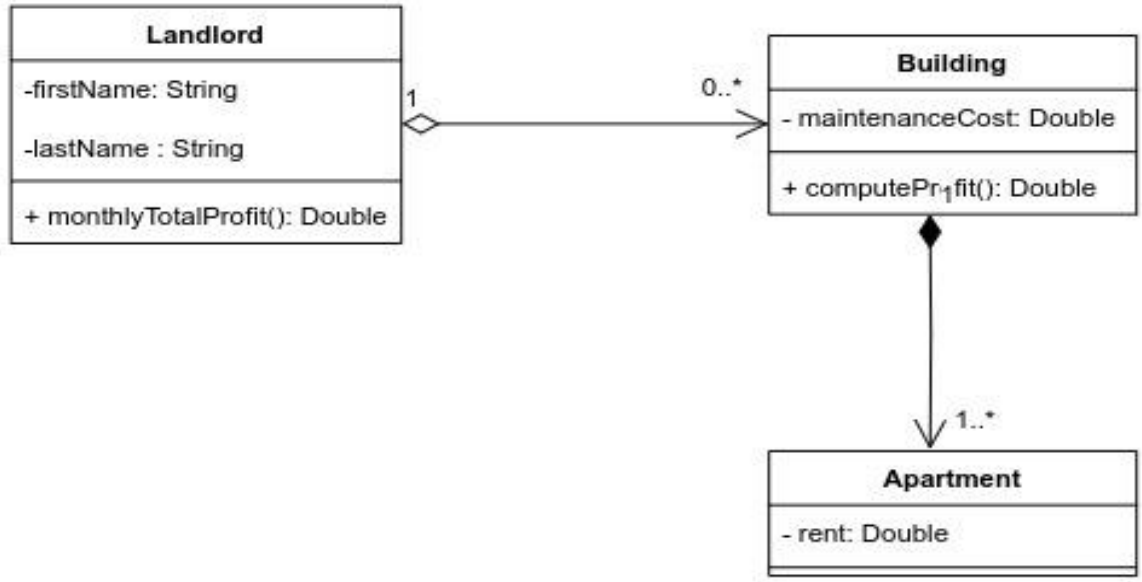
Lab3

1. **p1.equals(p2)** is **false** because in the equals method of PersonWithJob we test if p2 isinstanceof PersonWithJob which is not true. PersonWithJob is a Person but the opposite is not true.

   p2.equals(p1) is **true** because p1 which is an instance of PersonWithJob is a Person as well.

2. **Problem 2 design**



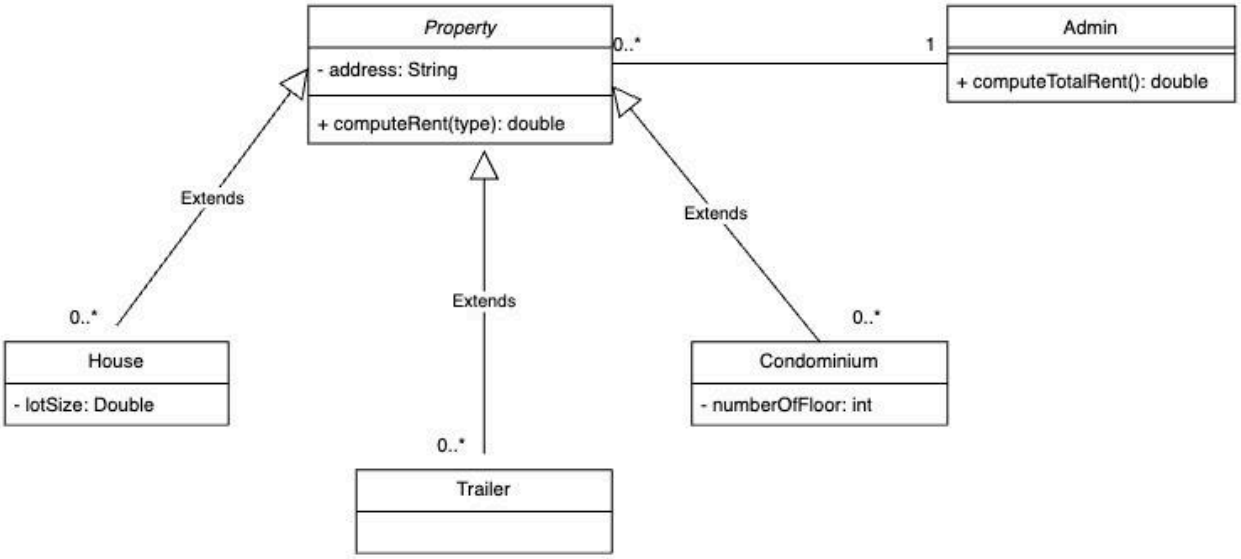
3. **Problem 3**

   A. Yes it makes sense because geometrically, a cylinder has a circular base, so it's tempting to model `Cylinder` as a subclass of `Circle`. But we have some problems here:
   - Cylinder IS-A Circle? Not really. A cylinder is not a circle, it's a 3D object with a circle as a base
   - Inheritance can lead to misleading behavior for example Cylinder.computeArea() would be confusing
   - This way is less flexible for OOP design

   B. Design with composition


Cylinder
- height: Double
+ computeVolume(): Double
1
1
Circle
- radius: Double
+ computeArea(): Double

Problem 4


Property
- address: String
+ computeRent(type): double
0..*
1
Admin
+ computeTotalRent(): double
Extends
Extends
Extends
0..*
0..*
0..*
House
- lotSize: Double
Condominium
- numberOfFloor: int
Trailer