

# Course Project

Winter 2019

Instructor: Raghava Mutharaju  
IIIT-Delhi



# Logistics

- Group project with team size of 2 or 3
  - If the team size is 1 then you will straightaway get some extra credit
- Each group works on a *different* dataset and preferably different domain as well
- There will be intermediate deadlines
- Expected deliverables
  - Demo (team)
  - Code, OWL file, query file (team)

# Project Tasks

1. Find the data that you want to model and query
2. Build an ontology to model the data
3. Convert the data into RDF triples that conform with the ontology you built
4. Store the triples in a triple store
5. Query (templates will be provided) the triples and show the results
6. Add SHACL constraints (if time permits/extra credit)
7. Develop an application on top of this structured data (extra credit)

# Data

- Choose any CSV file(s) that have 15-20 columns and around 10k rows
  - <https://dataverse.harvard.edu/>
  - <http://statweb.stanford.edu/~sabatti/data.html>
- Use a CSV reader to read the data
- Deadline: March 4, 2019. 11:59 pm
- Submission details
  - Link to the csv file(s)
  - Top 20 rows of the csv file(s)
  - Send us the summary: domain, number of columns, number of rows

# Ontology

- Minimum requirements
  - 10 classes
  - 2 or 3 levels of class hierarchy
  - 3-5 object properties
  - 3-5 data properties
  - Property hierarchy
    - 1-2 subproperty relations
    - 1 property chain (encouraged but optional)
  - Domain and Range for all the properties
  - 3-5 Existential class expressions
  - 3 Disjoint classes
  - Use at least 1-2 ontology design patterns
- Deadline: March 12, 2019. 11:59 pm
  - Submit the owl file (can be changed later)

# RDF Graph

- Convert the data in the CSV file(s) to a set of triples
- This is your instance data
- Use (import) the ontology and use the classes and properties in the triples
  - *cr-data:raita* *rdf:type* *cr-ont:Dish*
  - *cr-data:raita* *cr-ont:hasIngredient* *cr-data:curd*
  - *cr-data:curd* *rdf:type* *cr-ont:Ingredient*
- Ignore empty and NULL values in the CSV
- Be careful with complex cell values (might have to be broken down into multiple separate triples)
- Make use of a CSV library such as Apache Commons CSV



# Minimum Requirements

- Mapping file containing the mapping between the column number/name in the CSV and the class/property in the ontology. This can be in any format
- Number of triples in the RDF Graph should not be less than 500k
- Make use of at least one existing vocabulary such as Dublin Core, FOAF, schema.org, SKOS, SIOC, CC, etc. Modify your ontology accordingly
- Choose any triple store (next slide). There should be an even spread of triple stores among the project groups
- Load the triples into the triple store
- Deadline: [April 10, 2019](#)
- Deliverable: Submit the following screenshots
  - Mapping file
  - Your code/program that includes the triple store connection string and the result of the SPARQL count query



# Triple Stores

1. Jena TDB
2. Virtuoso
3. GraphDB
4. Blazegraph
5. Stardog (trial version available for 60 days)
6. RDFox
7. AllegroGraph
8. RDF4J
9. 4store
10. Mulgara