

Class: college_student
+ field: Type info: double //用一串数据统一存储学生所有信息 holding book:int// 持有数据数量 violations: int //是否违规
+ method(): Type college_student() Check_violation():int //查询学生是否又违规借阅记录 book_borrow():int //返回数值到holding book book_return():int // 同样 info_check():double//信息检测

Class: college_book
+ field: Type info: double //用一串数据统一存储书本所有信息 state:int //表示状态 可分为预定,在架上,借出 location:double //图书馆内的位置 book:int //图书自身状态
+ method(): Type college_book() Check_state():int //查询图书状态 relocate():double //图书位置更新 info_check():double//信息检测 check_book(): int //图书是否受损等

Class: college_administer
+ field: Type book_info1: double //用一串数据存储书本所有信息 book_info2: int //图书自身状态 student_info:double // 前述的学生信息 book_state:int //表示状态 可分为预定,在架上,借出 book_location:double //图书图书馆内的位置 student_violation:int //学生违纪情况
+ method(): Type college_administer() /*权限主要为更改 设计中reload既能查询又能更改,下同*/ reload_book_info1():double reload_book_info2():double reload_book_state():int reload_book_location():double reload_student_info():double reload_student_vialation():int

Class: automobile_manu
+ field: Type model:int //细致的分类 比如a级轿车,全尺寸suv等 /*这里的结构体仅为形式化的引用,详细信息不列出*/ batch number: struct_batch // 生产批号 包含日期 流水线位置等信息 data released: int // 交付日期 manufacturing state: struct_manu //制造状态 包含剩余时间 repo_num(): int //库存数目
+ method(): Type automobile_manu() classify_model(): int //返回分类信息 check_batch(): struct_batch 生产信息的确认 manufacture():struct_manu //返回制造状态 至state exp_time(): struct_manu //返回至state剩余时间部分 data_released_check():int //返回交付日期 repo_check(): int //返回至库存项目 manu_exception(): int //返回至state的意外处理

Class: automobile_owner
+ field: Type model:int //细致的分类 比如a级轿车,全尺寸suv等 /*这里的结构体仅为形式化的引用,详细信息不列出*/ car_info:struct_info //车内铭牌上的所有信息(包括牌照!) insurance_info: struct_info //车的所有保险信息 car_state: struct_car_info //车的多角度信息 包括基本使用情况和是否能上路等都项信息的复杂结构体
+ method(): Type automobile_owner() classify_model(): int //返回分类信息 check_car_info():struct_info //车内铭牌上的所有信息,只能检查不能更改 reload_insurance_info():struct_info reload_car_state(struct_info):struct_car_info //根据前述已更改信息确认车的信息

Class: automobile_TMA :Traffic management agency
+ field: Type /*简单的用数组表示数据库 实际肯定没那么简单*/ car_info:struct_info //车内铭牌上的所有信息 car_info_database:car_info[]// 存储所有车辆静态信息的数据库 car_data:struct_data //每个车单独的动态信息(违章 年检等) car_data_database:car_data[]// 所有车辆动态信息的数据库
+ method(): Type automobile_TMA() /*对数据库的操作写为*/ check_car_info():struct_info //车内铭牌上的所有信息单独录入 info_database_reload(car_info):car_info_database // 静态数据库录入,有权更改 check_car_data():struct_data //车的最近动态记录单独录入 data_database_reload(car_data):car_data_database // 动态数据库录入,有权更改 manage_data_database():car_data_database //动态数据库的操作