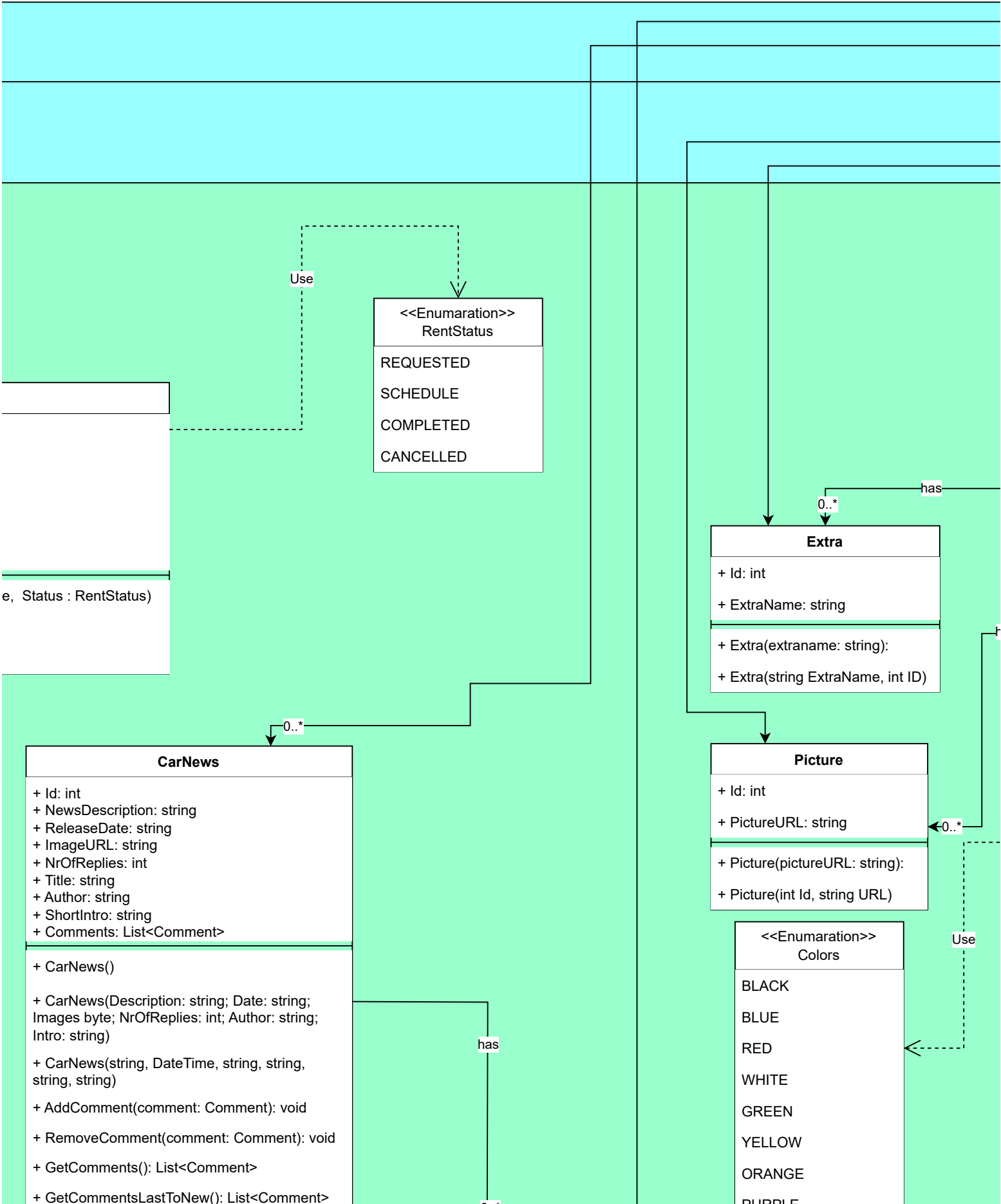


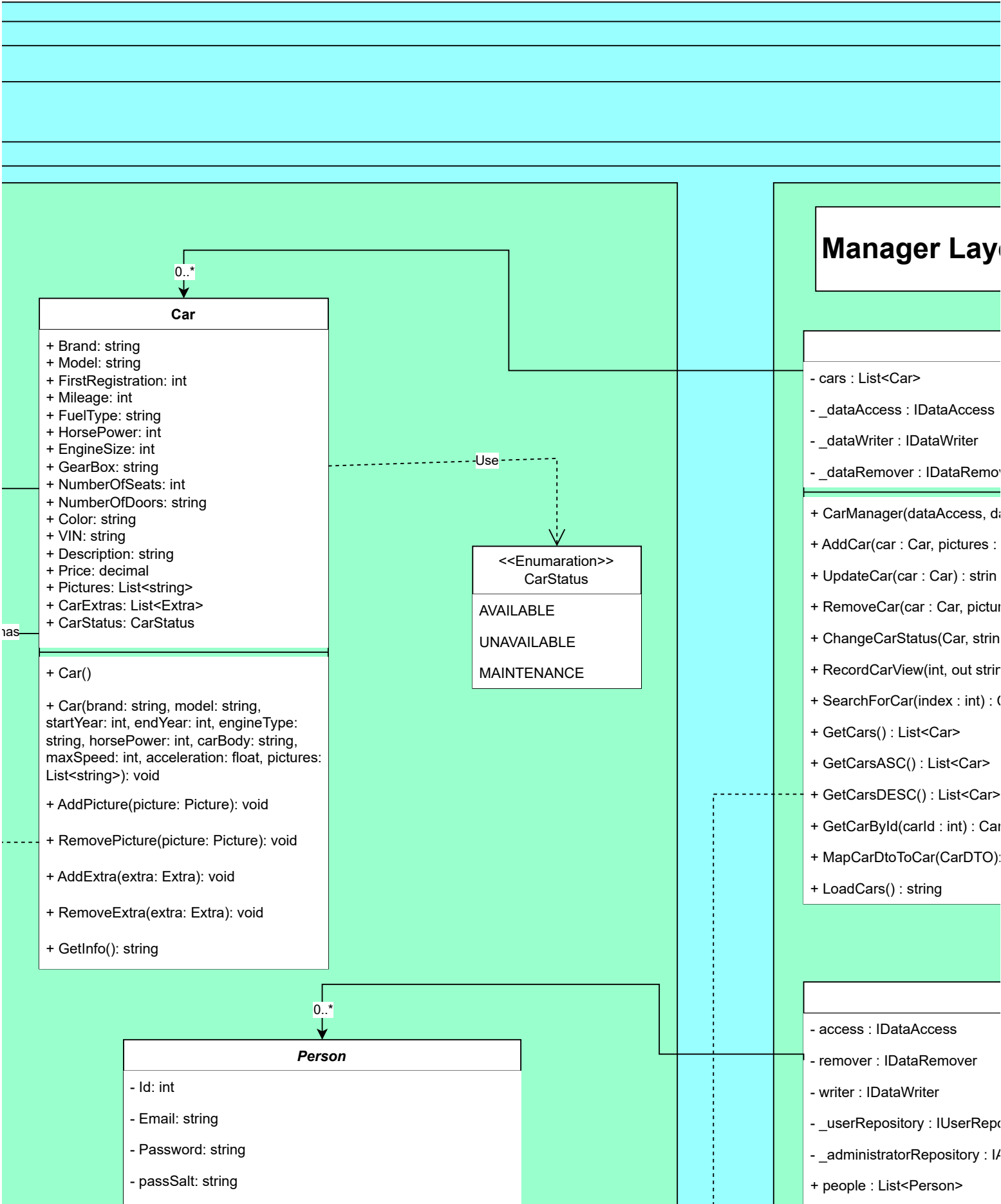
Logic Layer

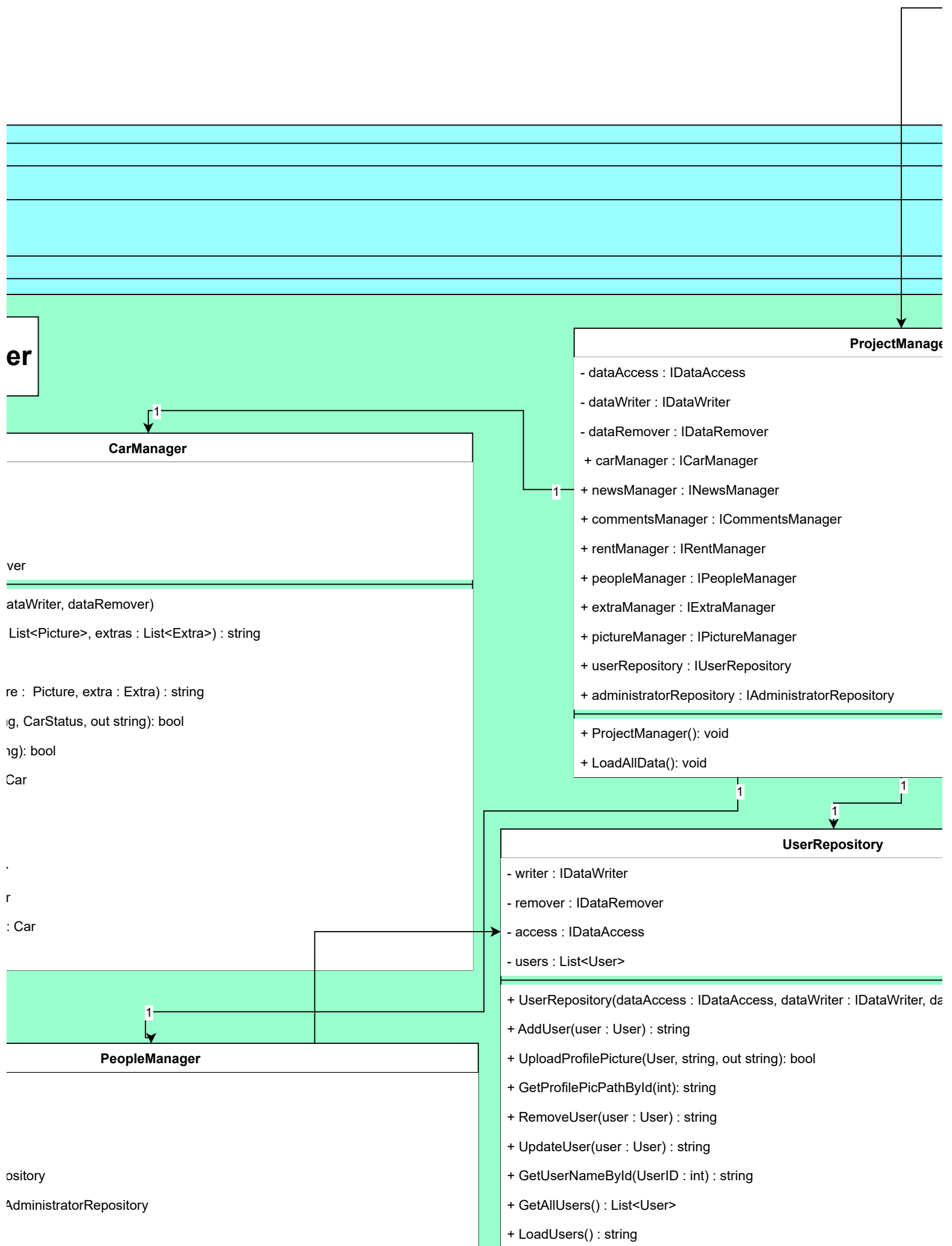
Entity Layer

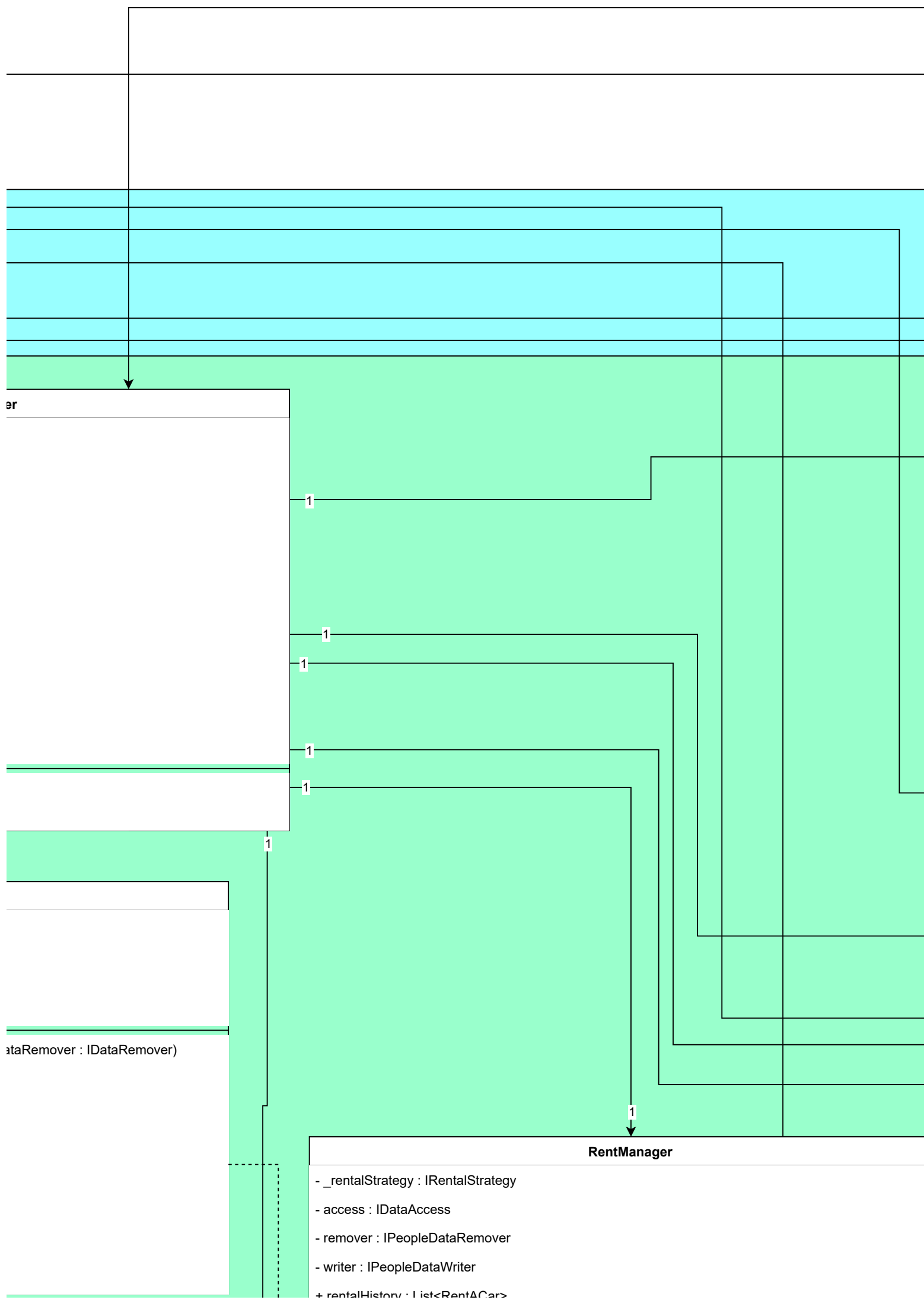
RentACar

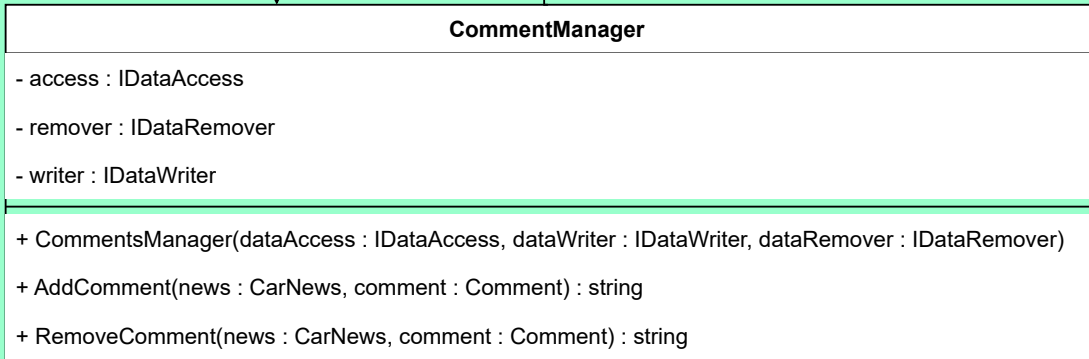
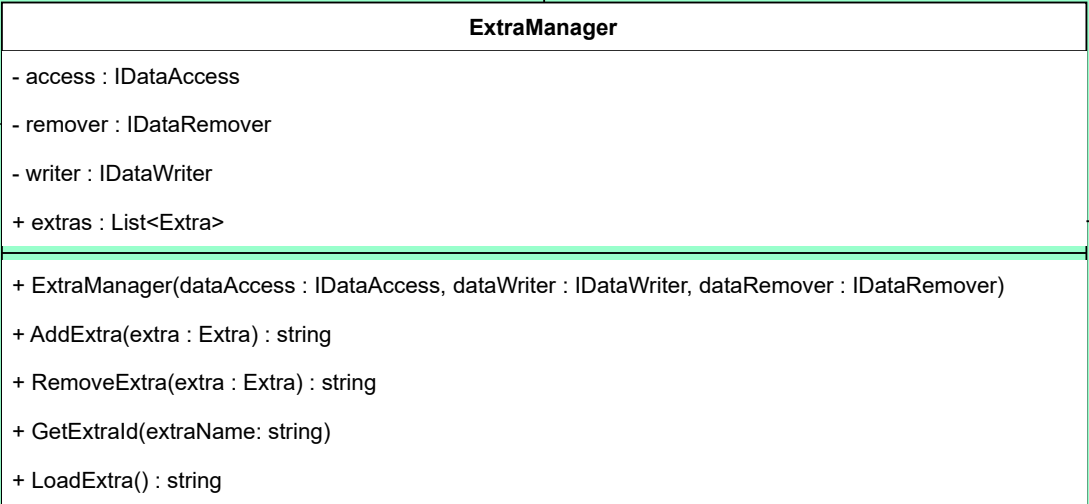
- + User User
 - + Car Car
 - + StartDate: DateTime
 - + ReturnDate: DateTime
 - + Status: RentStatus
-
- + RentACar(user : User, car : Car, startDate : DateTime, returnDate : DateTime)
 - + RentACar()
 - + ChangeStatus(Status : RentStatus): void



