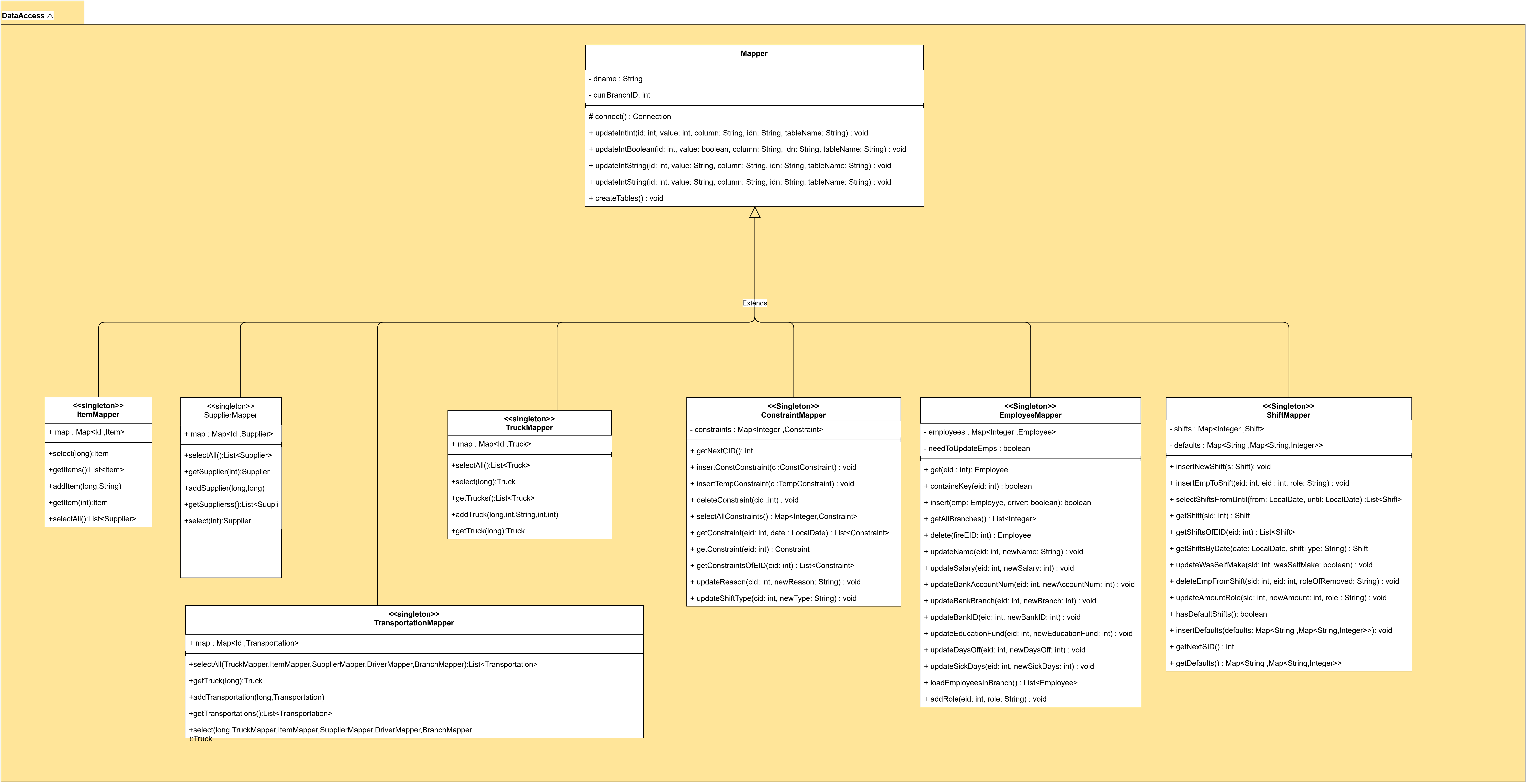
DataAccess △
Mapper
- dname : String
- currBranchID: int
# connect() : Connection
+ updateIntInt(id: int, value: int, column: String, idn: String, tableName: String) : void
+ updateIntBoolean(id: int, value: boolean, column: String, idn: String, tableName: String) : void
+ updateIntString(id: int, value: String, column: String, idn: String, tableName: String) : void
+ updateIntString(id: int, value: String, column: String, idn: String, tableName: String) : void
+ createTables() : void
Extends
<<singleton>>
ItemMapper
+ map : Map<Id ,Item>
+select(long):Item
+getItems():List<Item>
+addItem(long,String)
+getItem(int):Item
+selectAll():List<Supplier>
<<singleton>>
SupplierMapper
+ map : Map<Id ,Supplier>
+selectAll():List<Supplier>
+getSupplier(int):Supplier
+addSupplier(long,long)
+getSupplierss():List<Suupli
+select(int):Supplier
<<singleton>>
TruckMapper
+ map : Map<Id ,Truck>
+selectAll():List<Truck>
+select(long):Truck
+getTrucks():List<Truck>
+addTruck(long,int,String,int,int)
+getTruck(long):Truck
<<Singleton>>
ConstraintMapper
- constraints : Map<Integer ,Constraint>
+ getNextCID(): int
+ insertConstConstraint(c :ConstConstraint) : void
+ insertTempConstraint(c :TempConstraint) : void
+ deleteConstraint(cid :int) : void
+ selectAllConstraints() : Map<Integer,Constraint>
+ getConstraint(eid: int, date : LocalDate) : List<Constraint>
+ getConstraint(eid: int) : Constraint
+ getConstraintsOfEID(eid: int) : List<Constraint>
+ updateReason(cid: int, newReason: String) : void
+ updateShiftType(cid: int, newType: String) : void
<<Singleton>>
EmployeeMapper
- employees : Map<Integer ,Employee>
- needToUpdateEmps : boolean
+ get(eid : int): Employee
+ containsKey(eid: int) : boolean
