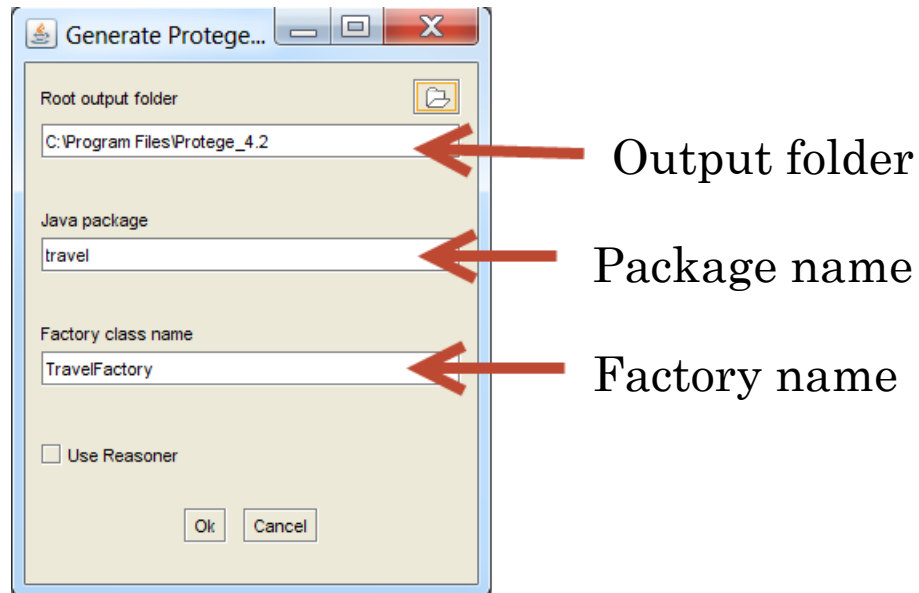
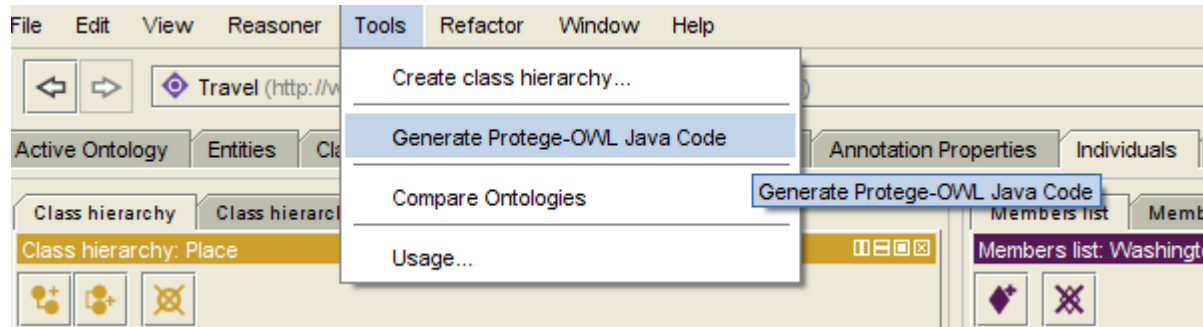


Generate Java code in Protege 4.2

Generate code



How to use it

- Basically helper classes
- We can use an instance of the OWLOntology class and an IRI to fetch classes and individuals
- In order for it to be intuitive, we may want to put on some object or data properties on our individuals, i.e. FirstName, LastName or UserId

Example

Load the ontology

```
//Ontology parameters  
OWLOntologyManager manager = OWLManager.createOWLOntologyManager();  
URI uri = URI.create("http://www.semanticweb.org/ontologies/2013/2/Travel");  
IRI iri = IRI.create(uri);  
File f = new File("Travel.owl");  
  
//Load ontology  
OWLOntology ontology = manager.loadOntologyFromOntologyDocument(f);
```

Bind individual to helper class

```
//Get all male objects
Set<OWLEntity> entity = ontology.getEntitiesInSignature(
    IRI.create(URI.create(("http://www.semanticweb.org/ontologies/2013/2/Travel#Male"))));

DefaultMale individualMartin = null;
//Iterate male objects
for(OWLEntity e: entity){
    for(OWLIndividual i: e.asOWLClass().getIndividuals(ontology)){
        //System.out.println(i.toStringID());
        IRI userIRI = IRI.create(i.toStringID());
        DefaultMale anonymusMale = new DefaultMale(ontology, userIRI);

        //Locate correct user
        if(anonymusMale.getHasUserId().contains(martinClient.getUserId())){
            individualMartin = anonymusMale;
            break;
        }
    }
}
}
```

Use the helper class

```
if(individualMartin != null){
    System.out.println("Martin was found!");

    //Print individual info
    System.out.println(individualMartin.toString());
    System.out.println(individualMartin.getHasFirstName().toArray()[0] + " " +
individualMartin.getHasLastName().toArray()[0] + ", with UserID: " +
individualMartin.getHasUserId().toArray()[0]);

    //Print individual reference
    if(individualMartin.hasHasResidence()){
        Collection<Place> residence = (Collection<Place>) individualMartin.getHasResidence();
        System.out.print("Individual lives in: ");
        for(Place p1: residence){
            System.out.println(p1.getHasPlaceName().toArray()[0]);
        }
    }
}
```

```
Male(  
  hasDrinksTypePreference: Americano;  
  hasFoodTypePreference: Cheeseburger, Maki, ThaiWok;  
  hasResidence: Oslo;  
  hasTrip: TripMartinToWashington;  
  hasFirstName: Martin;  
  hasGPSLocation: 14;  
  hasLastName: Folkeseth;  
  hasUserId: 101;  
)
```

Martin Folkeseth, with UserID: 101

Individual lives in: Oslo