Q1. (20 points) How does Semantic Web relate to XML Schemas? What do ontologies provide us that XML and XML Schema don’t?

Q2. (10 points) What formats (aka serializations) can RDF be represented in?

Q3. (10 points) Which FOAF property/ies is/are used to identify a person/agent?

Q4. (20 points) What are ontologies in the Semantic Web context? Introduce especially their role in and importance for.

Q5. (25 points) Model the following ontology:
   • Herbivores, Carnivores, and Omnivores are animals
   • People are omnivores
   • People have names of type “string”
   • Vegetarians are people who are herbivores
   • “eats” is a property of animals and the values of the property must be of the type “food”
   • Meat and Veggies are types of food

   (a) Using RDFS?

   (b) Using OWL?

Q6. (20 points, (5 points bonus included)) Given the following header of an OWL ontology, state the purpose and meaning/importance/effect of each line:

<table>
<thead>
<tr>
<th>Line #</th>
<th>Line code</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>&lt;rdf:RDF</td>
</tr>
<tr>
<td>2</td>
<td>xmlns    = &quot;<a href="http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#">http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#</a>&quot;</td>
</tr>
<tr>
<td>3</td>
<td>xml:base = &quot;<a href="http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#">http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#</a>&quot;</td>
</tr>
<tr>
<td>4</td>
<td>xmlns:vin = &quot;<a href="http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#">http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#</a>&quot;</td>
</tr>
<tr>
<td>5</td>
<td>xmlns:food = &quot;<a href="http://www.w3.org/TR/2004/REC-owl-guide-20040210/food#">http://www.w3.org/TR/2004/REC-owl-guide-20040210/food#</a>&quot;</td>
</tr>
<tr>
<td>6</td>
<td>xmlns:owl = &quot;<a href="http://www.w3.org/2002/07/owl#">http://www.w3.org/2002/07/owl#</a>&quot;</td>
</tr>
<tr>
<td>7</td>
<td>xmlns:rdf = &quot;<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a>&quot;</td>
</tr>
<tr>
<td>8</td>
<td>xmlns:rdfs = &quot;<a href="http://www.w3.org/2000/01/rdf-schema#">http://www.w3.org/2000/01/rdf-schema#</a>&quot;</td>
</tr>
<tr>
<td>9</td>
<td>xmlns:xsd = &quot;<a href="http://www.w3.org/2001/XMLSchema#">http://www.w3.org/2001/XMLSchema#</a>&quot;</td>
</tr>
</tbody>
</table>
Q1. (20 points) How does semantic Web relate to XML Schemas? What do ontologies provide us that XML and XML Schema don’t?

This issue is also related to the issue of using XML or RDF, addressed in a previous question. First of all, let us quote from the OWL Guide recommendation:

- An ontology differs from an XML Schema in that it is a knowledge representation, not a message format. Most industry based Web standards consist of a combination of message formats and protocol specifications. These formats have been given an operational semantics, such as, “Upon receipt of this PurchaseOrder message, transfer Amount dollars from AccountFrom to AccountTo and ship Product.” But the specification is not designed to support reasoning outside the transaction context. For example, we won’t in general have a mechanism to conclude that because the Product is a type of Chardonnay it must also be a white wine.

- One advantage of OWL ontologies will be the availability of tools that can reason about them. Tools will provide generic support that is not specific to the particular subject domain, which would be the case if one were to build a system to reason about a specific industry-standard XML schema. [...] They will benefit from third party tools based on the formal properties of the OWL language, tools that will deliver an assortment of capabilities that most organizations would be hard pressed to duplicate.

Also, XML data is very sensitive to the XML Schema it refers to. If the XML Schema changes, the same XML data may become invalid, i.e., being rejected by Schema-aware parsers. Somewhat similar dependence on RDF Schemas and Ontologies exist for RDF data, too: if the RDF Schema or OWL Ontology changes, the inferences drawn from the RDF data may change. However, the core RDF data is still usable, there is no notion of the data being “rejected” by, e.g., a parser due to a Schema/Ontology change. In general, RDF is more robust against changing of Schemas and Ontologies than XML is versus Schemas. Note that a GRDDL transformation from XML to RDF may be given by an XML Schema as described in the GRDDL specification. This allows any XML document that validates according to the XML Schema given at the namespace URI of the XML vocabulary to be converted to RDF.

Q2. (10 points) What formats (aka serializations) can RDF be represented in?

RDF statements (or triples) can be encoded in a number of different formats, whether XML based (e.g., RDF/XML) or not (Turtle, N-triples, ...). In general it does not really matter which of these formats (or serializations) are used to express data—the information is represented in RDF triples and the particular format is only the “syntactic sugar”. Most RDF tools can parse several of these serialization formats.

Q3. (10 points) Which FOAF property/ies is/are used to identify a person/agent?

Solution
foaf:mbox and foaf:mbox sha1sum
Mentioning one property is sufficient

Q4. (20 points) What are ontologies in the Semantic Web context? Introduce especially their role in and importance for.
Ontologies define the concepts and relationships used to describe and represent an area of knowledge. Ontologies are used to classify the terms used in a particular application, characterize possible relationships, and define possible constraints on using those relationships. In practice, ontologies can be very complex (with several thousands of terms) or very simple (describing one or two concepts only).

