL

使得当某一项不存在时查询不至于失败

如果组匹配，则扩展解决方案，如果不匹配，则给出原始解决方案

查询获取 name 和 age（仅当age 存在时才获取，age 不存在只获取 name）：


PREFIX info: <http://somewhere/peopleInfo#>

PREFIX vcard: <http://www.w3.org/2001/vcard-rdf/3.0#>

```
SELECT ?name ?age
WHERE
{
```

```
?person vcard:FN ?name .
OPTIONAL { ?person info:age ?age }
```


```
}
```

	name	 age
1	"John Smith"	"25"^^xsd:integer
2	"Becky Smith"	"23"^^xsd:integer
3	"Sarah Jones"	
4	"Matt Jones"	

OPTIONAL with FILTER

```
PREFIX info:    <http://somewhere/peopleInfo#>
PREFIX vcard:  <http://www.w3.org/2001/vcard-rdf/3.0#>
```

```
SELECT ?name ?age
WHERE
{
  ?person vcard:FN ?name .
  OPTIONAL { ?person info:age ?age . FILTER ( ?age > 24 ) }
}
```

	name	 age
1	"John Smith"	"25"^^xsd:integer
2	"Becky Smith"	
3	"Sarah Jones"	
4	"Matt Jones"	


```
PREFIX info:    <http://somewhere/peopleInfo#>
PREFIX vcard:  <http://www.w3.org/2001/vcard-rdf/3.0#>
```

```
SELECT ?name ?age
WHERE
{
```

```

?person vcard:FN ?name .
OPTIONAL { ?person info:age ?age . }
FILTER ( !bound(?age) || ?age > 24 )
}

```

	name	 age
1	"John Smith"	"25"^^xsd:integer
2	"Sarah Jones"	
3	"Matt Jones"	

"Becky Smith" < 24.

UNION

合并

PREFIX foaf: <http://xmlns.com/foaf/0.1/>

PREFIX vCard: <http://www.w3.org/2001/vcard-rdf/3.0#>

SELECT ?name

WHERE

```

{
  { [] foaf:name ?name } UNION { [] vCard:FN ?name }
}

```

	name
1	"John Smith"
2	"Becky Smith"
3	"Sarah Jones"
4	"Matt Jones"

等价:

PREFIX foaf: <http://xmlns.com/foaf/0.1/>

PREFIX vCard: <http://www.w3.org/2001/vcard-rdf/3.0#>

SELECT ?name

```

WHERE
{
  [] ?p ?name
  FILTER ( ?p = foaf:name || ?p = vCard:FN )
}

```

记录数据来源

PREFIX foaf: <http://xmlns.com/foaf/0.1/>

PREFIX vCard: <http://www.w3.org/2001/vcard-rdf/3.0#>

```

SELECT ?name1 ?name2
WHERE
{
  { [] foaf:name ?name1 } UNION { [] vCard:FN ?name2 }
}

```

name1	name2
"Matt Jones"	
"Sarah Jones"	
	"Becky Smith"
	"John Smith"

OPTIONAL and UNION

OPTIONAL有助于增加找到的解决方案，UNION有益于从两种可能性中合并解决方案

PREFIX foaf: <http://xmlns.com/foaf/0.1/>

PREFIX vCard: <http://www.w3.org/2001/vcard-rdf/3.0#>

```

SELECT ?name1 ?name2
WHERE
{
  ?x a foaf:Person
  OPTIONAL { ?x foaf:name ?name1 }
  OPTIONAL { ?x vCard:FN ?name2 }
}

```

name1	name2
"Matt Jones"	
"Sarah Jones"	
	"Becky Smith"
	"John Smith"

ORDER BY

排序

ORDER BY DESC: 降序

SELECT ?name

WHERE

```
{
    ?x foaf:name ?name
}
```

ORDER BY ?name

DISTINCT

唯一, 去重

PREFIX foaf: <<http://xmlns.com/foaf/0.1/>>

SELECT DISTINCT ?name

WHERE

```
{
    ?x foaf:name ?name
}
```

OFFSET

查询结果偏移

LIMIT

限制返回结果数量

```
SELECT ?name
WHERE { ?x foaf:name ?name }
ORDER BY ?name
LIMIT 5
OFFSET 10
```