+ insert(emp: Employye, driver: boolean): boolean
+ getAllBranches() : List<Integer>
+ delete(fireEID: int) : Employee
+ updateName(eid: int, newName: String) : void
+ updateSalary(eid: int, newSalary: int) : void
+ updateBankAccountNum(eid: int, newAccountNum: int) : void
+ updateBankBranch(eid: int, newBranch: int) : void
+ updateBankID(eid: int, newBankID: int) : void
+ updateEducationFund(eid: int, newEducationFund: int) : void
+ updateDaysOff(eid: int, newDaysOff: int) : void
+ updateSickDays(eid: int, newSickDays: int) : void
+ loadEmployeesInBranch() : List<Employee>
+ addRole(eid: int, role: String) : void
<<Singleton>>
ShiftMapper
- shifts : Map<Integer ,Shift>
- defaults : Map<String ,Map<String,Integer>>
+ insertNewShift(s: Shift): void
+ insertEmpToShift(sid: int. eid : int, role: String) : void
+ selectShiftsFromUntil(from: LocalDate, until: LocalDate) :List<Shift>
+ getShift(sid: int) : Shift
+ getShiftsOfEID(eid: int) : List<Shift>
+ getShiftsByDate(date: LocalDate, shiftType: String) : Shift
+ updateWasSelfMake(sid: int, wasSelfMake: boolean) : void
+ deleteEmpFromShift(sid: int, eid: int, roleOfRemoved: String) : void
+ updateAmountRole(sid: int, newAmount: int, role : String) : void
+ hasDefaultShifts(): boolean
+ insertDefaults(defaults: Map<String ,Map<String,Integer>>): void
+ getNextSID() : int
+ getDefaults() : Map<String ,Map<String,Integer>>
<<singleton>>
TransportationMapper
+ map : Map<Id ,Transportation>
+selectAll(TruckMapper,ItemMapper,SupplierMapper,DriverMapper,BranchMapper):List<Transportation>
+getTruck(long):Truck
+addTransportation(long,Transportation)
+getTransportations():List<Transportation>
+select(long,TruckMapper,ItemMapper,SupplierMapper,DriverMapper,BranchMapper
):Truck