An example for the role of ontologies or rules on the Semantic Web is to help data integration when, for example, ambiguities may exist on the terms used in the different data sets, or when a bit of extra knowledge may lead to the discovery of new relationships.

A general example may help. A bookseller may want to integrate data coming from different publishers. The data can be imported into a common RDF model, e.g., by using converters to the publishers’ databases. However, one database may use the term “author”, whereas the other may use the term “creator”. To make the integration complete, and extra “glue” should be added to the RDF data, describing the fact that the relationship described as “author” is the same as “creator”. This extra piece of information is, in fact, an ontology, albeit an extremely simple one.

Languages like RDF Schemas and various variants of OWL provide languages to express ontologies in the Semantic Web context. These are stable specifications, published in 2004.

Q5. (25 points) Model the following ontology:

- Herbivores, Carnivores, and Omnivores are animals
- People are omnivores
- People have names of type “string”
- Vegetarians are people who are herbivores
- “eats” is a property of animals and the values of the property must be of the type “food”
- Meat and Veggies are types of food

(a) Using RDFS?

Solution in RDFS:

```rdfs
:Herbivore rdfs:subClassOf :Animal .
:Carnivore rdfs:subClassOf :Animal .
:Omnivore rdfs:subClassOf :Animal .
:Person rdfs:subClassOf :Omnivore .
:hasName rdfs:domain :Person .
:hasName rdfs:range xsd:string .
:Vegetarian rdfs:subClassOf :Person .
:Vegetarian rdfs:subClassOf :Herbivore .
:eats rdfs:range :Food .
:Meat rdf:type :Food .
:Veggies rdf:type :Food .
```

(a) Using OWL?

Solution in OWL:

```owl
Class(:Herbivore partial :Animal)
Class(:Carnivore partial :Animal)
Class(:Omnivore partial :Animal)
Class(:Person partial :Omnivore
  restriction(:hasName allValuesFrom(xsd:string)))
DatatypeProperty(:hasName)
```
Class(:Vegetarian partial :Person :Herbivore)
ObjectProperty(:eats domain(:Animal) range(:Food))
Individual(:Meat type(:Food))
Individual(:Veggies type(:Food))

Q6. (20 points) Given the following header of an OWL ontology, state the purpose and meaning/importance/effect of each line:

<table>
<thead>
<tr>
<th>Line #</th>
<th>Line code</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><code>&lt;rdf:RDF</code></td>
</tr>
<tr>
<td>2</td>
<td>xmlns =&quot;<a href="http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#">http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#</a>&quot;</td>
</tr>
<tr>
<td>3</td>
<td>xml:base =&quot;<a href="http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#">http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#</a>&quot;</td>
</tr>
<tr>
<td>4</td>
<td>xmlns:vin =&quot;<a href="http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#">http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#</a>&quot;</td>
</tr>
<tr>
<td>5</td>
<td>xmlns:food =&quot;<a href="http://www.w3.org/TR/2004/REC-owl-guide-20040210/food#">http://www.w3.org/TR/2004/REC-owl-guide-20040210/food#</a>&quot;</td>
</tr>
<tr>
<td>6</td>
<td>xmlns:owl =&quot;<a href="http://www.w3.org/2002/07/ontology#">http://www.w3.org/2002/07/ontology#</a>&quot;</td>
</tr>
<tr>
<td>7</td>
<td>xmlns:rdf =&quot;<a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a>&quot;</td>
</tr>
<tr>
<td>8</td>
<td>xmlns:rdfs =&quot;<a href="http://www.w3.org/2000/01/rdf-schema#">http://www.w3.org/2000/01/rdf-schema#</a>&quot;</td>
</tr>
<tr>
<td>9</td>
<td>xmlns:xsd =&quot;<a href="http://www.w3.org/2001/XMLSchema#">http://www.w3.org/2001/XMLSchema#</a>&quot;</td>
</tr>
</tbody>
</table>

From [http://www.w3.org/TR/owl-guide/](http://www.w3.org/TR/owl-guide/)

The first two declarations identify the namespace associated with this ontology. The first makes it the default namespace, stating that unprefixed qualified names refer to the current ontology. The second identifies the namespace of the current ontology with the prefix `vin:`. The third identifies the base URI for this document (see below). The fourth identifies the namespace of the supporting food ontology with the prefix `food:`.

The fifth namespace declaration says that in this document, elements prefixed with `owl:` should be understood as referring to things drawn from the namespace called [http://www.w3.org/2002/07/ontology#](http://www.w3.org/2002/07/ontology#). This is a conventional OWL declaration, used to introduce the OWL vocabulary.

OWL depends on constructs defined by RDF, RDFS, and XML Schema datatypes. In this document, the `rdf:` prefix refers to things drawn from the namespace called [http://www.w3.org/1999/02/22-rdf-syntax-ns#](http://www.w3.org/1999/02/22-rdf-syntax-ns#). The next two namespace declarations make similar statements about the RDF Schema (`rdfs:`) and XML Schema datatype (`xsd:`) namespaces.

As an aid to writing lengthy URLs it can often be useful to provide a set of entity definitions in a document type declaration (DOCTYPE) that precedes the ontology definitions. The names defined by the namespace declarations only have significance as parts of XML tags. Attribute values are not namespace sensitive. But in OWL we frequently reference ontology identifiers using attribute values. They can be written down in their fully expanded form, for example "http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#merlot". Alternatively, abbreviations can be defined using an ENTITY definition, for example: