Q8. Common ways to make ontologies inconsistent would be:

1) putting an individual in two disjoint classes,

2) declaring two equivalent classes as disjoint

3) violating a maximum or exact cardinality constraint using different individuals.

In the specific case of this ontology we could make it inconsistent by:

1) Creating an individual and 'x' and assigning it both to class 'Office' and 'Building' (`:x rdf:type :Office, :Building ;`) which are disjoint.

3) Having two individuals 'researchOrg1' and 'researchOrg2' of `rdf:type :ResearchOrganization` which are different individuals and having:

```
:aliceBrown       rdf:type :YoungResearcher ;
             :hasContractWith :researchOrg1,
                                 :researchOrg2 ;
```

It causes an inconsistency since an employee can have a contract with exactly one organisation and not more (exact cardinality constraint) and these two organisations are not the same individual:

```
[ rdf:type owl:AllDifferent ;
  owl:distinctMembers ( :researchOrg1
                        :researchOrg2
                      )
] .
```

Q9. The Complex Role Inclusion axiom would be: (hasOffice o isContainedIn o isBuildingOf o isInstituteOf) ⊆ hasContractWith

V_C = {:Employee, :Office, :Building, :Institute, :ResearchOrganization}

V_{OP} = {:hasOffice, :isContainedIn, :isBuildingOf, :isInstituteOf, :hasContractWith}

Turtle Syntax:

```
:hasContractWith owl:propertyChainAxiom ( :hasOffice

                                           :isContainedIn

                                           :isBuildingOf

                                           :isInstituteOf

                                         ) ;
```

Functional syntax:

SubObjectPropertyOf(ObjectPropertyChain(:hasOffice :isContainedIn :isBuildingOf :isInstituteOf) :hasContractWith)

Q10. The created ontology, including the complex role inclusion axiom defined in Q9, would not satisfy the global restrictions on the axioms of an OWL 2 DL ontology because you cannot use a complex property in a cardinality restriction (or in a Functional Property axiom). As a matter of fact for `:hasContractWith` we have that an employee can have exactly one contract with a research organisation so it has both an ObjectExactCardinality restriction and it is a FunctionalObjectProperty; by defining a complex role inclusion axiom with `:hasContractWith` we would not satisfy one of the global restrictions (restriction on simple roles) and also decidability would not be guaranteed.

Without the complex role inclusion axiom defined in Q9 all restrictions would be satisfied. The other restrictions (on `owl:topDataProperty`, on datatypes, on property hierarchy, on anonymous individuals) would be satisfied since:

- the ontology does not include any axiom on `owl:topDataProperty` and `owl:topDataProperty` doesn't have any superproperties,
- the ontology uses datatypes from OWL 2 datatype map and no data ranges are defined,
- there are no cyclic definitions,
- there are no anonymous individuals.

The restriction on simple roles would be satisfied by removing the complex role inclusion axiom defined in Q9 since we would have only simple object properties. Otherwise, if there were no cardinality restrictions on `:hasContractWith` and it wasn't functional the restriction on simple roles would be satisfied as well.

Q11.

1. Find all the offices that host at least one graduate fellow and order the results by employee ID

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX :
<http://www.semanticweb.org/marta/ontologies/2022/0/MeleroCavalloMartina#>

SELECT ?officeiri ?employeeID

WHERE {

  ?officeiri rdf:type :Office .

  ?officeiri :hostsEmployee ?employeeiri .

  ?employeeiri :hasEmployeeID ?employeeID .
  ?employeeiri rdf:type :GraduateFellow .

} ORDER BY ?employeeID
```

2. Find all the senior researchers with ID lower than 5000 who are members of the laboratory named “AIMH”.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX :
<http://www.semanticweb.org/marta/ontologies/2022/0/MeleroCavalloMartina#>

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT ?employeeiri ?employeeID

WHERE {

  ?employeeiri rdf:type :SeniorResearcher .

  ?employeeiri :hasEmployeeID ?employeeID .

  ?employeeiri :isMemberOfLaboratory ?labiri .

  ?labiri rdf:type :Laboratory .

  ?labiri :hasLaboratoryName "AIMH" .

  FILTER (?employeeID < "5000"^^xsd:integer)

}
```

3. Find all the laboratories that have a total number of research group greater than 2.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX :
<http://www.semanticweb.org/marta/ontologies/2022/0/MeleroCavalloMartina#>

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT ?labiri (COUNT(?groupiri) AS ?num)

WHERE {

  ?labiri rdf:type :Laboratory .

  ?labiri :hasResearchGroup ?groupiri .

  ?groupiri rdf:type :ResearchGroup .

} GROUP BY ?labiri HAVING (COUNT(?groupiri) > "2"^^xsd:integer)
```