

Đặc tả bài toán:

Cho mô hình lớp một phần của ứng dụng Hệ thống đặt vé xem phim trực tuyến. Mỗi khách hàng (*Customer*) có thể có nhiều lượt đặt (*booking*) vé (*Ticket*) xem phim, nhưng mỗi lượt đặt vé chỉ được thực hiện bởi một khách hàng. Một buổi chiếu phim (*Show*) chiếu một bộ phim (*Movie*), có thể có nhiều vé xem phim và một bộ phim có thể có nhiều buổi chiếu phim.

Thông tin bộ phim bao gồm: mã số bộ phim (*id*), tên bộ phim (*title*), thể loại (*genre*), năm phát hành (*release year*), đạo diễn (*director*), thời lượng tính bằng phút (*duration*) và dàn diễn viên (*actors*).

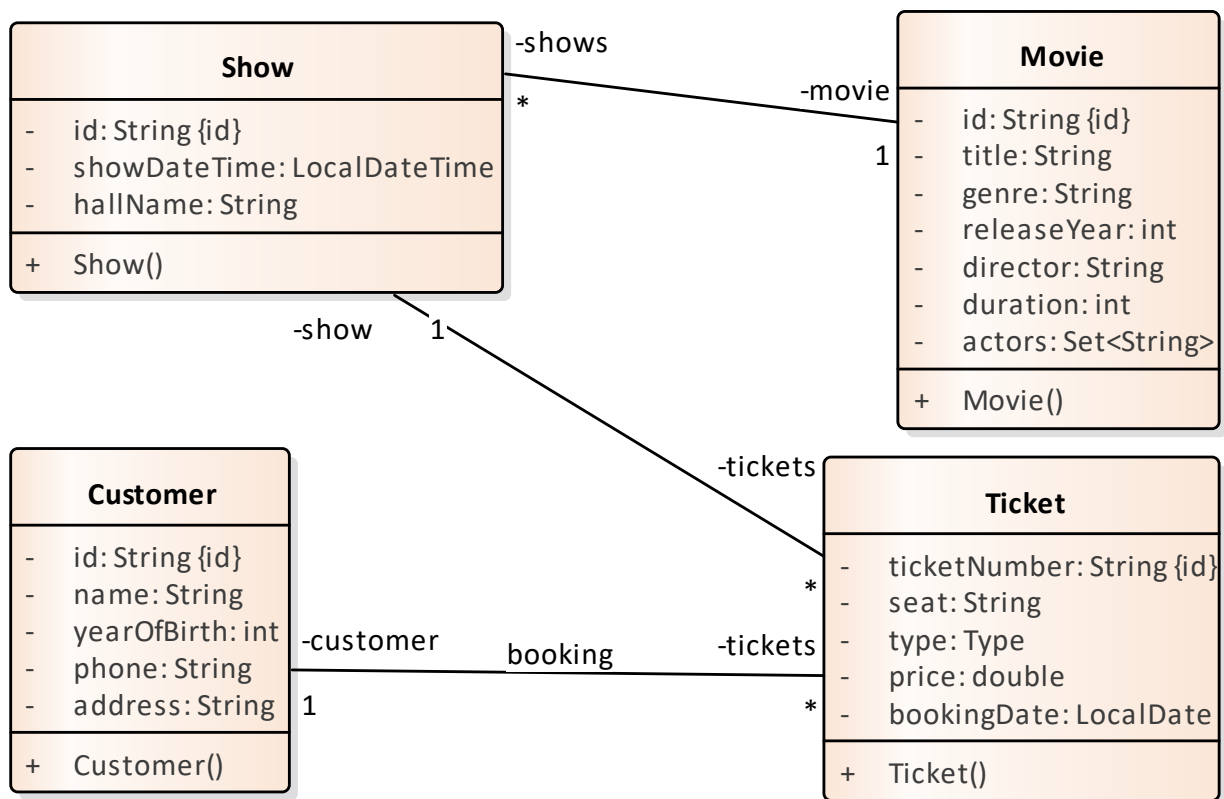
Thông tin khách hàng bao gồm: mã số khách hàng (*id*), tên khách hàng (*name*), năm sinh (*year of birth*), số điện thoại (*phone*), địa chỉ (*address*).

Thông tin buổi chiếu phim bao gồm: mã số buổi chiếu (*id*), ngày giờ chiếu (*show date time*), tên phòng chiếu (*hall name*).

