XML

- Markup language (like HTML)
- XML stands for eXtensible Markup Language
- XML was designed to store and transport data (nothing else)
  - cares about what the data is
- XML was designed to be self-descriptive
- Both human- and machine-readable

Find Giannis Antetokounmpo

- Note the differences in the URL of the page you have visited

- https://dbpedia.org/page/Giannis_Antetokounmpo
- https://dbpedia.org/data/Giannis_Antetokounmpo.n3
- https://dbpedia.org/data/Giannis_Antetokounmpo will download

Exercise 6

- Find all players on GA’s draft team that started playing in 2012 (see how this looks on GA’s turtle data page)

  "2012"^^xsd:gYear

Exercise 6

```
SELECT ?players
FROM <http://dbpedia.org/data/Giannis_Antetokounmpo>
WHERE {
   dbr:Giannis_Antetokounmpo dbo:draftTeam ?team .
   ?players dbo:team ?team .
   ?players dbo:activeYearsStartYear "2012"^^xsd:gYear .
}
```

Exercise 7

- For the previous statement, retrieve all player numbers

- Translation: players & numbers
  for those on GA’s draft team that started in 2012

Exercise 7

```
SELECT ?players ?number
FROM <http://dbpedia.org/data/Giannis_Antetokounmpo>
WHERE {
   dbr:Giannis_Antetokounmpo dbo:draftTeam ?team .
   ?players dbo:team ?team .
   ?players dbo:activeYearsStartYear "2012"^^xsd:gYear .
   ?players dbo:number ?number .
}
```

Exercise 8 - FILTER

- **Add FILTER to your WHERE statement for number 34**

```
SELECT ?players ?number
FROM <http://dbpedia.org/data/Giannis_Antetokounmpo>
WHERE {
   dbr:Giannis_Antetokounmpo dbo:draftTeam ?team .
   ?players dbo:team ?team .
   ?players dbo:activeYearsStartYear "2011"^^xsd:gYear .
   ?players dbo:number ?number .

   FILTER (?number = "34")
}
```

Exercise 9 - COUNT

Instead of

?variable

Use

COUNT (? variable) OR (COUNT (?variable) as ?count)

Starting from G.A.'s draft team (1 triple)

add another triple
and count one variable (don't forget to add to the select clause)

Ontology

- Either:
  - the branch of metaphysics dealing with the nature of being
- Or:
  - a set of concepts and categories in a subject area or domain that shows their properties and the relations between them

More formally – what is an ontology

Ontology is a formal explicit description of:

- **concepts** in a domain of discourse (a.k.a. **classes**),
- **properties** of each concept describing various features and attributes of the concept (**slots**; sometimes called **roles**),
- and **restrictions** on slots (**facets**; sometimes called role restrictions).

take-home message

- An ontology together with a set of individual instances of classes constitutes a knowledge base.

- Ontology + class instances = knowledge base

- **In reality, there is an overlap between where the ontology ends and the knowledge base begins**

Classes, subclasses, …

- Classes describe concepts in the domain.
  Example,
  - a class of wines represents all wines.
  - Specific wines are instances of this class.

- A class can have **subclasses** that represent concepts that are **more specific** than the superclass.
- For example, we can divide the class of all wines into red, white, and rosé wines. Alternatively, we can divide a class of all wines into sparkling and non-sparkling wines. Let’s draw this together

THE AMERICAN COLLEGE OF GREECE
Pierce
Deree
Alba
1875

Slots (properties)

- Slots describe properties of classes **(and instances)**
- Château Lafite Rothschild Pauillac wine:
  - has a full body
  - is produced by the Château Lafite Rothschild winery.
- Specifically: slot “body” has a value of “full” and slot “maker” with the value “Château Lafite Rothschild winery”
- **At the class level, we can say that instances of the class Wine will have slots describing their flavor, body, maker of the wine and so on…**