Foundations of Linked Data
Bonus Tasks

Task 1: Creating and Publishing Linked Data

Thomas Wehr, Prof. Dr. Andreas Harth

April 25, 2025

This is the first out of the three bonus tasks. All bonus tasks are voluntary. To pass this task, the correct solutions must be submitted until **May 18, 2025, 23:59h**. Students who pass all three tasks in time are awarded a 0.3 / 0.4 grade improvement in the final exam of the course.

Administrative Information

Following the inverted classroom principle, if you have problems or further questions, please post these into the **students forum on StudOn**. That way your colleagues can also contribute to and benefit from the answers.

1 Task: Description and Objective

The aim of this task is to make you familiar with creating and publishing RDF documents. This task will cover lectures 1 through 3 of the Foundations of Linked Data course.
The subtasks are ordered accordingly, so that you can step-by-step solve the task.

1.1 Set up programming environment

You are going to create RDF documents in Turtle Syntax using the file extension ".ttl". RDF documents are just plain text files. This means that you can use any text editor to create RDF documents. However, it is easier to write code within an integrated development environment (IDE) that supports syntax highlighting and debugging. We recommend to install Visual Code[1] and the Stardog RDF Grammar extension[2], but you are free to create your RDF document with other means.

[1] `https://code.visualstudio.com/`
[2] `https://marketplace.visualstudio.com/items?itemName=stardog-union.stardog-rdf-grammars`

1.2 Creating the first RDF document

The first RDF document you will create describes a player profile from the Bundesliga — a german football league. You must define appropriate terms when creating new triples, however, all URIs should reference to the file itself by using the relative hash URI. **Each item of the enumeration below describes one triple to be formulated (except the Bnode statements).** Use the datatype in parenthesis for the object. If none is given, declare an URI. For instance, the triples (a),(c), (f) and (F) could be formulated as following:

```
@prefix :       <#> .
@prefix xsd:    <http://www.w3.org/2001/XMLSchema#> .

:rose :name             "Max Rosenfelder"^^xsd:string ; # (a)
      :plays_for        :SCF ;                          # (c)
      :date_of_birth    "2003-02-10"^^xsd:date .        # (f)

:SCF  :name             "Sport Club Freiburg"@de .      # (B)
```

1.2.1 Player as the subject

Choose a Bundesliga player[3] and write the following triples:

(a) Player has a full name (string).
(b) Player's occupation is football player.
(c) Player plays for a team.
(d) Player has a sport number (int).
(e) Player plays on a specific position (choose one if multiple) .
(f) Player has a birthdate (date).
(g) Player has a nationality (choose one if multiple) .
(h) Player has a dominant foot.
(i) Player has a specific height (Blank Node with value and unit) .
(j) Player has a specific weight (Blank Node with value and unit) .

1.2.2 Team as the subject

Now add some triples about the team the player you chose plays for. Have a look at the clubs[4] to write down the following triples:

(A) Team plays in the Bundesliga.
(B) Team has a German name (RDFLiteral).
(C) Team has a home venue.
(D) Bundesliga has a German name (RDFLiteral).

[3] `https://www.bundesliga.com/en/bundesliga/player/`
[4] `https://www.bundesliga.com/en/bundesliga/clubs`

(E) Another team of your choice plays in the Bundesliga.
(F) The team above has a German name (RDFLiteral).

You must have a turtle file with 20 triples in total. Do not add any more triples as we need the exact number for validating your result. Save your RDF data as a file locally as "fld.ttl".

1.3 Publishing the RDF document

After you have created your RDF document, you want to publish it that other students can access your data. For that, you will need a Web server which we have already set up for you on our Paul server which is located at the RRZE. You can access your account on Paul using a terminal and the Secure Shell Protocol (ssh). If you are a Windows user, you should use Microsoft's PowerShell that should be already installed on Windows 10.

1. Open your preferred terminal
2. Type `ssh (idm)@paul.ti.rw.fau.de` and confirm with your idm password.
3. Type `pwd` to check your current location. It should be "/home/rzlin/(idm)/".
4. Type `mkdir public_html` to create the folder `public_html`.
5. Press ctrl + d to close the connection with Paul (d for disconnect).

The configured Web server publishes all files that are stored in the folder `public_html`, which means that you just need to copy your RDF document to Paul. For that we use the Secure Copy Protocol (scp).

1. Open your preferred terminal
2. Type `scp (filepath)/fld.ttl (idm)@paul.ti.rw.fau.de:/home/rzlin/(idm)/public_html/fld.ttl` and confirm with your idm password.
3. Open a Web browser like Mozilla Firefox
4. Enter `https://paul.ti.rw.fau.de/~(idm)/` as URL. You should be able to see and download your copied RDF document.

You have managed to publish your first RDF document to the world. Congratulations!

OPTIONAL: If you want restrict access to FAU members only, you can use a prepared **.htaccess** file. The file can be found on StudOn and must not be changed. Copy it to the public_html folder with the secure copy protocol. After that, the file `fld.ttl` is only visible for user agents within the FAU VPN network.

1.4 Upload reference file to StudOn

We will track your progress on the bonus assignments on StudOn. For task 1, you need to upload a file so that we can grade you. Thus, upload a text file with the name "(idm).txt" to StudOn contaning a single line with the full hash URI of the Bundesliga in your RDF document. For example, if your RDF document is called "doc.ttl", hosted on our Paul Server, and contains the following triple:

```
:bob    :knows        :alice .
```

Then the hash URI of Bob in your RDF document is `https://paul.ti.rw.fau.de/~[YOUR IDM]/doc.ttl#bob`.

Key Takeaways of this Task

At the end of this task, you should be able to create RDF documents. Additionally, you used basics of `ssh` and `scp` in a terminal to access the Paul server. After publishing data, you will query data in the next task.