

OCL Exercise: University

Master Solution

Task 1. The Analysis Object Model is shown in Figure 1. It maps the given description in a rather straightforward way. Note that the associations between Student and University and between Lecturer and University have been merged into one association between Person and University. It is also possible to have only ‘one-way’ associations. This leads to less invariants.

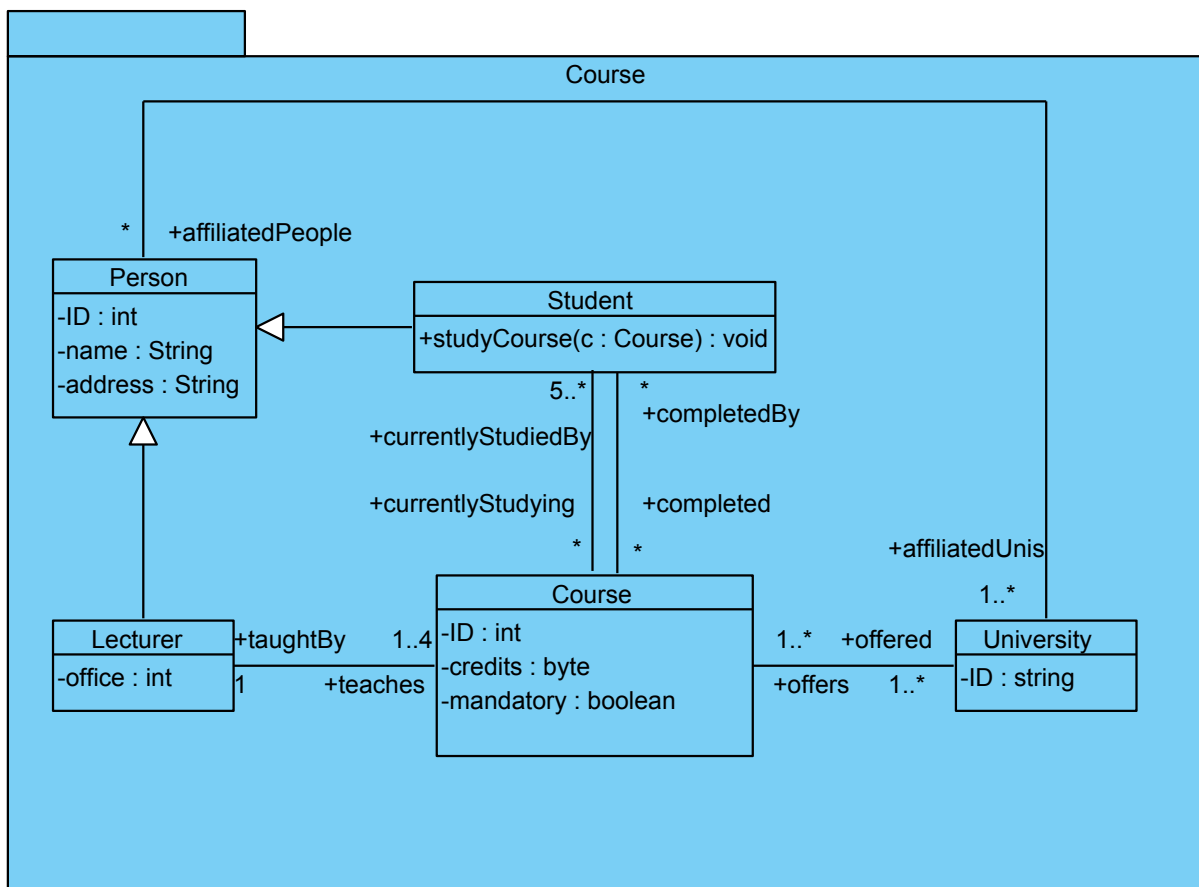


Figure 1: Class Diagram

Task 2. First, we give specifications for the uniqueness properties. We have to express that (1) all universities have unique IDs; (2) within one university each course has a unique

ID; and (3) within one university each student and lecturer has a unique ID. We use the “<>” symbol for inequality.

context University **inv**:

University.allInstances()—>**forAll**(u1, u2 | u1 <> u2 **implies** u1.ID <> u2.ID)

context University **inv**:

offers—>**forAll**(c1, c2 | c1 <> c2 **implies** c1.ID <> c2.ID)

context University **inv**:

affiliatedPeople—>**forAll**(p1, p2 | p1 <> p2 **implies** p1.ID <> p2.ID)

Next, we express constraints that were given in the description but cannot be expressed on the diagram: (1) mandatory courses are worth more than 5 credit points; and (2) students only take courses that are offered by the university they are enrolled to.

context Course **inv**:

self.mandatory **implies** **self**.credits > 5;

context Student **inv**:

currentlyStudying—>**forAll**(c | affiliatedUnis—>**exists**(u | u.offers—>**includes**(c)))

Task 3. As a precondition, we have to express that course *c* is offered by one of the universities the student is affiliated with. This precondition is necessary to maintain the fifth invariant above.

context Student::studyCourse(c: Course)

pre: **self**.affiliatedUnis—>**exists**(u | u.offers—>**includes**(c))

post: **self**.currentlyStudying—>**includes**(c)

Note that in the postcondition we do not have to specify that course *c* is offered by one of the universities the student is affiliated with as it is already expressed by an invariant.