





```

1  -- $ProjectHeader: use 0.393 Wed, 16 May 2007 14:10:28 +0200 opti $
2
3  -- Example illustrating pre- and postconditions
4
5  model healthcare
6  enum Patient_Status{Expectant, Immediate, Delyed, Minor}
7  enum Triage_Speciality{Doctor, Nurse, Technician}
8  enum Tech_Speciality{Diagnostic_Radiographer, Healthcare_Assistant, Paramedic}
9  enum Doc_Speciality{Emergency, Surgery, Anaesthesiology, Orthopaedics, Neurosurgery, Ophthalmology, Cardiology, Nephrology,
10 Radiology }
11 ----- enum Doc_Special{EmergencyDoc, SurgicalDoc, CardiologyDoc } ---
12 enum Gender{Male, Female}
13 -----
14 class Hospital
15 attributes
16   name:String
17   phone:Integer
18   address:String
19 end
20
21 class Departments
22 attributes
23   name:String
24   id:String
25   loc:String
26   phone:String
27 end
28
29 class People
30 attributes
31   fname:String
32   lname:String
33   gender:Gender
34   birthday:String
35 end
36
37 class AE_Department < Departments
38 attributes
39   size : Integer
40   capacity: Integer
41 end
42
43 class Patient < People
44 attributes
45   patientNo:String
46   dateadmit:String
47   timeadmit:String
48   accident:Boolean
49   status: Patient_Status
50   assigned:Boolean
51 end
52
53 class Staff < People
54 attributes
55   id:Integer
56   triagestaff:Boolean
57   address:String
58 end
59
60 class Triage_Staff < Staff
61 attributes
62   triageno:Integer
63   kind:Triage_Speciality
64 end
65
66 class Receptionist < Staff
67 end
68
69 class Porter < Staff
70 end
71
72 class Doctor < Triage_Staff
73 attributes
74   pager:Integer
75   specialization:Doc_Speciality
76   notes:String
77 end
78
79 class Nurse < Triage_Staff
80 attributes

```

```

79     nurseSpecialty:String
80     pager:Integer
81 end
82
83 class Technicians < Triage_Staff
84 attributes
85     techspecialty:Tech_Speciality
86     pager:Integer
87 end
88
89 -----composition-----
90 composition Consisting between
91     Hospital[1] role hospital
92     Departments[0..*] role departments
93 end
94
95 ----- patient Associations -----
96
97 association Admit between
98     AE_Department[1] role department
99     Patient[0..*] role patient
100 end
101 association Checks between
102     Triage_Staff[*] role staff
103     Patient[*] role patient
104 end
105 association WorkFor between
106     AE_Department[1] role department
107     Staff[1..*] role staff
108 end
109 -----aggregation-----
110
111 constraints
112
113 context AE_Department
114 inv: self.size>0 and self.capacity>0
115 inv: AE_Department.allInstances()->forAll(s|s.staff->notEmpty())
116 inv: self.staff->exists(s|oclIsTypeOf(Doctor))
117
118 context Staff
119 inv: Staff.allInstances()->forAll(s1,s2|s1<>s2 implies s1.id <> s2.id)
120 inv: Staff.allInstances()->exists(s|s.triagestaff=true)
121
122 context Triage_Staff
123 inv: Triage_Staff.allInstances()->exists(s|s.kind=#Doctor)
124 inv: Triage_Staff.allInstances()->exists(s|s.kind=#Nurse)
125 inv: Triage_Staff.allInstances()->exists(s|s.kind=#Technician)
126
127 context Patient
128 inv: Patient.allInstances()->forAll(p|p.status=#Expectant or p.status=#Immediate implies p.assigned=true)
129 inv: Patient.allInstances()->forAll(p|p.staff->notEmpty())
130
131

```