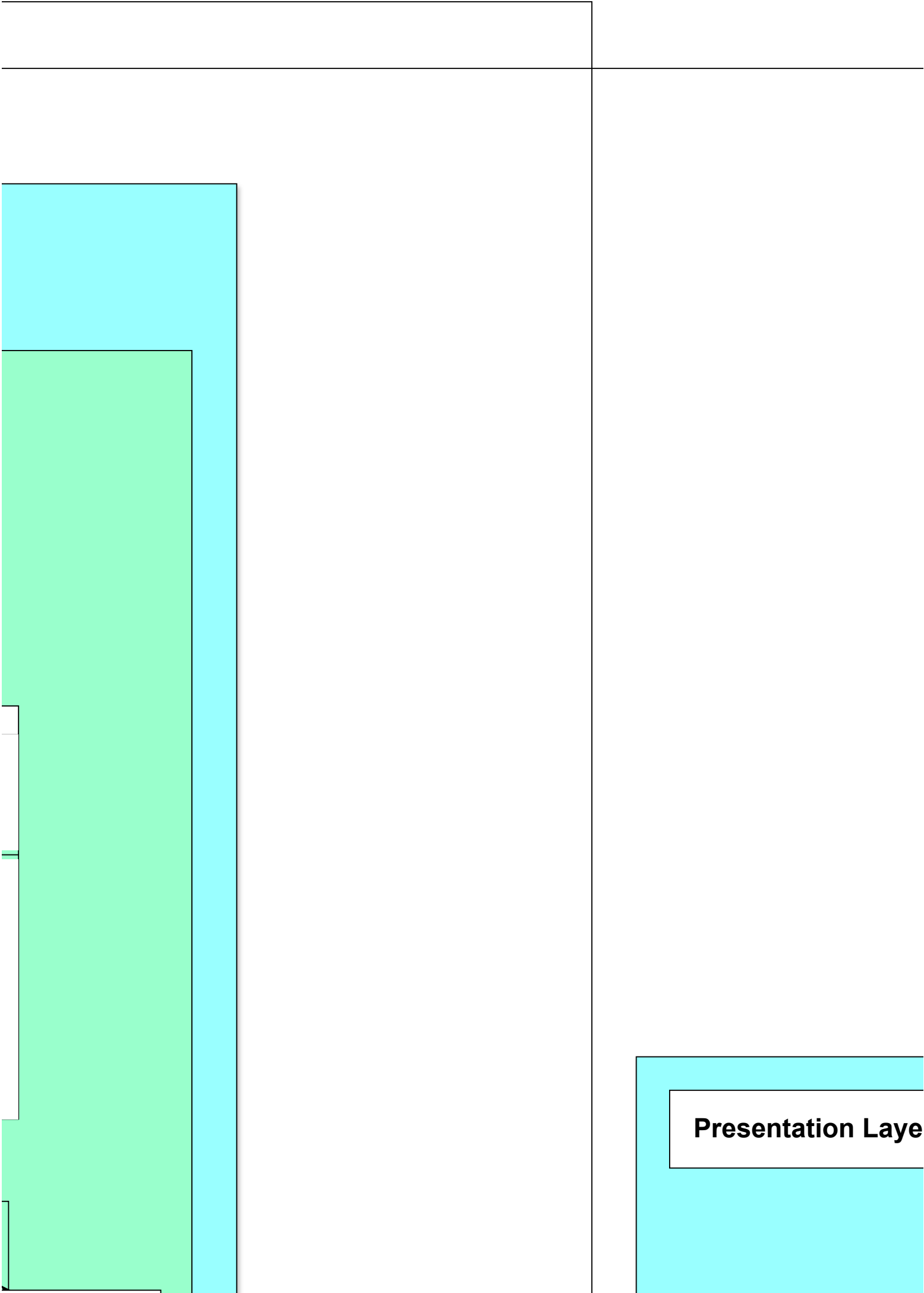




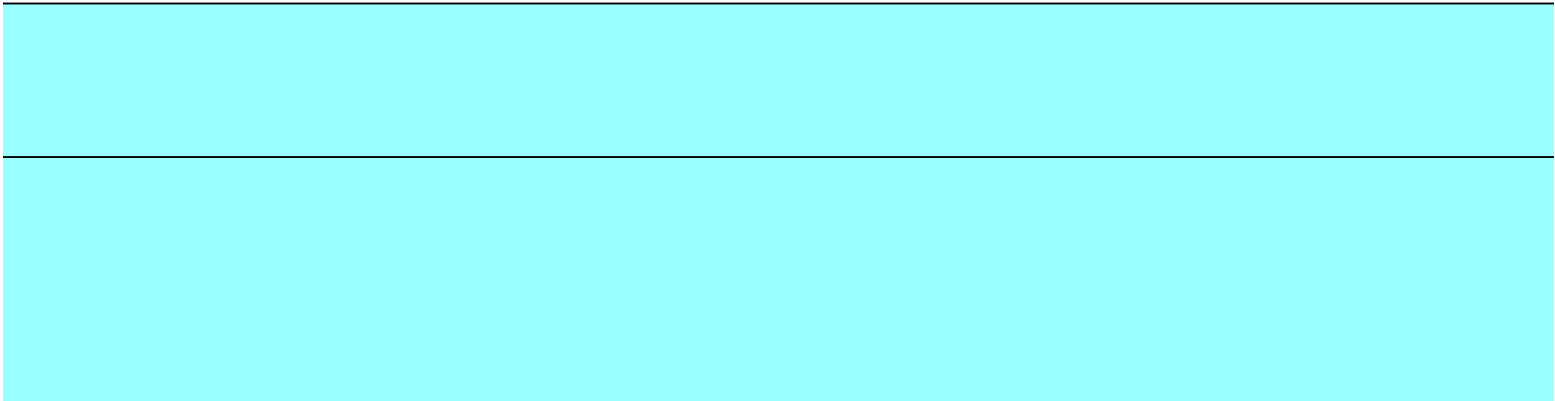


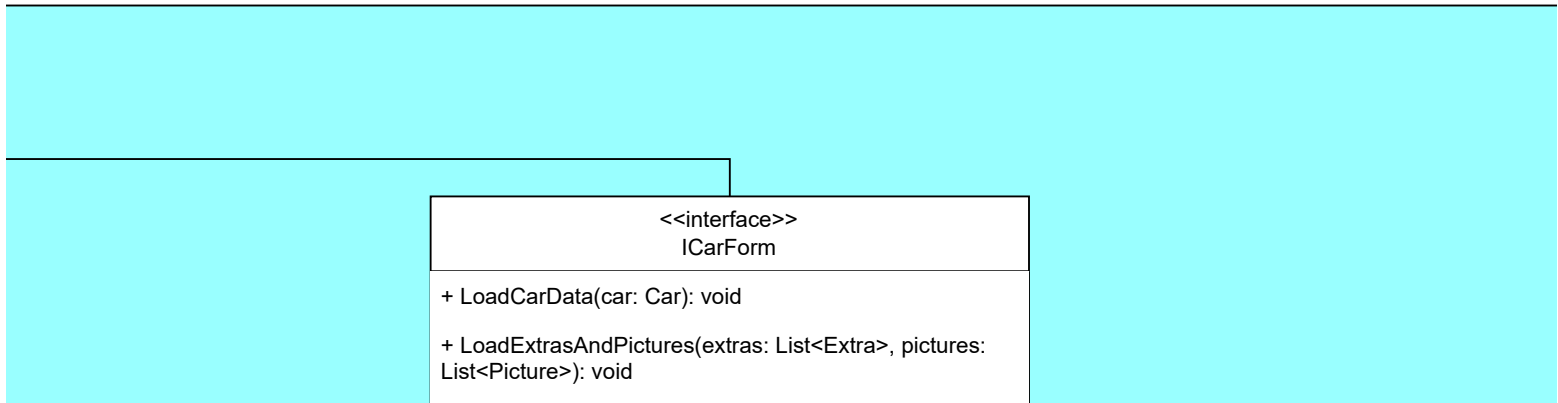


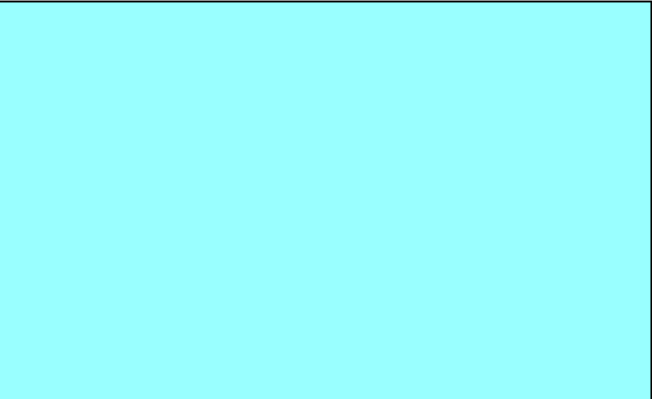








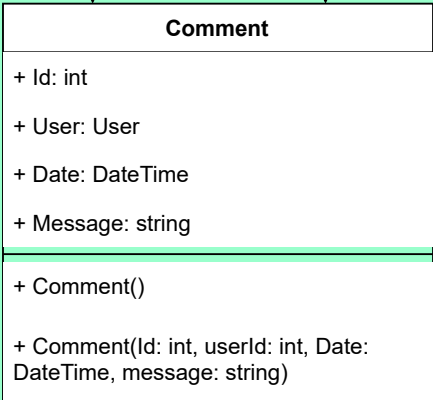




DTO Layer

+ GetCommentsById(int id); List<Comment>

0..*

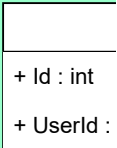
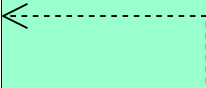
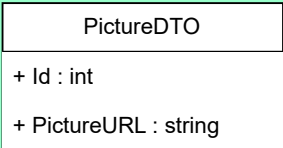
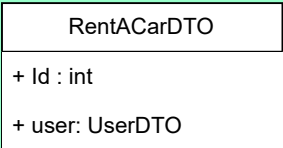


