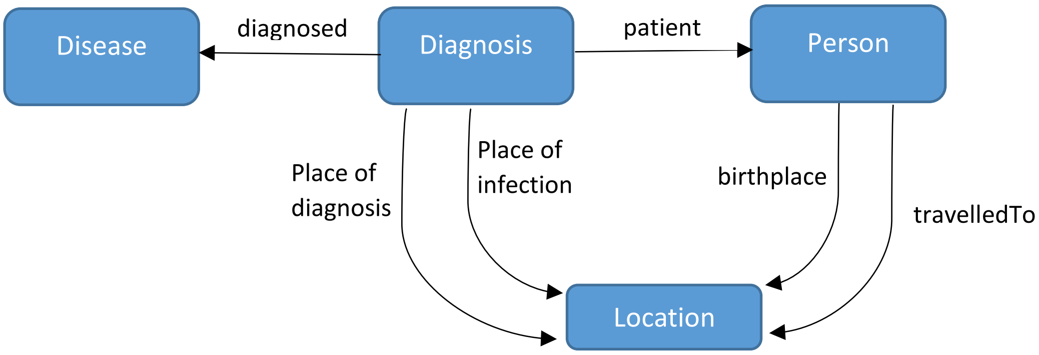
# OPEN DATA EXAM

**6th of June 2019.** *The exam will take 2 hours. Answer each question in the provided space.* Answers out of such space will not be considered. You are only allowed to have a pen and the papers provided by the lecturers on the table.

Name: …………………………………………………………………………………………………………………………….

### Question 1. Property Graphs [3p]



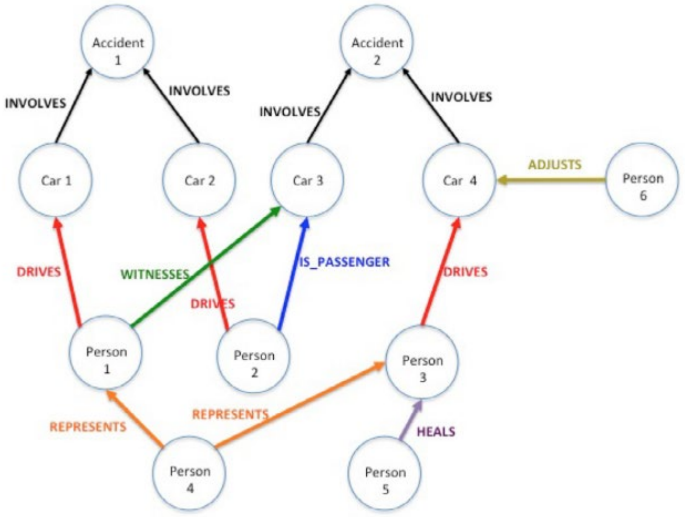
a) Consider the graph above that represents the schema that **all instances** in the graph would follow. Write a pattern to retrieve *all people infected with Chagas in a location where they did travel*. Assume disease contains a *name* attribute. Use the syntax we saw in the lectures to express the pattern **[1p]**

```
MATCH (:Disease {name: 'Chagas'})<-[:diagnosed]-(diag:Diagnosis)-[:patient]
  ->(p:Person)-[:travelledTo]->(:Location)<-[:Place of infection]-(diag:Diagnosis)
RETURN p
```

b) Sketch a **correct subgraph** of instances **matching** the following reachability pattern given below **[1p]**

(d:disease)-[e: diagnosed$^{-}$ ∘ patient]-(p:person)

c) An insurance company wants to develop a graph database to perform entity analysis. They want to detect fraud cases. As such, they develop the following graph schema **[1p]**



An accident has N cars and M people involved and they want to know **who and what cars participated in each accident**. This information is extremely relevant for the company and must not accept any kind of ambiguities. One of them, though, argues that this graph is not correct because it yields ambiguity about who participated in each accident. Is this person right? Justify your answer.

Finding fraud rings with a graph database becomes a simple question of walking the graph. Because graph databases are designed to query intricate connected networks, they can be used to identify fraud rings in a fairly straightforward fashion. Graph database queries can be added to the insurance company's standard checks, at appropriate points in time – such as when the claim is filed – to flag suspected fraud rings in real time.

