**LAB 7**

Một công ty cần viết một ứng dụng web cho công việc bán hàng của họ. Thông tin cần lưu trữ bao gồm:

* employee (**emp\_id**, full\_name, dob, email, phone, address, status)
* product (**product\_id**, name, description, unit, manufacturer\_name, status)
* customer (**cust\_id**, cust\_name, email, phone, address)
* product\_image (**product\_id, image\_id**, path, alternative)
* order (**order\_id**, order\_date, emp\_id, cust\_id)
* order\_detail **(order\_id, product\_id**, quantity, price, note)
* product\_price (**product\_id, price\_date\_time**, price, note)

Diễn giải:

* Một product có nhiều image, một image thuộc về một product. Status chỉ trạng thái kinh doanh của sản phẩm: 1- đang kinh doanh, 0 - tạm ngưng, -1 - không kinh doanh nữa.
* Employee có status: 1- đang làm việc, 0 - tạm nghỉ, -1 – nghỉ việc.
* Một order có nhiều order\_detail, một order\_detail thuộc về một order.
* Một order thuộc về một employee, một employeecó nhiều order.
* Một customer có nhiều order, một order chỉ thuộc một customer.
* Một product\_detail có một giá (price) được lưu trong product\_price. Một giá được xác định bằng product\_id và price\_date\_time. Tại thời điểm lập order, giá được lấy với price\_date\_time gần nhất.

Yêu cầu:

Tạo một dự án sử dụng SpringBoot thực hiện:

* Hiển thị các sản phẩm, chọn vào giỏ hàng và thanh toán (giả lập việc thanh toán qua thẻ nếu có thể)
* Một trang admin cho phép:
  + Các thao tác CRUD cho các đối tượng.
  + Thống kê order theo ngày, theo khoảng thời gian.
  + Thống kê order theo nhân viên bán hàng trong 1 khoảng thời gian.
  + *\*\*\* và các chức năng khác tự nghĩ ra.*

Dự án được upload lên git-hub với trang mark-down thể hiện thiết kế, các kết quả đạt được.

HƯỚNG DẪN:

Diagrams:

