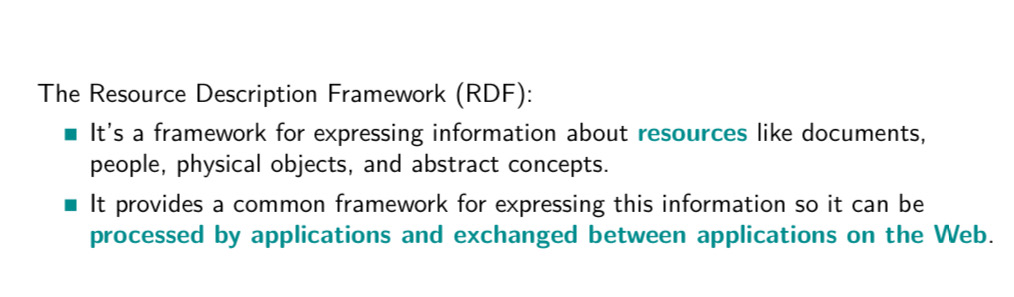
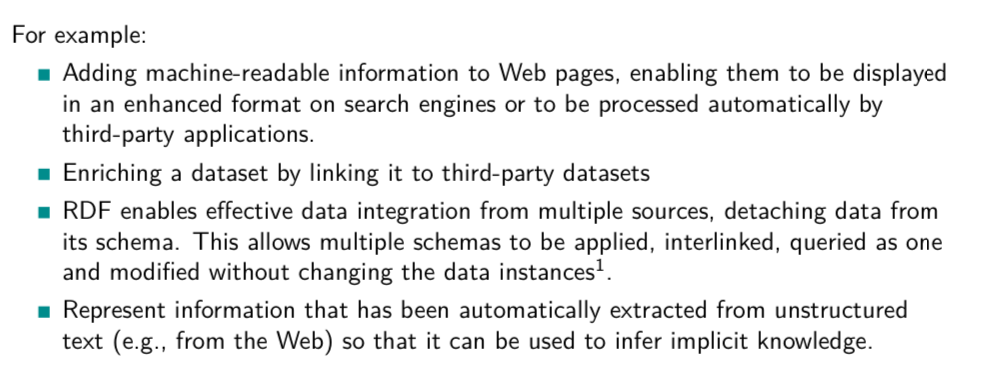
**Exercise session 1: RDF**