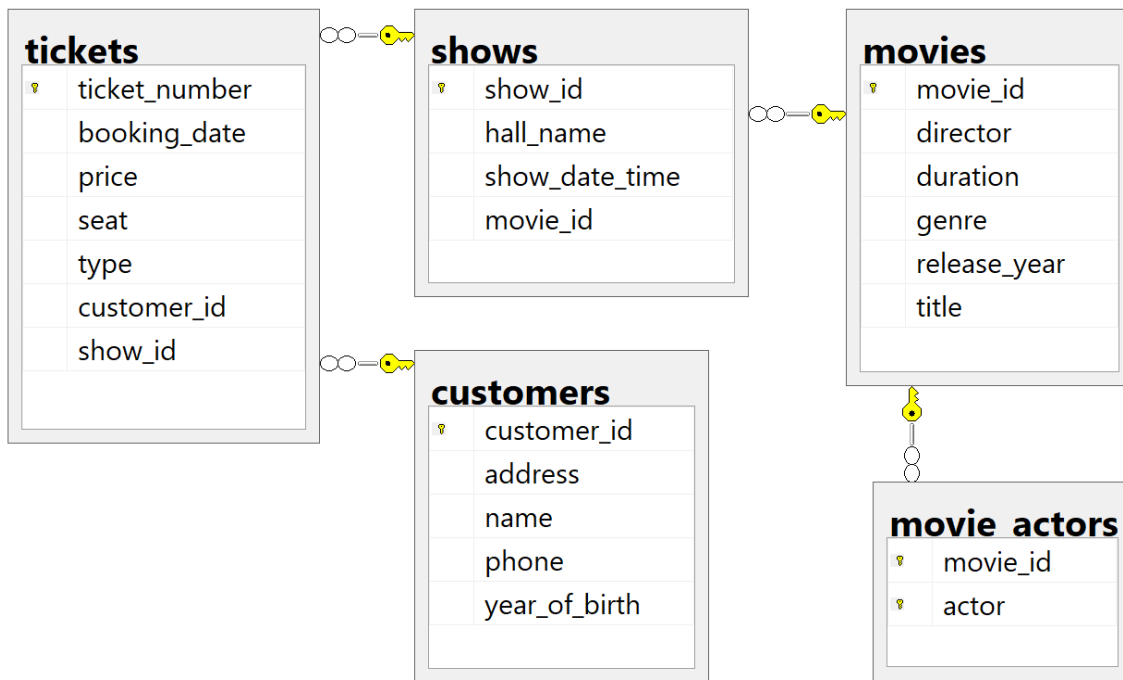
Thông tin vé xem phim bao gồm: số vé (*ticket number*), số ghế (*seat*), loại vé (*type*) có 2 loại: VIP và Standard, giá vé (*price*).

Khi khách hàng đặt vé xem phim, hệ thống sẽ lưu thông tin ngày đặt vé (*booking date*).

Mô hình lớp (*class diagram*)



Mô hình CSDL (database diagram)



Tạo các project tên gồm: `HọTênSV_MãSốSV_SốMáy_[Server/Client]`. Dùng ngôn ngữ lập trình JAVA kết nối CSDL quan hệ và hiện thực các yêu cầu sau:

Câu 1: Dùng JPA ORM ánh xạ các thực thể trong mô hình lớp sang mô hình cơ sở dữ liệu quan hệ (*tên database là [tên và mã số của sinh viên]*).

Câu 2: Nạp dữ liệu mẫu được cho sẵn hoặc dùng datafaker để nhập dữ liệu các bảng.

Hiện thực chương trình dựa trên mô hình lập trình mạng client - server sử dụng RMI

Câu 3: Thiết lập một server triển khai trên mạng LAN với các chức năng như sau: (*port là 4 số cuối MSSV, hostname là tên máy sinh viên làm bài*)

a) Viết các phương thức thêm/xóa/sửa/tìm kiếm theo thuộc tính khóa và lấy danh sách các đối tượng.

Trong đó:

- Chỉ cho phép cập nhật một buổi chiếu phim (*show*) nếu buổi chiếu phim đó chưa có khách hàng nào đặt vé.
- Thêm bộ phim mới, mã số bộ phim (*id*) phải bắt đầu là ký tự "M" theo sau ít nhất 3 ký số và thời lượng (*duration*) phải lớn hơn 0.

b) Tạo các index tăng dần trên ngày giờ chiếu phim và đạo diễn phim.

Liệt kê các buổi chiếu phim (*show*) trong ngày hiện tại, chiếu những bộ phim (*movie*) do đạo diễn nào đó đạo diễn (*tìm tương đối*).

+ `listShowsByCurrentDateAndDirector(String director): List<Show>`

c) Thống kê doanh số bán vé theo từng bộ phim, kết quả sắp xếp theo tựa phim.

+ `listTicketSalesByMovieSortedByTitle(): Map<Movie, Double>`

Câu 4: Viết các lớp kiểm thử (*dùng JUnit 5*), tạo các test case cho các phương thức ở câu 3.

Câu 5: Hiện thực chương trình client - server để thực thi các phương thức ở câu 3.