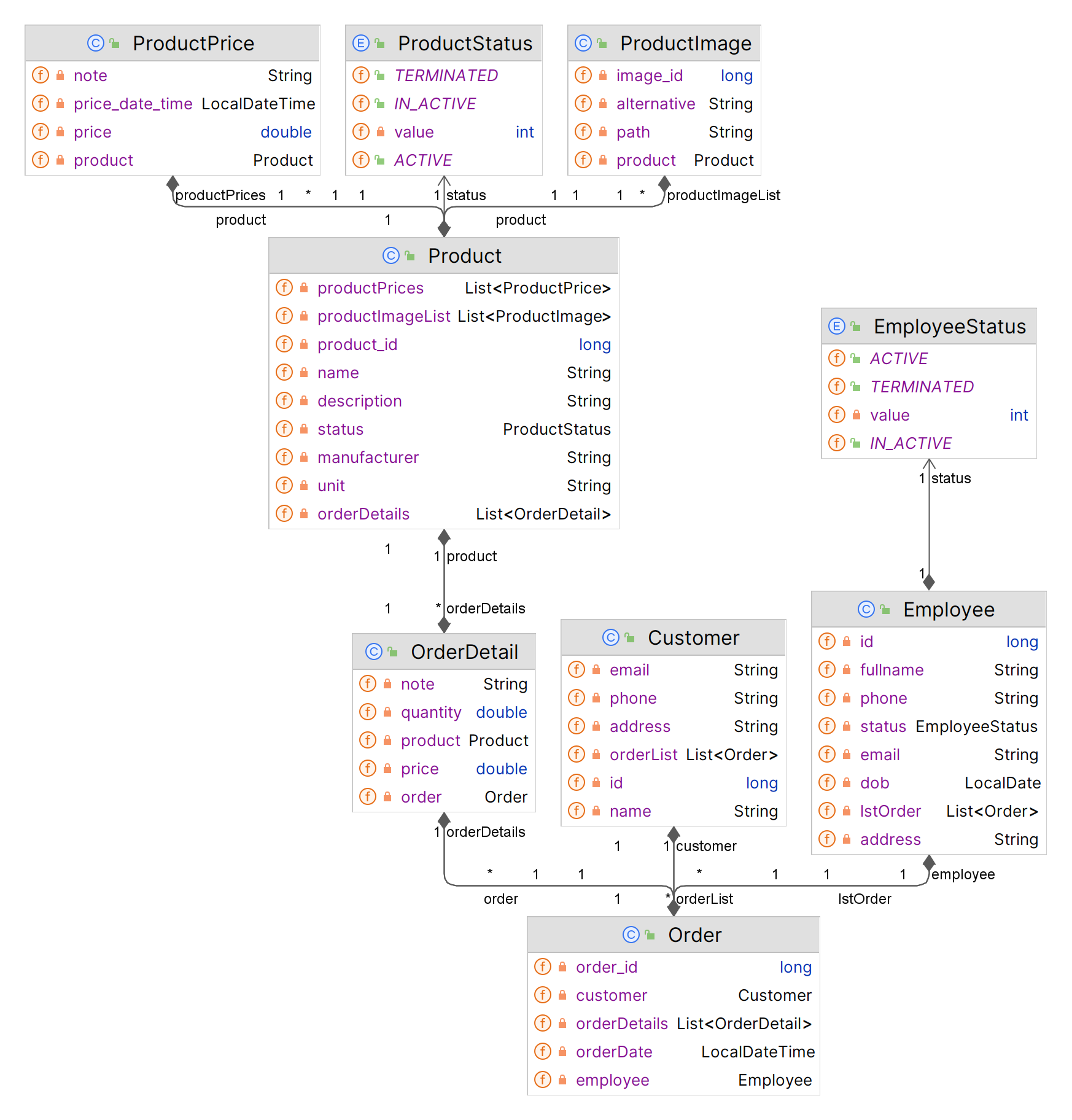


Figure : Class diagram

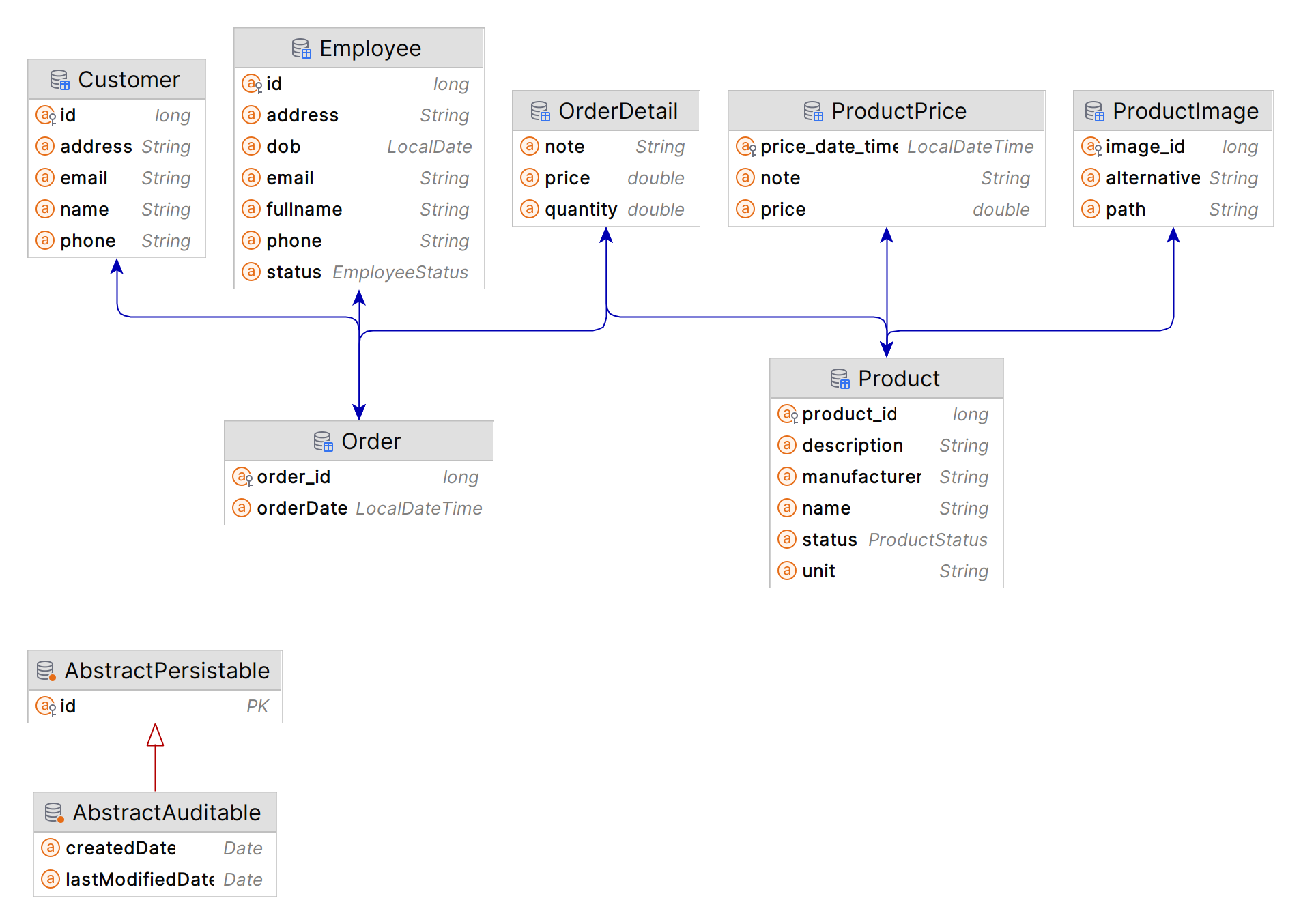