**RDF**

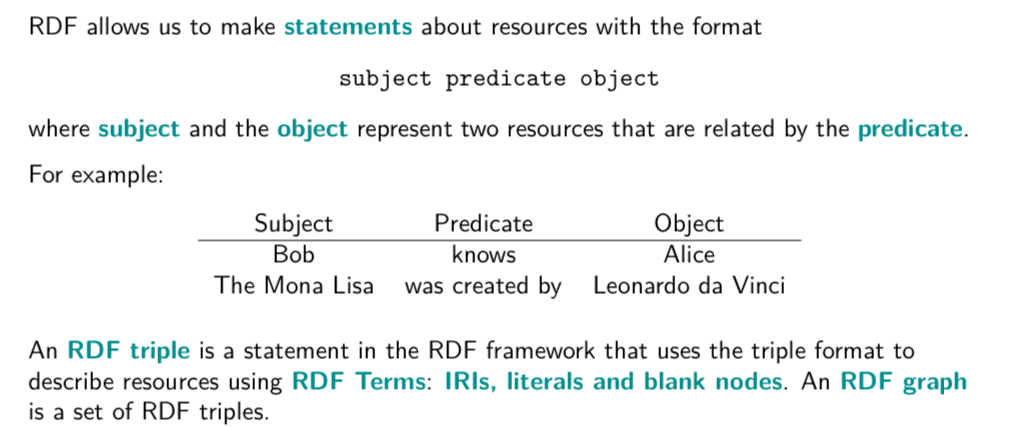


Why use RDF

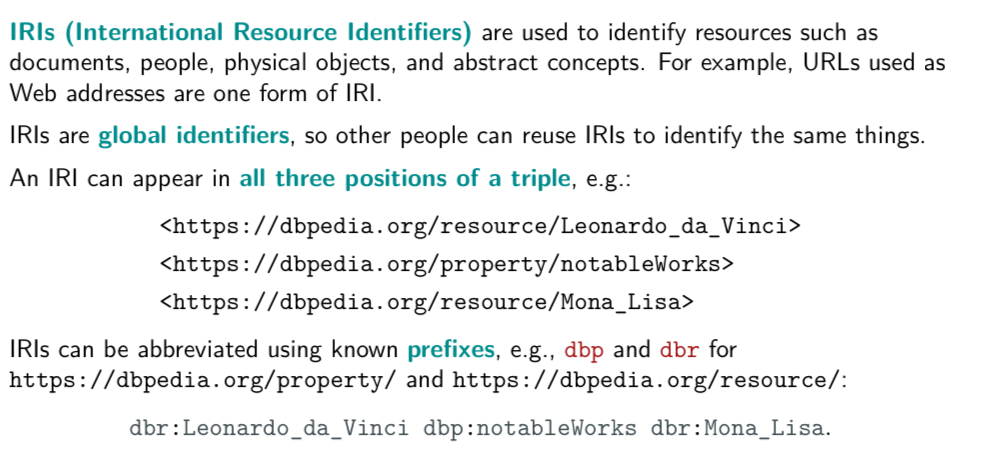


RDFallows make stamens

Subject predicate object

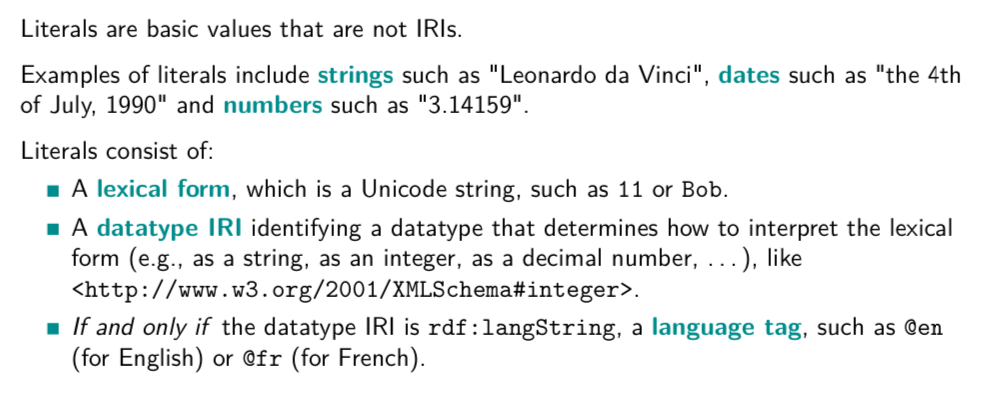


RDF triple are like formulas



URL are one form of IRL

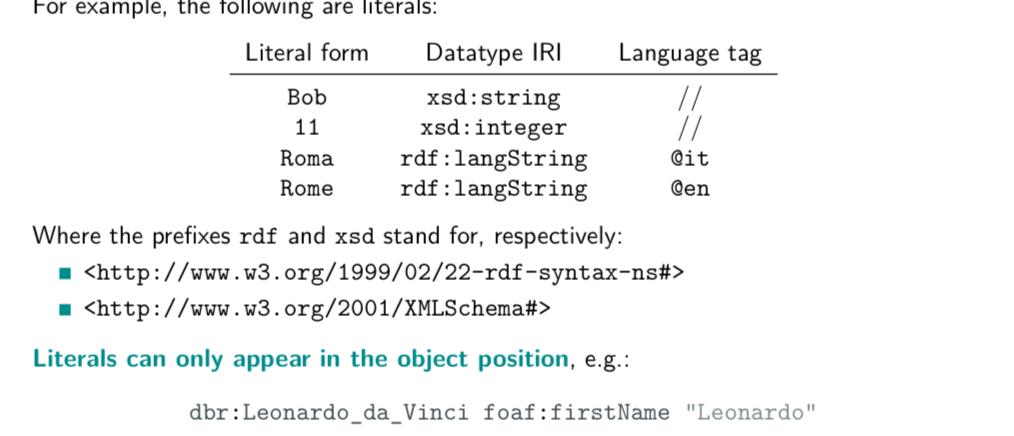
An IRI can appear in all three position of a triple