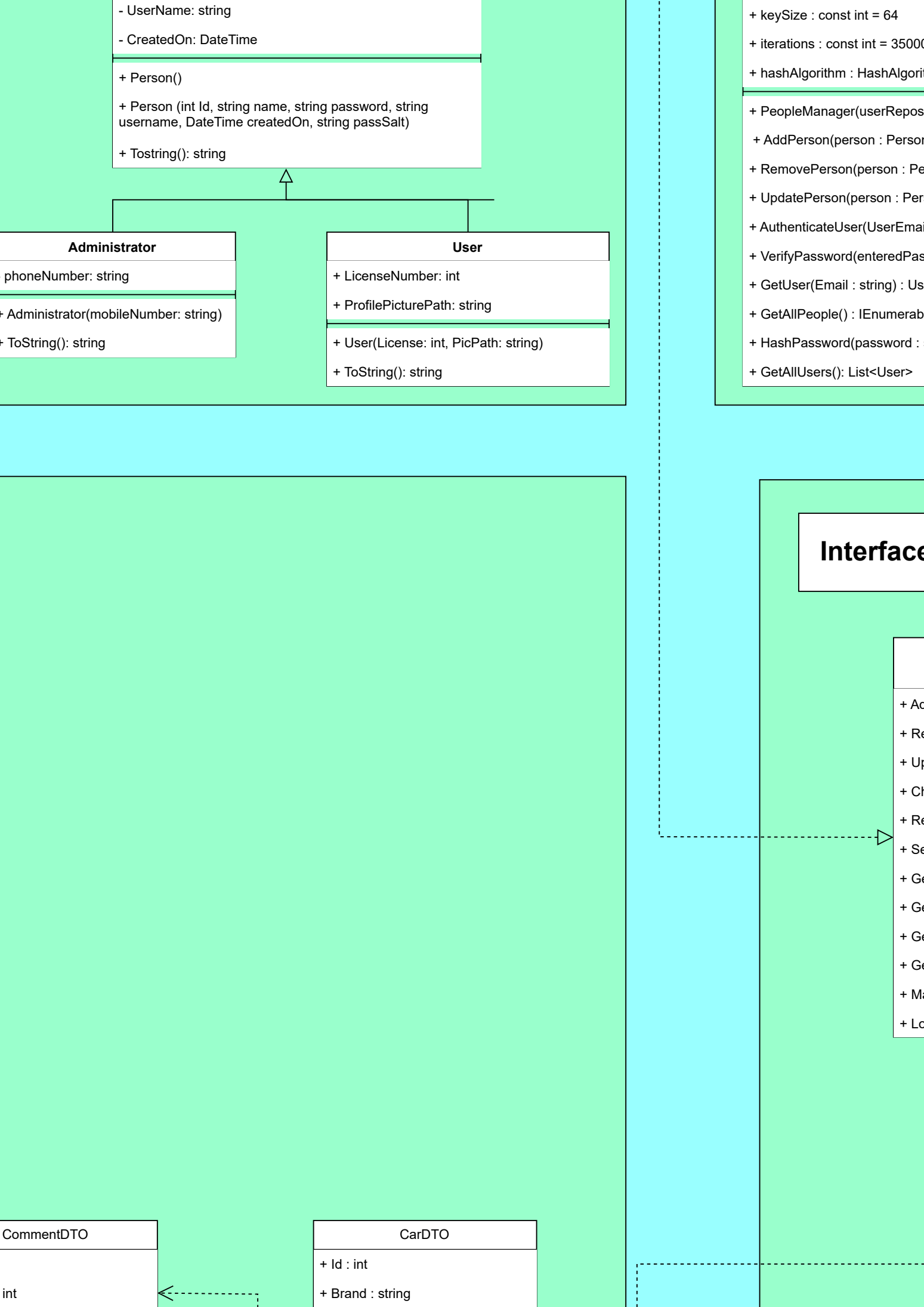
- PURPLE
- PINK
- BROWN
- GRAY
- CYAN
- MAGENTA
- NAVY
- LIME
- AQUA
- SILVER
- GOLD
- BRONZE
- VIOLET

-

+

+





00

thmName = SHA512

itory : IUserRepository, administratorRepository : IAdministratorRepository)

n) : string

erson) : string

son) : string

il : string, UserPass : string) : bool

sword : string, storedPass : string, base64Salt : string) : bool

er

le<Person>

string) : (string Hash, string Salt)

AdministratorRepository

- writer : IDataWriter

- remover : IDataRemover

- access : IDataAccess

- admins : List<Administrator>

+ AdministratorRepository(dataAccess: IDataAccess, dataWriter: IDataWriter)

+ AddAdmin(admin: Administrator) : string

+ RemoveAdmin(admin: Administrator) : string

+ UpdateAdmin(admin: Administrator) : string

+ GetAllAdministrators() : List<Administrator>

+ LoadAdmins() : string

Layer

<<interface>>
ICarManager

IdCar(car: Car, pictures: List<Picture>, extras: List<Extra>): string

RemoveCar(car: Car, picture: Picture, extra: Extra): string

UpdateCar(car: Car, pictures: List<Picture>, extras: List<Extra>): string

ChangeCarStatus(Car, string, CarStatus, out string): bool

RecordCarView(int, out string): bool

SearchForCar(index: int): Car

GetCars(): List<Car>

GetCarsASC(): List<Car>

GetCarsDESC(): List<Car>

GetCarById(carId: int): Car

MapCarDtoToCar(CarDTO): Car

LoadCars(): string

<<interface>>
ICarDataWriter

+ AddCar(Brand: string, Model: string, FirstRegistration: Date, int, Fuel: string, EngineSize: int, HP: int, Gearbox: string, NumberOfDoors: int, color: string, VIN: string, Status: string): void

+ AddCarDescription(CarId: int, Description: string, Price: decimal)

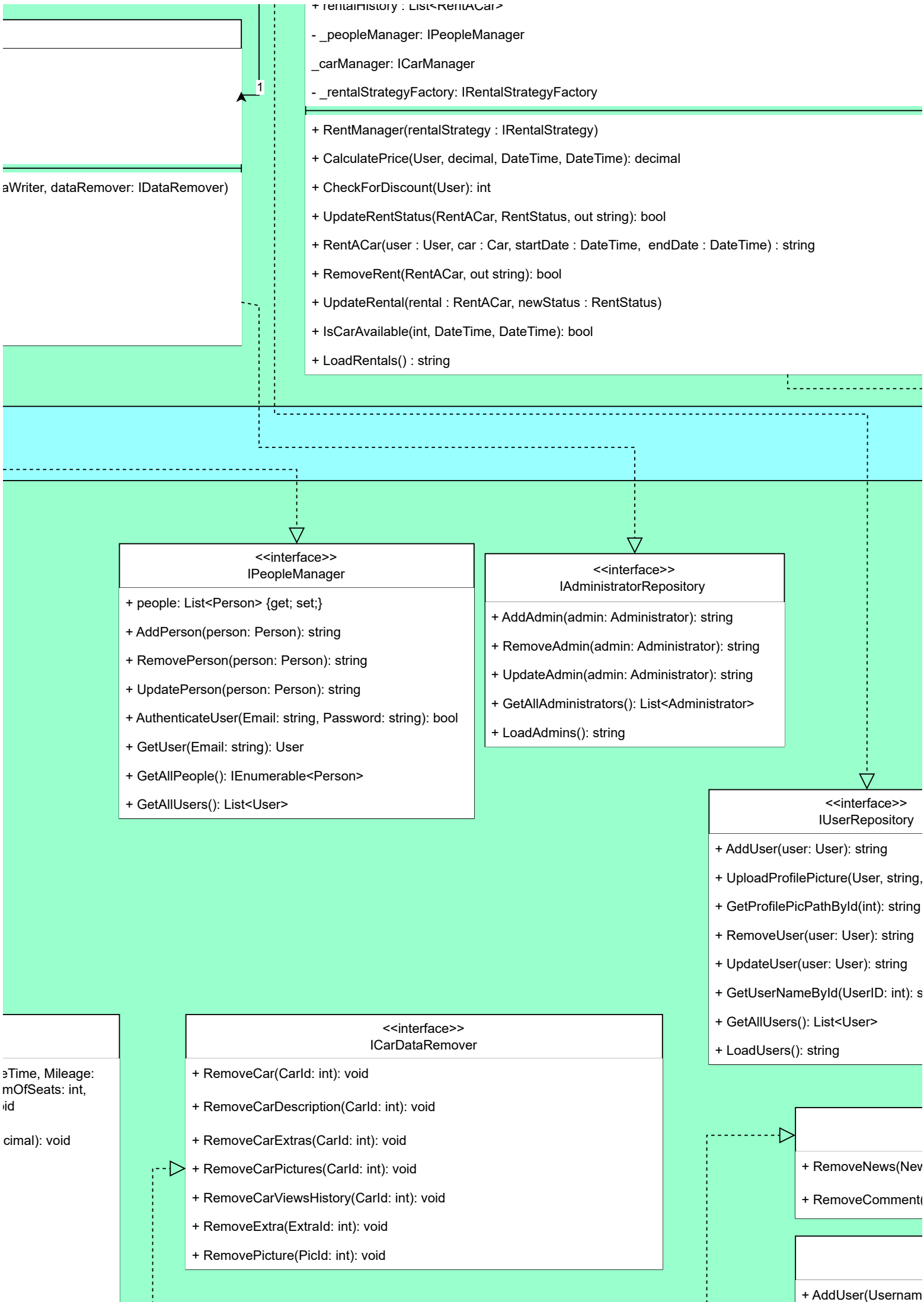
+ AddCarExtras(CarId: int, ExtraId: int): void