Figure : Entity Relationship Diagram

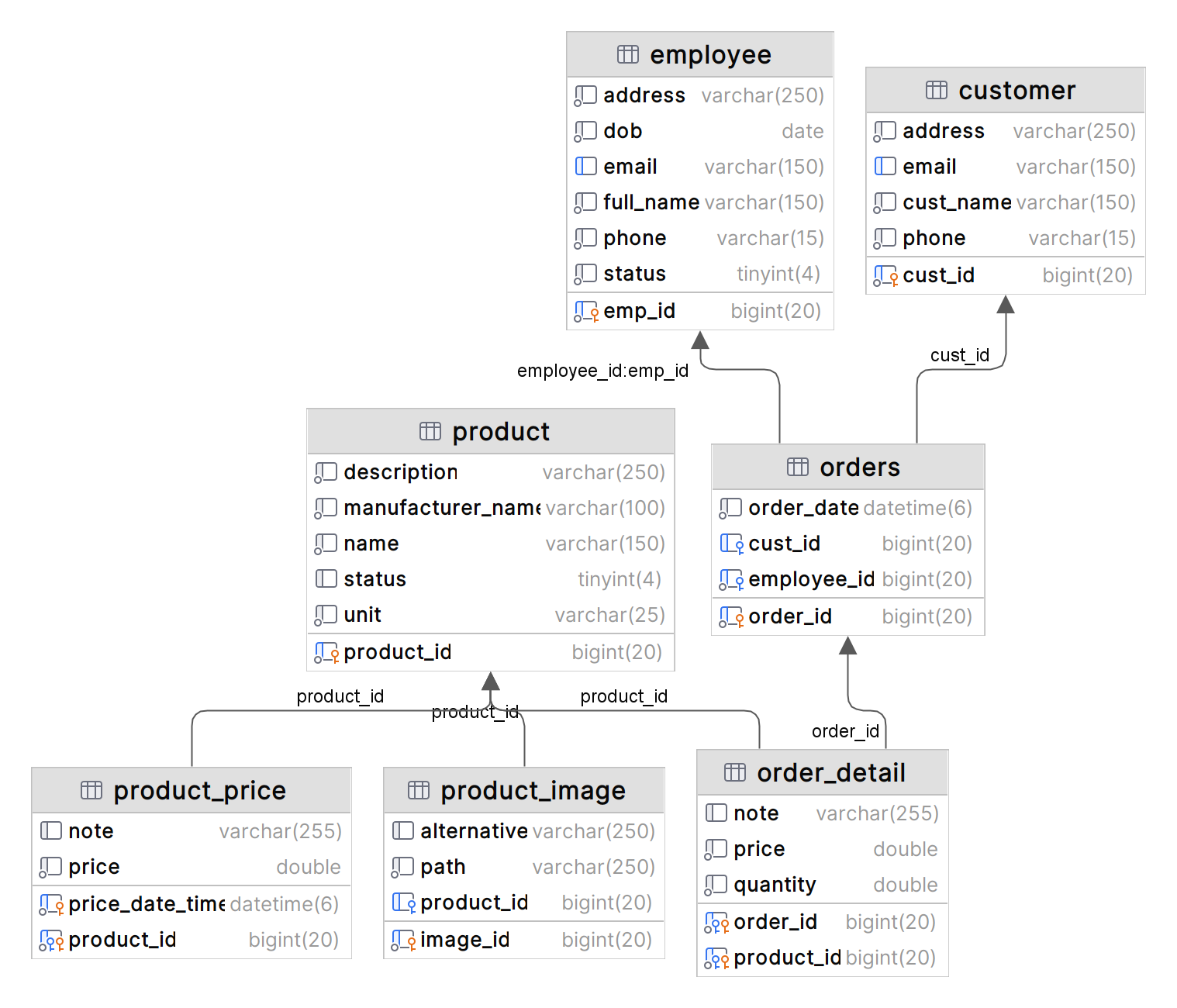


Figure : Database Diagram