**Question 2. Triplestores [1p]**

Assume a triplestore with the following indexes: SP, PO (where S stands for subject, P for predicate and O for Object). The following query arrives:

```
SELECT ?s ?o

WHERE ?s :p ?o
```

Will the triplestore use Index-Only Query Answering to answer this query? Justify your answer.

## Question 3. Knowledge Graphs [3p]

Consider the following relational data source:

Recipe(idR, name)

Ingredient(idI, name, description)

User (idU, location, alias)

Uses(idRecipe, idIngredient, qty, metric), where idRecipe is a FK on Recipe(idR) and idIngredient is a FK on Ingredient(idI)

- CHECK qty is a natural number
- CHECK metric IN ('spoon', 'gr', 'cup', 'unit', 'ml', 'scoop')

Loves(idUser, idIngredient), where idIngredient is a FK on Ingredient(idI) and idUser is a FK on User(idU)

Rates(idUser, idRecipe, rates), where idRecipe is a FK on Recipe (idR) and idUser is a FK on User(idU)

- CHECK rates IN 1 to 5

a) Use RDFS to represent a **TBOX modelling the above schema**. Draw it as an RDFS graph. Draw the TBOX concepts with rectangles. Your TBOX must capture as much semantics as possible. Clearly identify in the graph the RDFS constructs by using their proper URIs and define your own namespace prefix for the URIs you need to create. Assume the regime entailment is on. Thus, there is no need to relate your model to the RDFS core classes **[1,25p]**

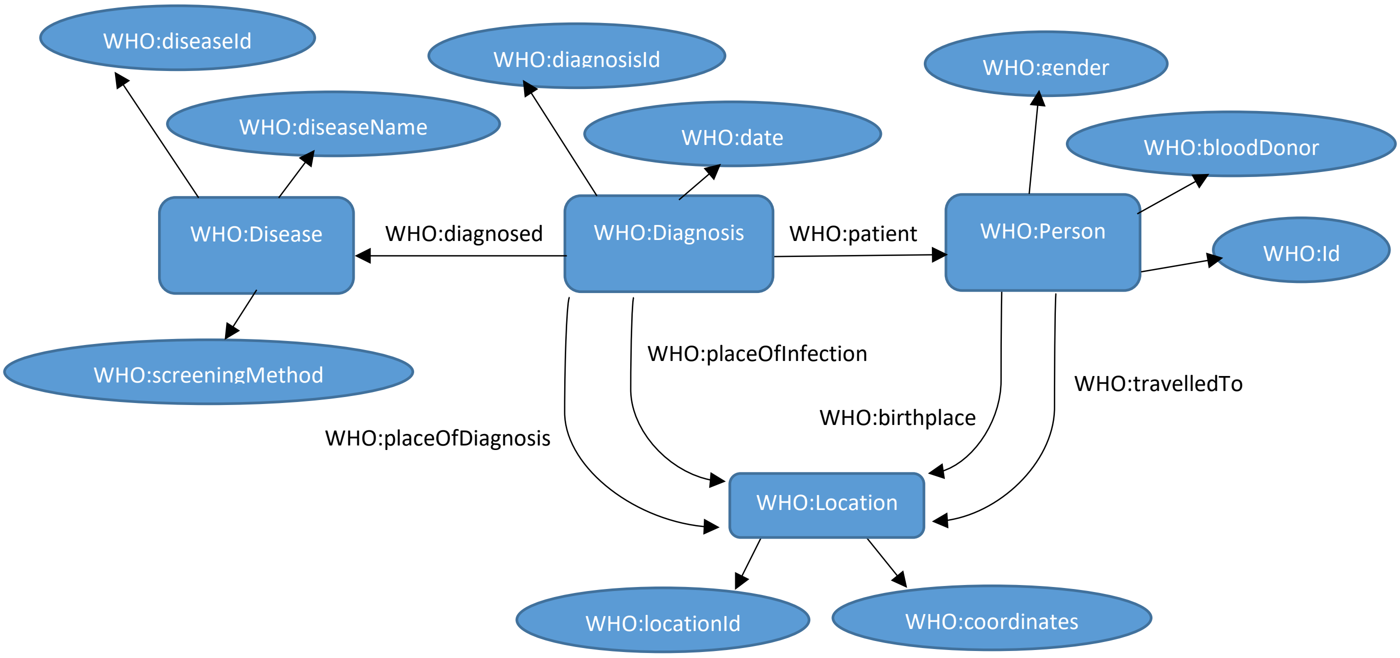
b) Now, **assert the following instances** to your RDF graph (draw them by the RDF graph sketched above and **separate the instances from the TBOX by a dashed line**): *Banana split is a recipe that uses 2 banana units and 1 scoop of chocolate. John is a user that rates banana split with 5* **[0,75p]**