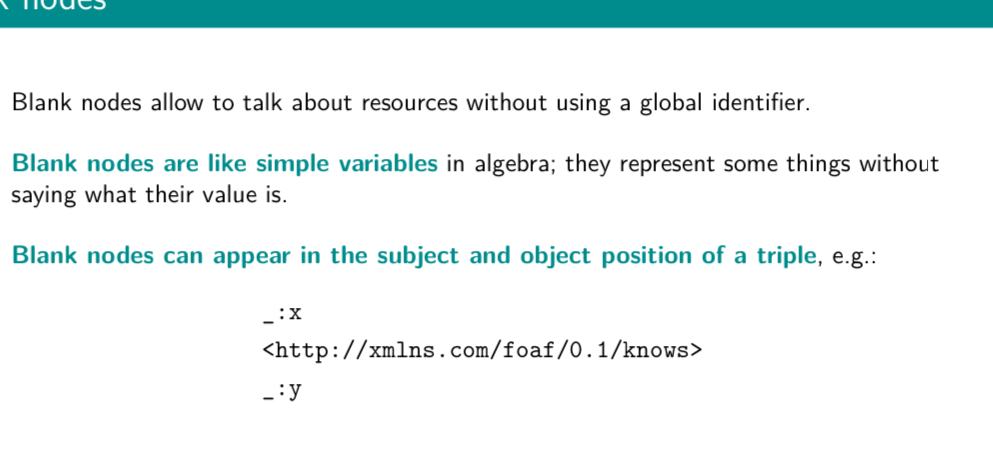


Use lite Lars to represent this basic values

Consist of

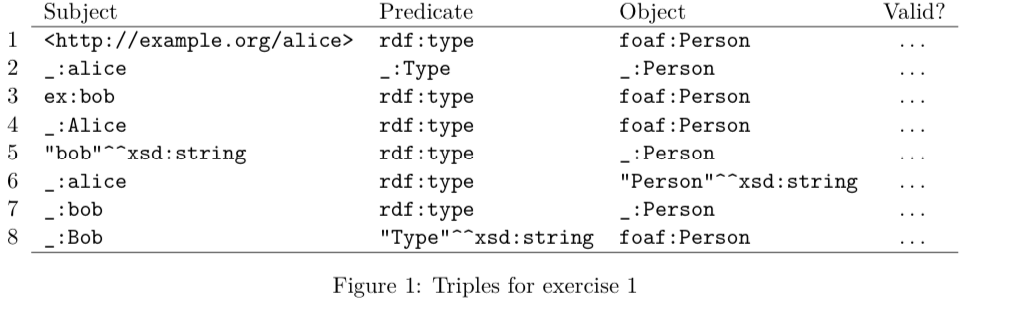
* lexical form: string of characters
* Data type IRI: tell you how to intepret the lexical form
* +compeltare





