



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

دانشکده ریاضی و علوم کامپیوتر

درس پایگاه داده‌ها

پیاده‌سازی پایگاه داده فروشگاه آنلاین

نگارش: کیارش مختاری دیزجی - ۹۸۳۰۰۳۲

استاد درس: دکتر حمید قراگوزلو

تیر ۱۴۰۳



فهرست مطالب

۲	۱ فاز اول: طراحی نمودار ER برای پایگاه داده فروشگاه آنلاین
۸	۲ فاز دوم: نرمال‌سازی
۱۶	۳ فاز سوم: ایجاد جداول و اضافه کردن دیتا
۱۸	۴ فاز چهارم: طراحی پرس و جو



۱ فاز اول: طراحی نمودار ER برای پایگاه داده فروشگاه آنلاین

نمودار ER برای سیستم خرید آنلاین شامل چندین جدول است که موجودیت‌های مختلف و روابط آن‌ها را نشان می‌دهد. در ادامه توضیحات دقیقی از جداول، کلیدها و کاردینالیتی آن‌ها آمده است:

۱. جدول کاربران (Users Table)

- کلید اصلی: `user_id`
- ستون‌ها: `user_id`, `username`, `password`, `email`, `contact_number`, `street`, `city`, `state`, `postal_code`, `country`, `registration_date`
- توضیح: اطلاعات مربوط به کاربرانی که در پلتفرم ثبت‌نام می‌کنند را ذخیره می‌کند. کاربران می‌توانند وارد سیستم شوند، پروفایل خود را به‌روزرسانی کنند و سفارشات ثبت کنند.
- روابط:
 - 1:N با سفارشات (Orders): یک کاربر می‌تواند چندین سفارش ثبت کند.
 - 1:N با سبد خرید (ShoppingCart): یک کاربر می‌تواند چند سبد خرید داشته باشد.
 - 1:N با تاریخچه خرید (PurchaseHistory): یک کاربر می‌تواند چندین ورودی در تاریخچه خرید داشته باشد.
 - 1:N با نظرات (Comments): یک کاربر می‌تواند چندین نظر بنویسد.

۲. جدول مدیران (Managers Table)

- کلید اصلی: `manager_id`
- ستون‌ها: `manager_id`, `username`, `password`, `email`, `registration_date`
- توضیح: اطلاعات مربوط به مدیرانی که می‌توانند نظرات و سیستم را مدیریت کنند را ذخیره می‌کند.
- روابط:
 - 1:N با نظرات (Comments): یک مدیر می‌تواند چندین نظر را مدیریت کند.



۳. جدول دسته‌بندی‌ها (Categories Table)

- کلید اصلی: category_id
- ستون‌ها: name, category_id
- توضیح: دسته‌بندی‌های مربوط به محصولات را ذخیره می‌کند.
- روابط:
- 1:N با محصولات (Products): یک دسته‌بندی می‌تواند چندین محصول داشته باشد.

۴. جدول برندها (Brands Table)

- کلید اصلی: brand_id
- ستون‌ها: status, name, brand_id
- توضیح: اطلاعات مربوط به برندهای محصولات را ذخیره می‌کند. وضعیت می‌تواند active, inactive یا old باشد.
- روابط:
- 1:N با محصولات (Products): یک برند می‌تواند چندین محصول داشته باشد.

۵. جدول محصولات (Products Table)

- کلید اصلی: product_id
- ستون‌ها: product_id, name, description, price, stock, category_id, brand_id, status, created_at
- توضیح: جزئیات مربوط به محصولات موجود در فروشگاه را ذخیره می‌کند.
- روابط:
- N:1 با دسته‌بندی‌ها (Categories): هر محصول متعلق به یک دسته‌بندی است.
- N:1 با برندها (Brands): هر محصول متعلق به یک برند است.
- 1:N با جزئیات سفارشات (OrderDetails): یک محصول می‌تواند در چندین جزئیات سفارش ظاهر شود.
- 1:N با اقلام سبد خرید (CartItems): یک محصول می‌تواند در چندین اقلام سبد خرید ظاهر شود.
- 1:N با نظرات (Comments): یک محصول می‌تواند چندین نظر داشته باشد.
- 1:N با تخفیفات محصولات (ProductDiscounts): یک محصول می‌تواند چندین تخفیف داشته باشد.