c) Represent in DL the following additional semantics (use the same URIs as in the previous sections). *There are two kind of users, amateurs and professionals. However, if a user is amateur s/he cannot be considered professional. In addition, all users must be either amateur or professional* **[0,5p]**

Now, express the above DL expressions in OWL **[0,5p]**

**Question 4. Data Integration [3p]**

Given the following target schema. Concepts are represented with squares and attributes with circles. Assume the relationship between concepts and attributes is always of the form WHO:hasX, where X is the URN of the name (e.g., WHO:Diagnosis WHO:hasDate WHO:date):



And the following data sources exposed with wrappers:

Wrapper1(diseaseId, diseaseName, screeningMethod)

Wrapper2 (personId, personGender, diseaseId, diseaseName)

Wrapper3(diagnosisId, DiagnosisDate, locationOfDiagnosisId, locationCoordinates, diseaseId, diseaseName)

a) Write the GAV **exact mappings** (i.e., close-world assumption) for the WHO:disease, WHO:diagnosis, WHO:person and WHO:location according to the available data sources. If you make any assumption clearly state it **[1p]**

WHO:Disease(diseaseId, diseaseName, screeningMethod) = Wrapper1(diseaseId, diseaseName, screeningMethod).
WHO:Disease(diseaseId, diseaseName) = Wrapper2(_, _, diseaseId, diseaseName).
WHO:Disease(diseaseId, diseaseName) = Wrapper3(_, _, _, _, diseaseId, diseaseName).

WHO:Diagnosis(diagnosisId, date, placeOfDiagnosis, diseaseId, diagnosed) =
Wrapper3(diagnosisId, DiagnosisDate, locationOfDiagnosisId, _, diseaseId, _).

WHO:Person(Id, gender) = Wrapper2(personId, personGender, _, _).

WHO:Location(locationId, coordinates) = Wrapper3(_, _, locationOfDiagnosisId, locationCoordinates, _, _).

b) Let us now consider LAV. Write the LAV **exact mappings** for the three wappers presented **[1p]**

Wrapper1(diseaseId, diseaseName, screeningMethod) = WHO:Disease(diseaseId, diseaseName, screeningMethod)
Wrapper2(personId, personGender, diseaseId, diseaseName) = WHO:Person(personId, personGender),
WHO:Disease(diseaseId, diseaseName).
Wrapper3(diagnosisId, DiagnosisDate, locationOfDiagnosisId, locationCoordinates, diseaseId, diseaseName) =
WHO:Diagnosis(diagnosisId, DiagnosisDate, locationOfDiagnosisId, diseaseId, _),
WHO:Location(locationOfDiagnosisId, locationCoordinates),
WHO:Disease(diseaseId, diseaseName).

c) Assume the target schema and the wrappers you defined for the two previous sections. **For each setting**, discuss and justify if the following query over the target schema could be answered. Justify your answer **[1p]**

```
PREFIX WHO: <http://www.who.int/>
SELECT ?y ?r ?s
WHERE {
         ?x WHO:hasDiseaseId ?y ; hasDiseaseName ?n.
         ?z WHO:diagnosed ?x ; WHO:patient ?w .
         ?w WHO:hasId ?s ; WHO:hasGender ?r .
         FILTER regex(?n, “Chagas”)
      }
```

This query retrieves the disease ID (?y), gender (?r), and patient ID (?s) for patients diagnosed with diseases whose name contains "Chagas".
In the GAV setting, we can answer the query because the required attributes (diseaseId, diseaseName, diagnosed, patientId, and gender) are available through the specified mappings.
In the LAV setting, we can also answer the query because the required attributes are included in the views defined by the wrappers.