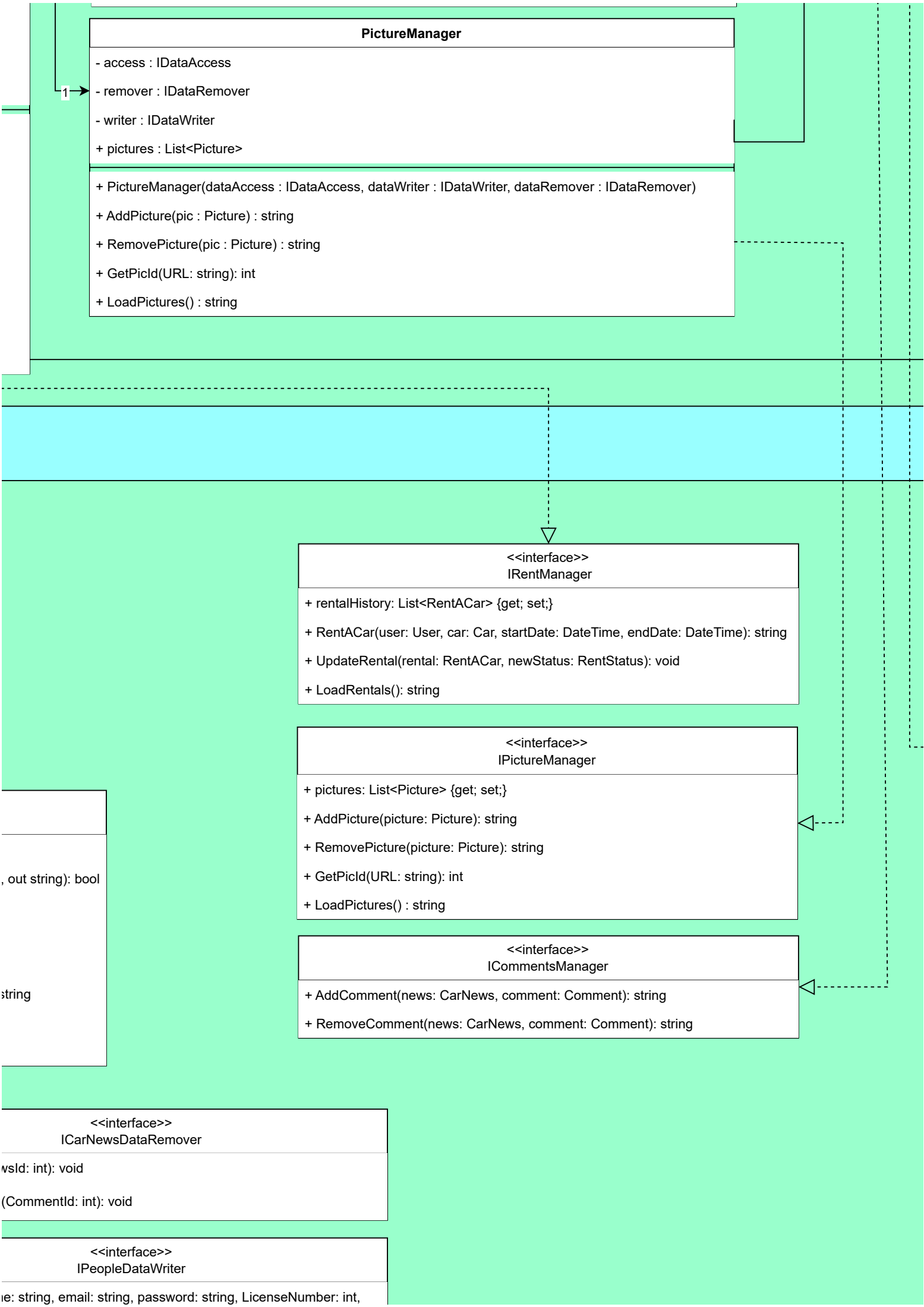
+ AddCarPictures(CarId: int, PictureId: int): void

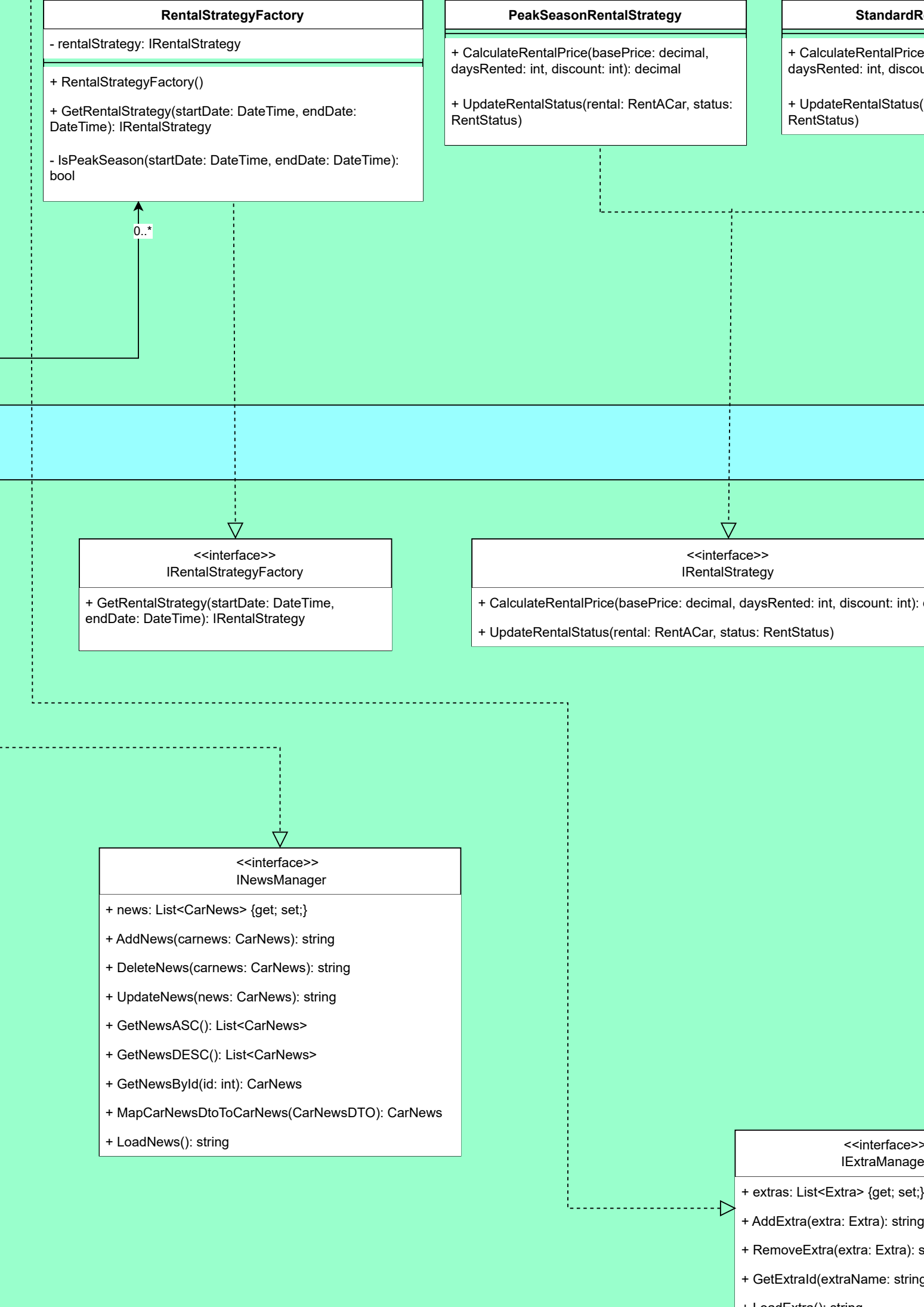
+ RecordCarView(CarId: int): void

+ UpdateCar(car: Car): void

+ UpdateCarDescription(car: Car): void







entalStrategy

```
(basePrice: decimal,
  amt: int): decimal
```

```
rental: RentACar, status:
```

