第七次实验

58119125
蒋卓洋

一．实验要求：

针对下列自然语言，使用 SPARQL 查询语句返回结果:

1. Who are the creators of Guernica and Sunflowers, respectively?
2. List all the artists who live in Spain or other places.
3. List all paintings, their names, and the corresponding techniques.

二．要求分析：

1. 要求 1 SPARQL：

```
SELECT ?creator
WHERE {
    {?creator ex:creatorOf ex:sunflowers.}
    UNION{?creator ex:creatorOf ex:guernica.}
}
```

2. 要求 2 SPARQL：

```
SELECT ?artists
WHERE{
    ?artists a ex:Artist.
    OPTIONAL{
        ?artists ex:homeAddress ?x.
        ?x ex:country "Spain".
    }
}
```

3. 要求 3 SPARQL：

```
SELECT ?names ?techniques
WHERE{
    ?names a ex:Painting;
            ex:technique ?techniques.
}
```

三．代码实现：

1. 要求 1 代码实现：

```java
// We do a simple SPARQL SELECT-query that retrieves all resources of type `ex:Artist`,
// and their first names.
String queryString = "PREFIX ex: <http://example.org/> \n";
queryString += "PREFIX foaf: <" + FOAF.NAMESPACE + "> \n";
queryString += "SELECT ?creator \n";
queryString += "WHERE { \n";
queryString += "    {?creator ex:creatorOf ex:sunflowers.} \n";
queryString += "    UNION{?creator ex:creatorOf ex:guernica.}";
queryString += "}";

TupleQuery query = conn.prepareTupleQuery(queryString);

// A QueryResult is also an AutoCloseable resource, so make sure it gets closed when done.
try (TupleQueryResult result = query.evaluate()) {
    // we just iterate over all solutions in the result...
    for (BindingSet solution : result) {
        // ... and print out the value of the variable binding for ?s and ?n
        System.out.println("?creator = " + solution.getValue("creator"));
    }
}
```

图 1.要求 1 代码实现

2. 要求 2 代码实现：

```
// We do a simple SPARQL SELECT-query that retrieves all resources of type `ex:Artist`,
// and their first names.
String queryString = "PREFIX ex: <http://example.org/> \n";
queryString += "PREFIX foaf: <" + FOAF.NAMESPACE + "> \n";
queryString += "SELECT ?artists \n";
queryString += "WHERE { \n";
queryString += "    ?artists a ex:Artist. \n";
queryString += "    OPTIONAL{ \n";
queryString += "        ?artists ex:homeAddress ?x. \n";
queryString += "        ?x ex:country \"Spain\". \n";
queryString += "    }";
queryString += "}";

TupleQuery query = conn.prepareTupleQuery(queryString);

// A QueryResult is also an AutoCloseable resource, so make sure it gets closed when done.
try (TupleQueryResult result = query.evaluate()) {
    // we just iterate over all solutions in the result...
    for (BindingSet solution : result) {
        // ... and print out the value of the variable binding for ?s and ?n
        System.out.println("?artists = " + solution.getValue("artists"));
    }
}
```

图 2.要求 2 代码实现

3. 要求 3 代码实现：

```
// We do a simple SPARQL SELECT-query that retrieves all resources of type `ex:Artist`,
// and their first names.
String queryString = "PREFIX ex: <http://example.org/> \n";
queryString += "PREFIX foaf: <" + FOAF.NAMESPACE + "> \n";
queryString += "SELECT ?names ?techniques \n";
queryString += "WHERE { \n";
queryString += "    ?names a ex:Painting; \n";
queryString += "             ex:technique ?techniques.";
queryString += "}";

TupleQuery query = conn.prepareTupleQuery(queryString);

// A QueryResult is also an AutoCloseable resource, so make sure it gets closed when done.
try (TupleQueryResult result = query.evaluate()) {
    // we just iterate over all solutions in the result...
    for (BindingSet solution : result) {
        // ... and print out the value of the variable binding for ?s and ?n
        System.out.println("?names = " + solution.getValue("names"));
        System.out.println("?techniques = " + solution.getValue("techniques"));
    }
}
```

图 3.要求 3 代码实现

四. 实验结果：

1. 要求 1 实现结果：

图 4.要求 1 实现结果

2. 要求 2 实现结果：

图 5.要求 2 实现结果

3. 要求 3 实现结果：

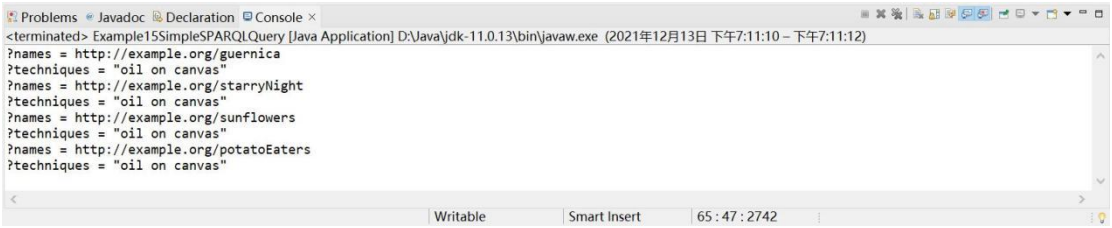

图 6.要求 3 实现结果