۶. جدول سفارشات (Orders Table)

- کلید اصلی: `order_id`
- ستون‌ها: `shipping_info_id`, `total_amount`, `status`, `order_date`, `user_id`, `order_id`
- توضیح: اطلاعات مربوط به سفارشات ثبت شده توسط کاربران را ذخیره می‌کند. وضعیت می‌تواند `processing`, `pending`, `shipped`, `cancelled`, `delivered` یا `returned` باشد.
- روابط:
- N:1 با کاربران (Users): هر سفارش توسط یک کاربر ثبت می‌شود.
- 1:N با جزئیات سفارشات (OrderDetails): یک سفارش می‌تواند چندین جزئیات سفارش داشته باشد.
- 1:N با تاریخچه خرید (PurchaseHistory): یک سفارش می‌تواند بخشی از چندین ورودی تاریخچه خرید باشد.
- N:1 با اطلاعات حمل و نقل (ShippingInfo): هر سفارش یک اطلاعات حمل و نقل دارد.

۷. جدول اطلاعات حمل و نقل (ShippingInfo Table)

- کلید اصلی: `shipping_info_id`
- ستون‌ها: `delivery_date`, `shipping_date`, `carrier`, `tracking_number`, `shipping_info_id`, `status`
- توضیح: اطلاعات مربوط به جزئیات حمل و نقل سفارشات را ذخیره می‌کند. وضعیت می‌تواند `shipped`, `pending`, `delivered` یا `returned` باشد.
- روابط:
- 1:N با سفارشات (Orders): یک اطلاعات حمل و نقل می‌تواند به چندین سفارش مرتبط باشد.



۸. جدول جزئیات سفارشات (OrderDetails Table)

- کلید اصلی: `order_detail_id`
- ستون‌ها: `price`, `quantity`, `product_id`, `order_id`, `order_detail_id`
- توضیح: اطلاعات دقیق مربوط به محصولات در هر سفارش را ذخیره می‌کند.
- روابط:
- N:1 با سفارشات (Orders): هر جزئیات سفارش متعلق به یک سفارش است.
- N:1 با محصولات (Products): هر جزئیات سفارش حاوی یک محصول است.

۹. جدول سبد خرید (ShoppingCart Table)

- کلید اصلی: `cart_id`
- ستون‌ها: `updated_at`, `created_at`, `user_id`, `cart_id`
- توضیح: اطلاعات مربوط به سبد خرید هر کاربر را ذخیره می‌کند.
- روابط:
- N:1 با کاربران (Users): هر سبد خرید متعلق به یک کاربر است.
- 1:N با اقلام سبد خرید (CartItem): یک سبد خرید می‌تواند چندین اقلام داشته باشد.

۱۰. جدول اقلام سبد خرید (CartItem Table)

- کلید اصلی: `cart_item_id`
- ستون‌ها: `quantity`, `product_id`, `cart_id`, `cart_item_id`
- توضیح: اطلاعات مربوط به اقلام در هر سبد خرید را ذخیره می‌کند.
- روابط:
- N:1 با سبد خرید (ShoppingCart): هر اقلام سبد خرید متعلق به یک سبد خرید است.
- N:1 با محصولات (Products): هر اقلام سبد خرید حاوی یک محصول است.



۱۱. جدول تاریخچه خرید (PurchaseHistory Table)

- کلید اصلی: history_id
- ستون‌ها: history_id, user_id, order_id, purchase_date
- توضیح: تاریخچه خرید کاربران را ذخیره می‌کند.
- روابط:
- N:1 با کاربران (Users): هر ورودی تاریخچه خرید متعلق به یک کاربر است.
- N:1 با سفارشات (Orders): هر ورودی تاریخچه خرید به یک سفارش مرتبط است.