decimal

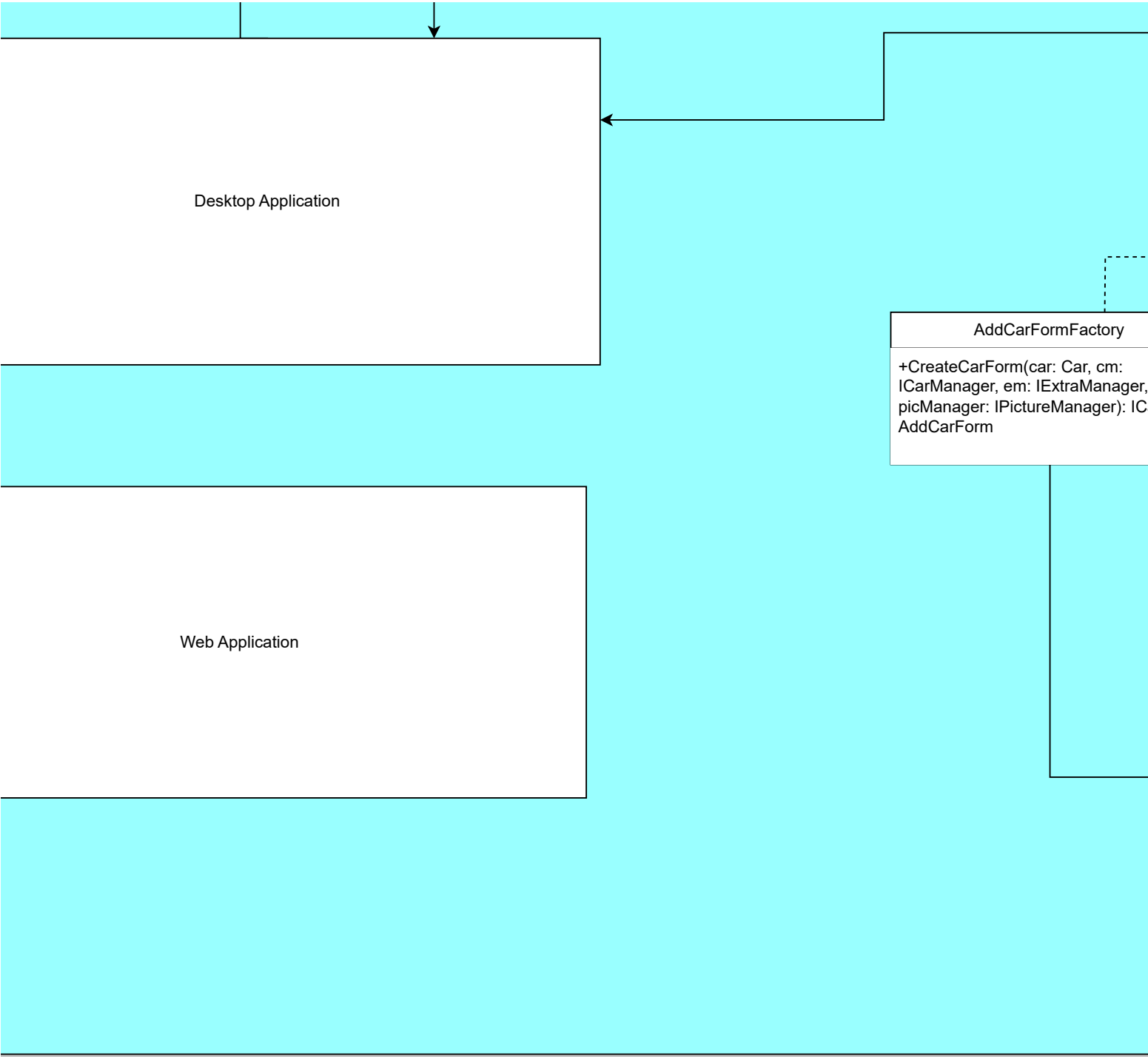
►
r

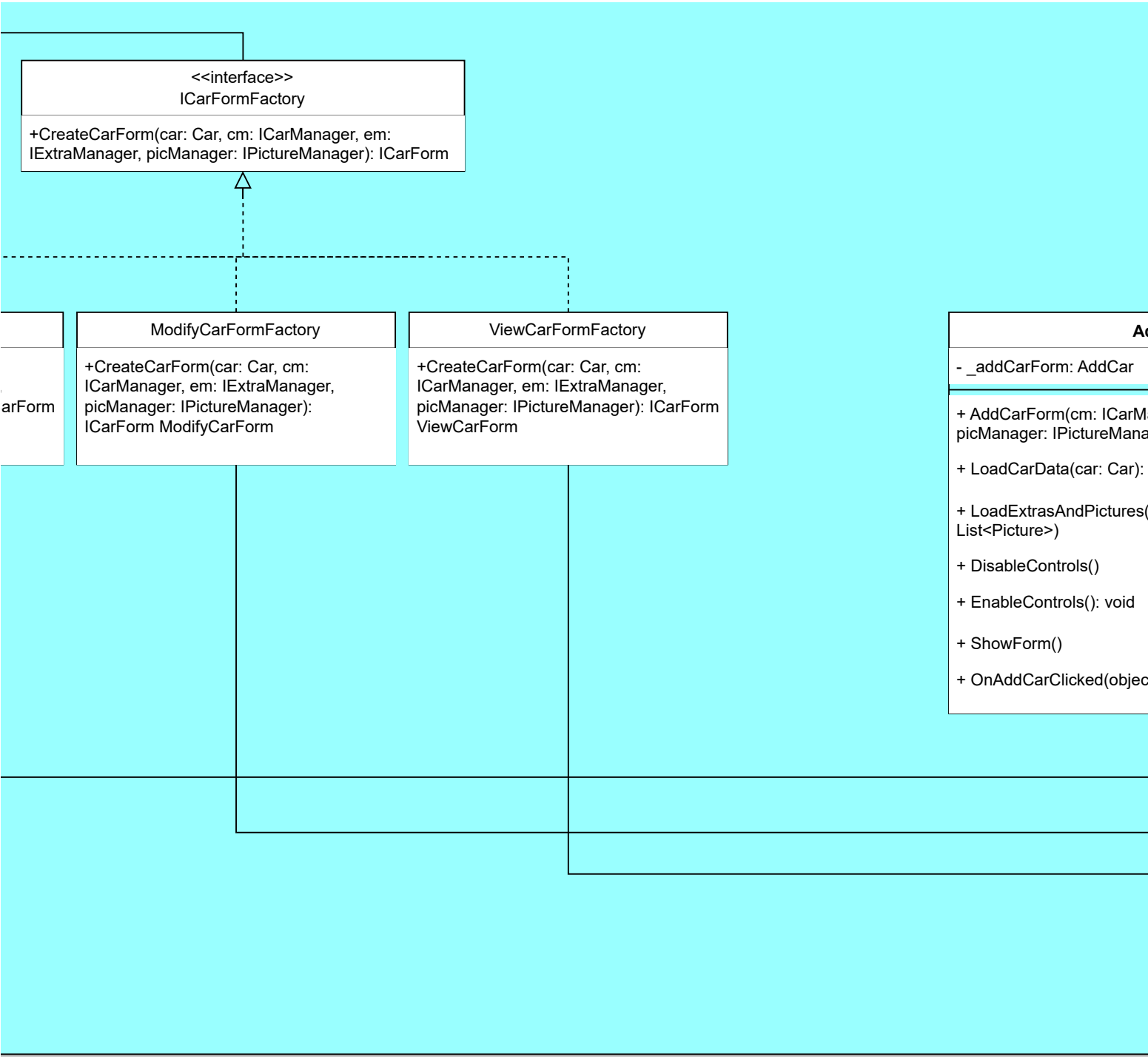
string

