**Competency Questions**

Q1: What is the course SOEN6431 about?

Q2: Which topics is Manthan competent in?

Q3: Which topics were covered in the first lecture of SOEN6431?

Q4: Which subjects are offered by SOEN department?

Q5: Show me the subjects related to Finance.

Q6: How many subjects did Jaynil studied?

Q7: Provide the worksheets of 3rd and 4th week in COMP6741.

Q8: Provide me the 4th lectures detail of SOEN 6431.

Q9: In which subjects Jaynil scored grade A?

Q10: Which lecture covers "SPARQL"?

**Vocabulary**

There are a variety of vocabularies accessible for various reasons. We used several vocabularies to build the knowledge graph, like DBP, FOAF, DC, RDF, and RDFS. We utilised foaf:name from the FOAF ontology to define the entity in the knowledge graph that is related to individuals. We used DBP as a prefix to attach a dbpedia link in triples, and this DBP is used in many places, such as topics of lectures. To specify the course information and course number, the DC vocabulary's dc:description and dc:identifier attributes are applied. We also used the rdfs:seeAlso property to add course resources like an overview, as well as the rdfs:comment property to offer a summary of the lecture specifics.

To build a knowledge graph, we applied FOAF, RDFS, and RDF Ontologies to define custom classes and properties. We developed the uni:University, uni:Course, uni:Lecture, and uni:Student classes from rdfs:Class to describe details about university, course, subject, lecture, and student. We made different properties such as uni:creditIs, uni:subjectOf, uni:slideIs, uni:topicIs, uni:grade and uni:idnumberIs from rdf:property to attach details about additional information about students and courses. To add a description of classes and properties, the RDFS vocabulary terms rdfs:comment, rdfs:domain, and rdfs:range are used.

**Knowledge Base Construction**

We used two CSV files from the https://opendata.concordia.ca/datasets/ to construct the knowledge base of courses offered by the university: CU SR OPEN DATA CATALOG-47958298.csv and CU SR OPEN DATA CATALOG DESC-45649197.csv. First, we combined these two files because one contains a list of courses and the other includes descriptions of these courses. Then, using pandas, we read this merged CSV file. This file was iterated to add triples of the course's name, department, code, credit, and description. We also include all the data from two courses, COMP6741 and SOEN6431, including outlines, slides, and worksheets. We created a course folder and put data of these two courses in it. To add the links of slides and worksheets we used the “OS” library and added the triples to the knowledge graph.

In the knowledge graph, we additionally inserted topics (with DBpedia links) for each lecture. We used SPOTLIGHT to produce the links for each topic and created a CSV file with all the topics and DBpedia URLs. We added topics to lecture triples by using this CSV file. Two students were added to the knowledge graph, with triples of name, identification number, email, and a list of completed courses with grades.

**Queries**

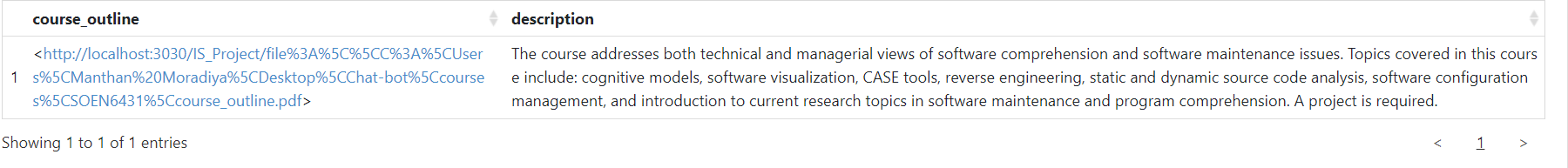
**QUE 1: what is the course SOEN6431 about?**

In this query, we select the course, which is the part of **SOEN** subject and has the identifier **6431**. We displayed the description of this course, and we also select the outline if it is available. For that we used **OPTIONAL** keyword.

Text

Description automatically generated

**Output:**



**QUE 2: Which topics is Manthan competent in?**

In this query, we selected the list of **course** completed by student **Manthan** and retrieved the **topics** of each lecture of these subjects.

Text

Description automatically generated with low confidence

**Output:**

Graphical user interface, text, application, email

Description automatically generated

**QUE 3: Which topics were covered in the first lecture of SOEN6431?**

In this query, first we selected course, which is part of **SOEN6431,** and which have **identifier 1 (Lecture 1)** and displayed the **topics** of this lecture.

Graphical user interface, text

Description automatically generated

**Output:**

Graphical user interface, text, application, email

Description automatically generated

**QUE 4: Which subjects are offered by SOEN department?**

In this query, we selected the course, which is the part of **SOEN** department and displayed the list of this **course name**.

**Graphical user interface, text

Description automatically generated**

**Output:**

Graphical user interface, text, application, email

Description automatically generated

**QUE 5: Show me the subjects related to Finance.**

In this query, we selected the name of all **subjects** available in knowledge base and by using **FILETER** keyword, only displayed the **course name** which contains **Finance** keyword.

Graphical user interface, text

Description automatically generated

**Output:**

Graphical user interface, text, application, email

Description automatically generated

**QUE 6: How many subjects did Jaynil studied?**

In this query, we selected the list of **completed subjects** of student whose name is **Jaynil** and count the total number of subjects using **COUNT** keyword.

**Text

Description automatically generated**

**Output:**

Graphical user interface, application

Description automatically generated

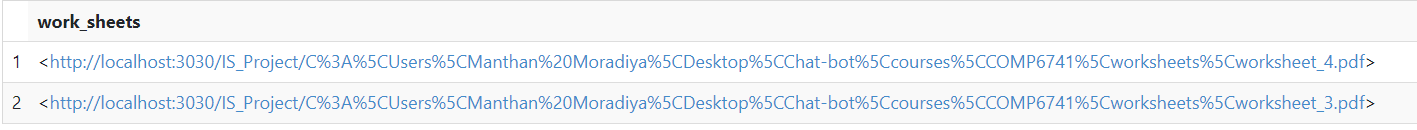
**QUE 7: Provide the worksheets of 3rd and 4th week in COMP6741.**

In this query, we selected the lectures of **COMP6741** and displayed the link of worksheets of **week 3** and **week 4**. To select the two lectures we used **UNION** keyword.

**Text

Description automatically generated**

**Output:**

****

**QUE 8: Provide me the 4th lectures detail of SOEN 6431.**

In this query, we selected the lecture of **SOEN6431**, whose **lecture number is 4** and displayed the **slides** and **topics** of this lecture. We also select **worksheet** if it is available using **OPTIONAL** keyword.

**Text

Description automatically generated**

**Output:**

**A picture containing application

Description automatically generated**

**QUE 9: In which subjects Jaynil scored grade A?**

In this query, we selected the **list of subjects completed** by student **Jaynil** and has **grade A**. With the use of this selected subject number, we displayed the **name of this subject** using **FILTER** and **contains** keyword.

**A picture containing timeline

Description automatically generated**

**Output:**

**Graphical user interface

Description automatically generated with low confidence**

**QUE 10: Which lecture covers "SPARQL"?**

In this query, we selected the lectures in which **SPARQL** topis is **covered** and displayed the **name** of this lectures.

**Text

Description automatically generated**

**Output:**

**Graphical user interface, text, application

Description automatically generated**