```
----------------Employee part----------------

sig Employee {
     employeeID: Int,
} {
     employeeID > 0
}

fact uniqueEmployee{
     all e1,e2: Employee| (e1 != e2) implies (e1.employeeID
!= e2. employeeID)
}

----------------PairOfWorkers part----------------

sig PairOfWorkers {
     employees: some Employee
} {
     #employees = 2
}

fact uniquePairOfWorkersForEmployee {
     all e :Employee| lone p : PairOfWorkers| ((e in
p.employees))
}

fact mapPairOfWorkersOnLists{
     #PairOfWorkers = #List
}

fact differentPairPerList{
     all l1,l2: List| (l1!=l2) implies (l1.pair!=l2.pair)
}
```