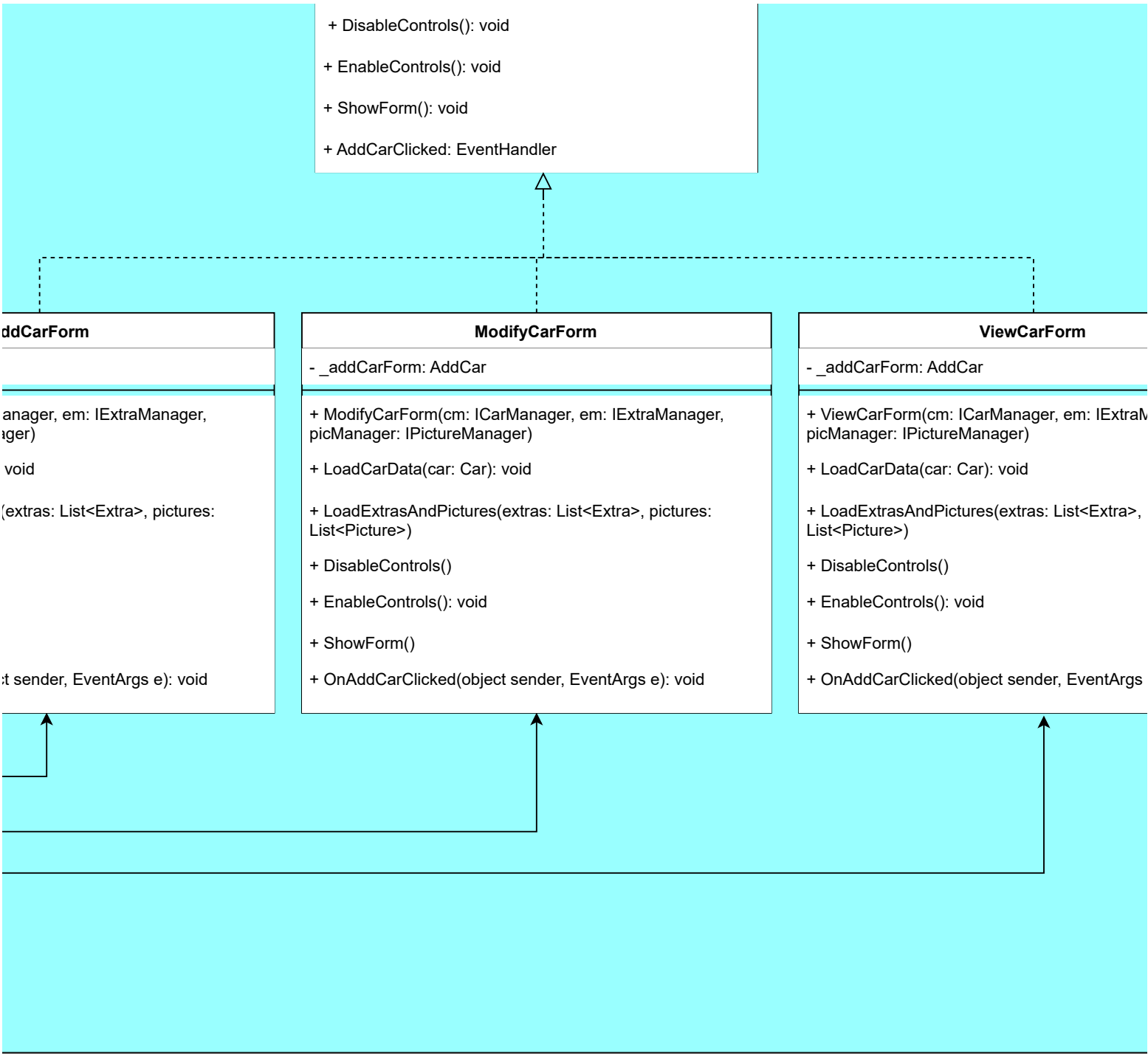
```

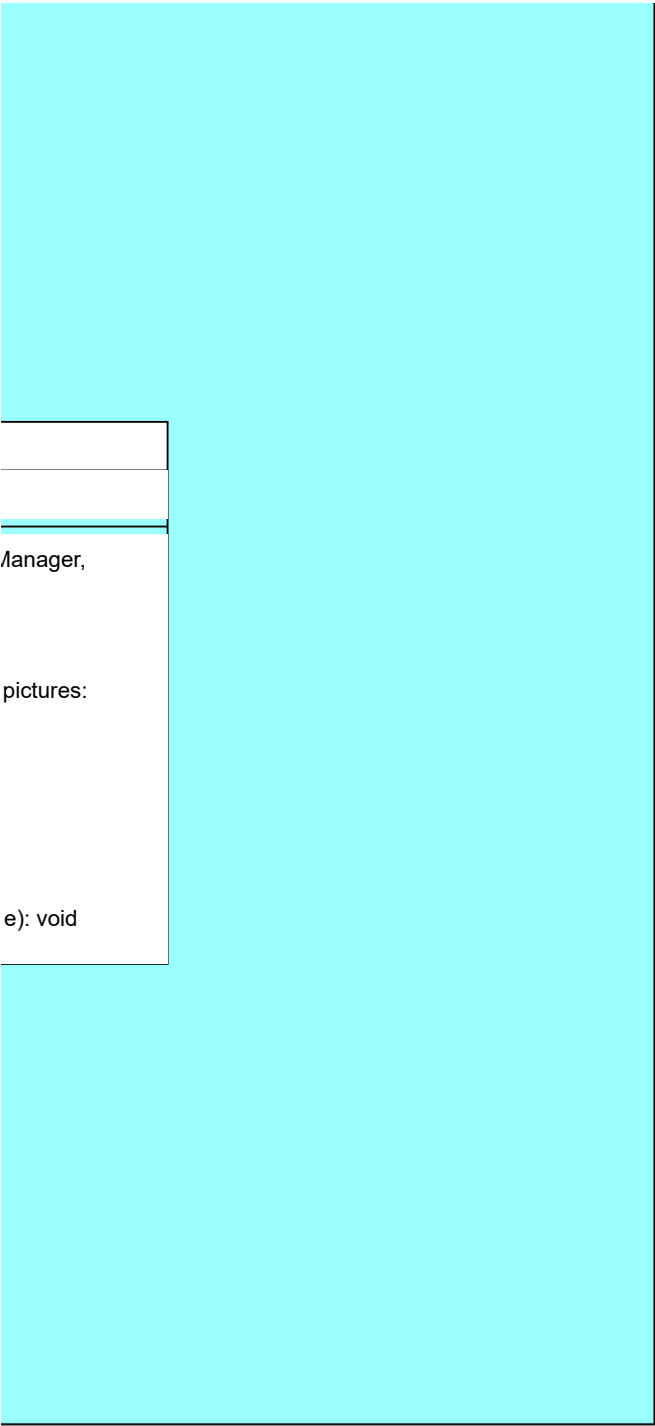
j): int

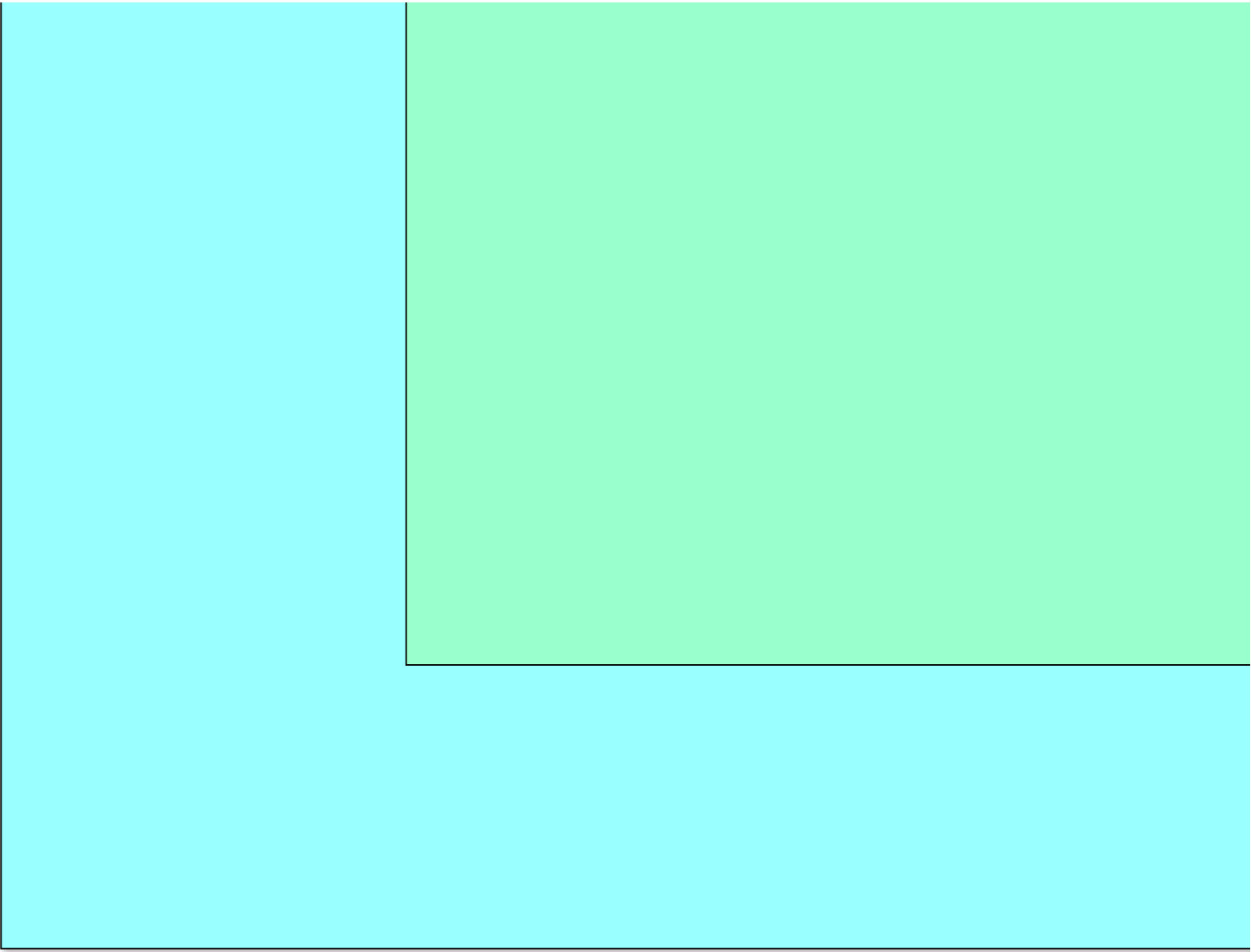
```