Represent something without saying what their value is

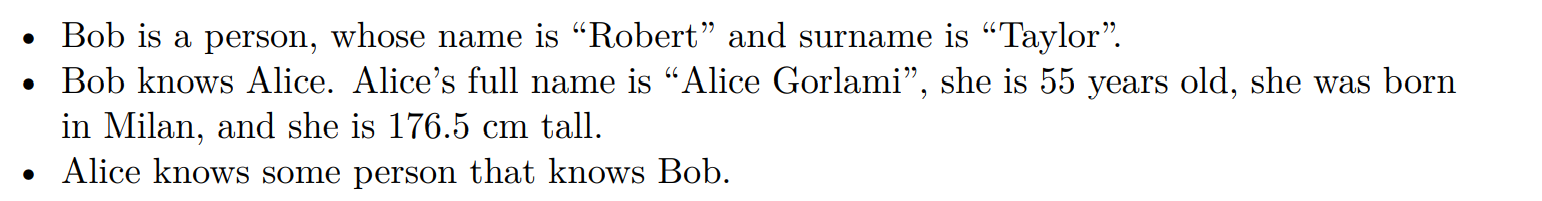
## EXERCISE 1.1: RDF TRIPLES



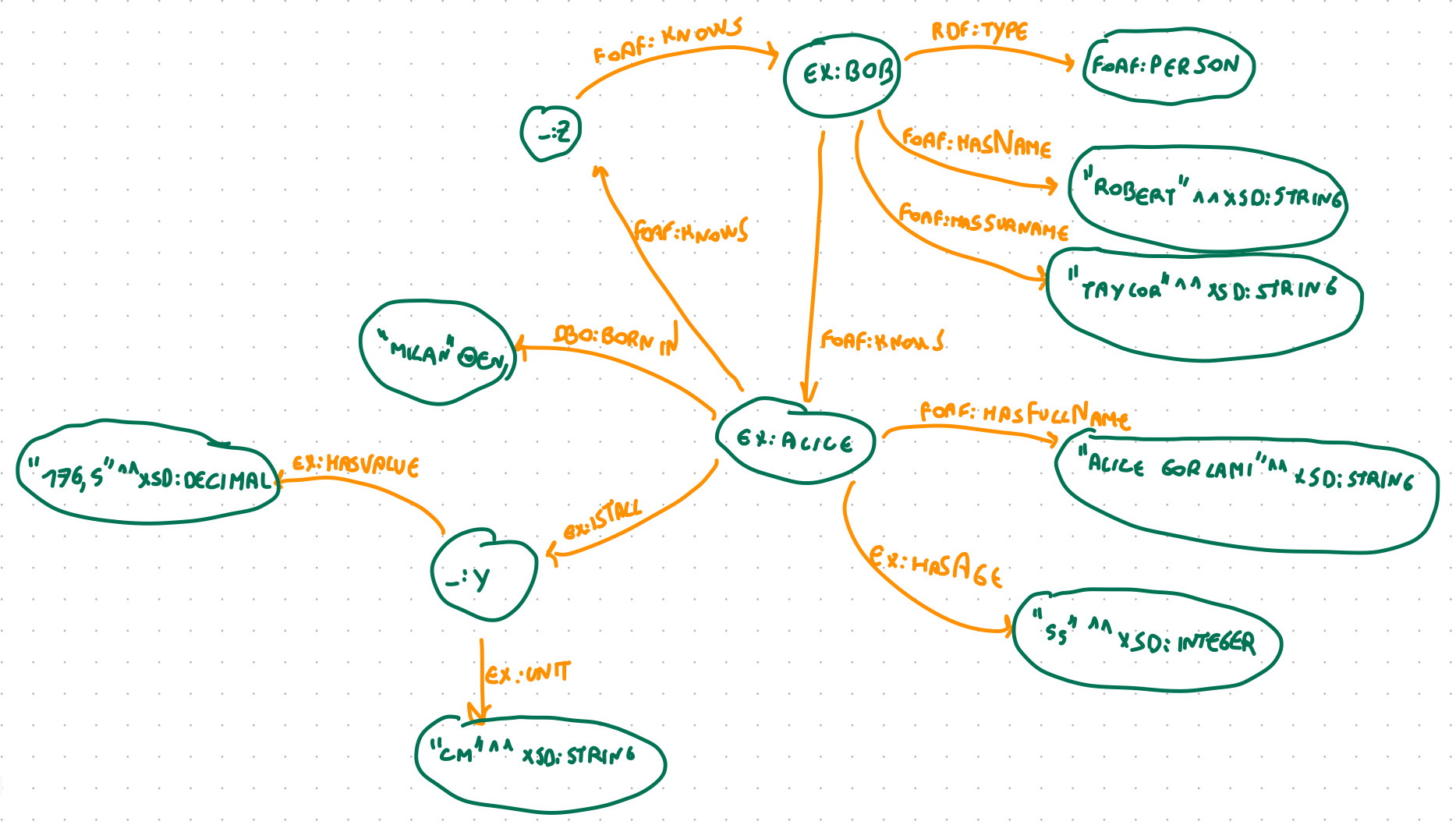
1. True
2. False, blank node as predicate
3. True
4. True
5. False, literal as subject
6. True
7. True
8. False, literal as predicate

FOAF: Friend of a friend

## EXERCISE 1.2: RDF TRIPLES



| **SUBJECT** | **PREDICATE** | **OBJECT** |
| --- | --- | --- |
| ex:bob | rdf:type | foaf:person |
| ex:bob | foat:hasName | “Robert”^^xsd:string |
| ex:bob | foaf:hasSurname | “Taylor”^^xsd:string |
| ex:bob | foaf:knows | ex:Alice |
| ex:alice | foaf:hasFullName | “Alice Gorlami”^^xsd:string |
| ex:alice | foaf:hasAge | “55”^^xsd:integer |
| ex:alice | dbo:bornIn | “Milan”@en |
| ex:alice | ex:isTall | \_:y |
| \_:y | ex:unit | “cm”^^xsd:string |
| \_:y | ex:value | “176,5”^^xsd:decimal |
| ex:alice | foaf:knows | \_:z |
| \_:z | foaf:knows | es:bob |



# EXERCISE 2.1 BLANK NODES AND RDF GRAPHS