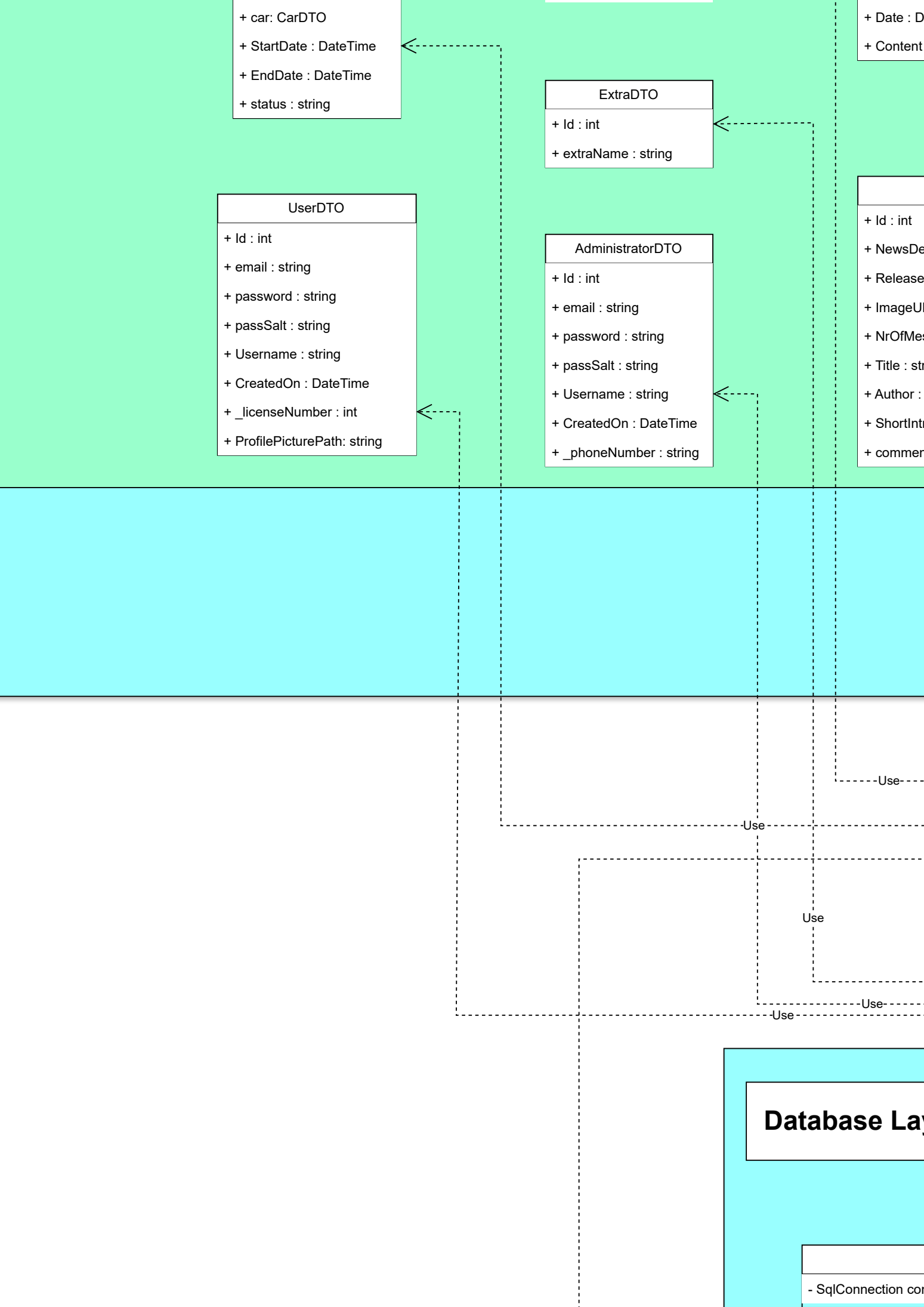


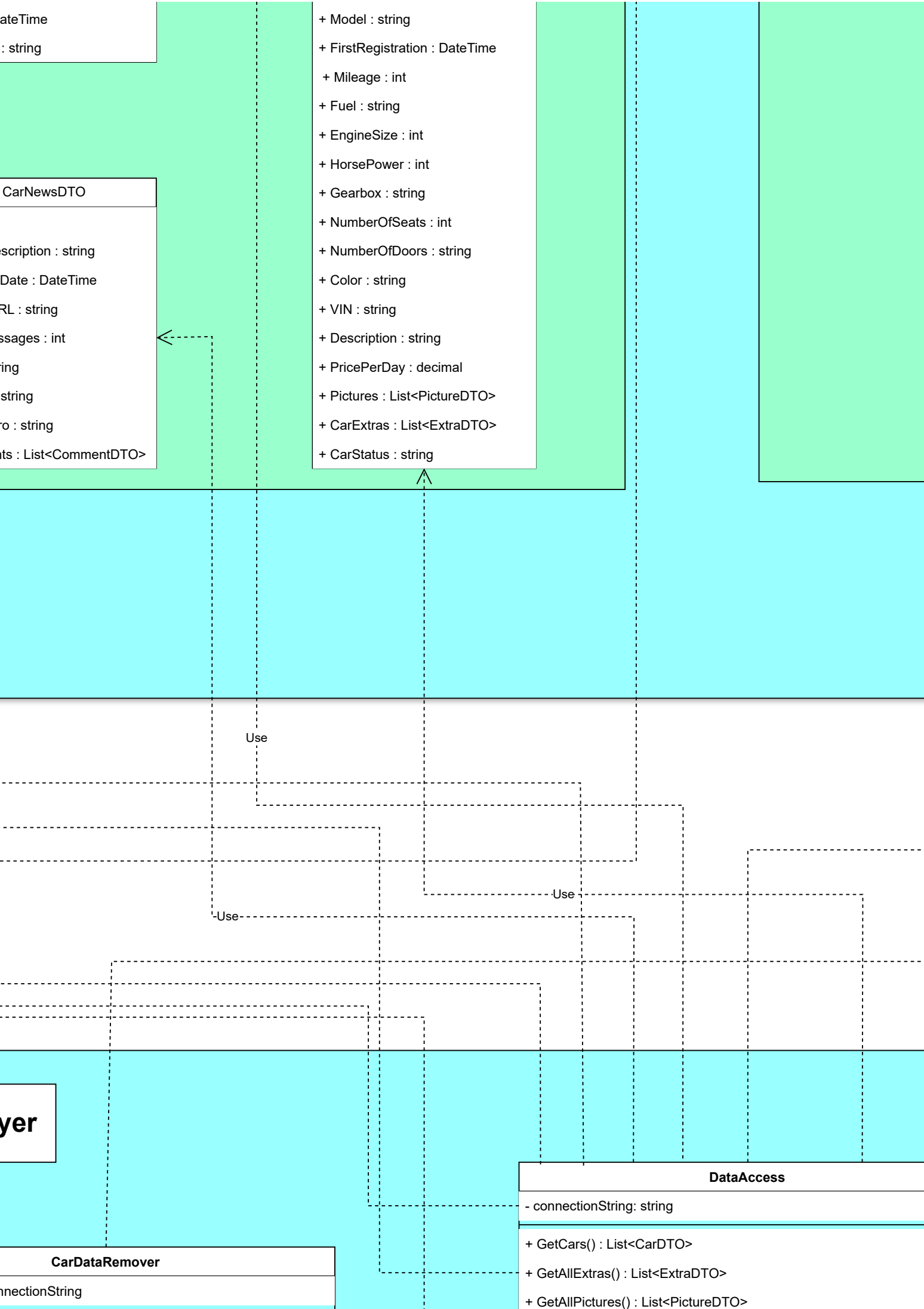


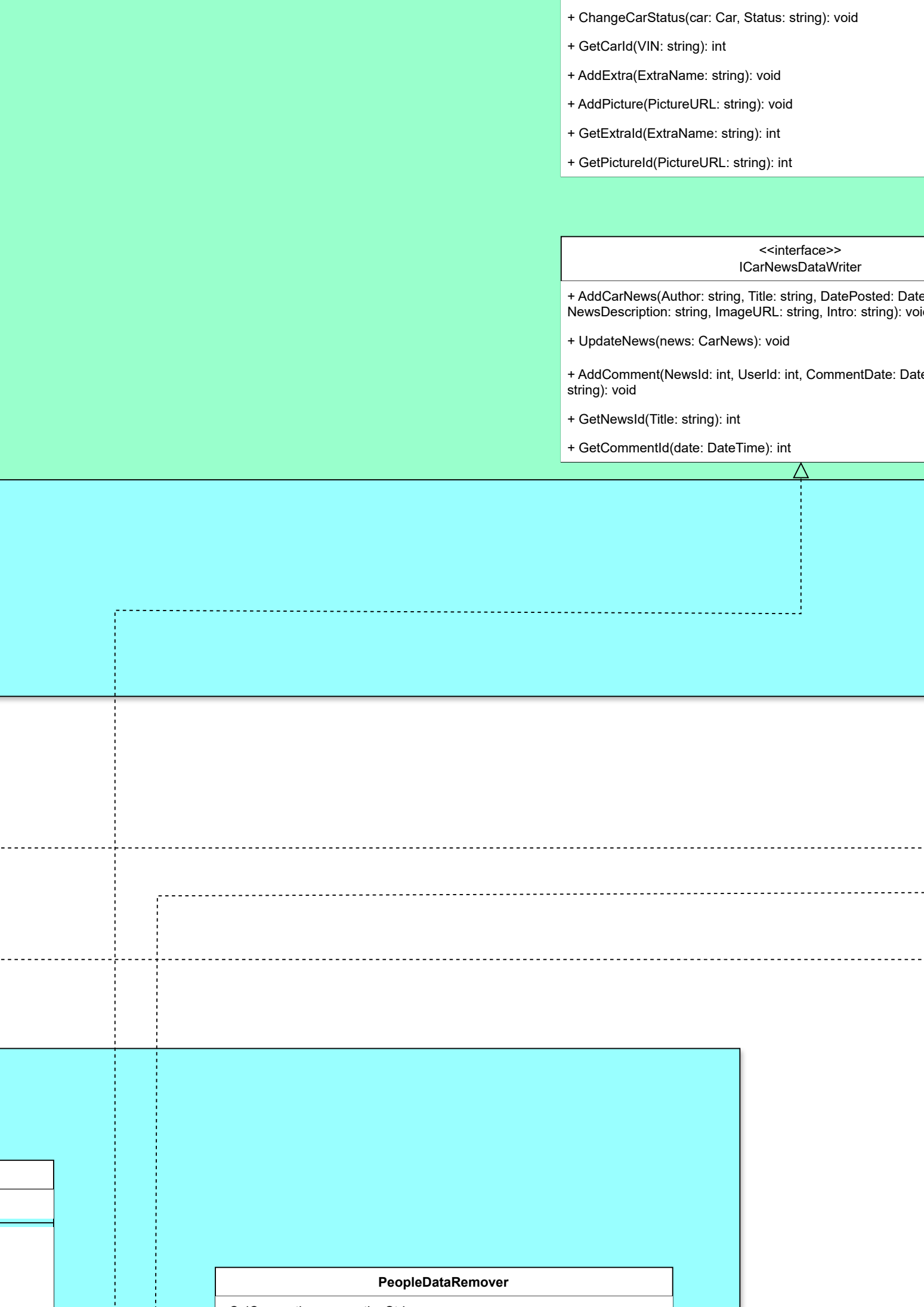














CreatedOn: DateTime

+ UploadProfilePicture()

+ AddAdmin(Username: string, Password: string, CreatedOn: DateTime)

+ UpdateUser(userId: int, Username: string, Password: string, licenseNumber: int, CreatedOn: DateTime)

+ UpdateAdministrator(adminId: int, Username: string, Password: string, phoneNumbers: List<string>)

+ RentACar(CarId: int, UserId: int, Status: string): void

+ UpdateRent(rentalId: int, Status: string)

+ RemoveUser(userId: int)

+ RemoveProfilePicture(userId: int)

+ RemoveAdministrator(adminId: int)

+ RemoveRental(rentalId: int)

ie, Salt: string): void

ire(User: user, relativeFilePath: string): void

me: string, email: string, password: string, PhoneNumber: string,
ie, PassSalt: string): void

d: int, Username: string, email: string, password: string,
CreatedOn: DateTime): void

tion(adminId: int, username: string, email: string, passwordHash:
r: string, CreatedOn: DateTime, PassSalt: string): void

int, UserId: int, StartDate: DateTime, EndDate: DateTime,

: RentACar): void

<<interface>>
IPeopleDataRemover

rd: int): void

ture(UserId: int): void

minId: int): void

nt: RentACar): void