۱۲. جدول نظرات (Comments Table)

- کلید اصلی: comment_id
- ستون‌ها: comment_id, product_id, user_id, comment, comment_date, status, moderated_by
- توضیح: نظرات کاربران در مورد محصولات را ذخیره می‌کند. وضعیت می‌تواند approved یا inappropriate باشد.
- روابط:
- N:1 با محصولات (Products): هر نظر به یک محصول مرتبط است.
- N:1 با کاربران (Users): هر نظر توسط یک کاربر نوشته می‌شود.
- N:1 با مدیران (Managers): هر نظر توسط یک مدیر مدیریت می‌شود.

۱۳. جدول تخفیفات (Discounts Table)

- کلید اصلی: discount_id
- ستون‌ها: discount_id, name, description, discount_percentage, start_date, end_date
- توضیح: اطلاعات مربوط به تخفیفات و پیشنهادات ویژه را ذخیره می‌کند.
- روابط:
- 1:N با تخفیفات محصولات (ProductDiscounts): یک تخفیف می‌تواند به چندین محصول اعمال شود.

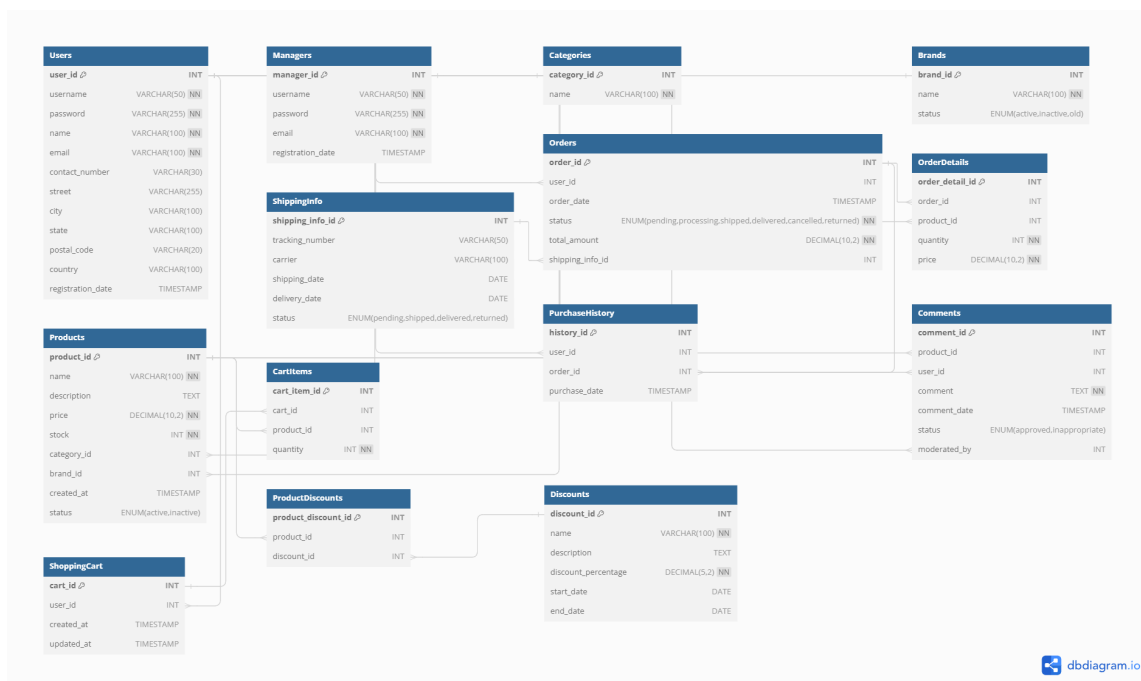


۱۴. جدول تخفیفات محصولات (ProductDiscounts Table)

- کلید اصلی: product_discount_id
- ستون ها: discount_id, product_id, product_discount_id
- توضیح: رابطه بین محصولات و تخفیفات را ذخیره می کند.
- روابط:
- N:1 با محصولات (Products): هر ورودی تخفیف محصول به یک محصول مرتبط است.
- N:1 با تخفیفات (Discounts): هر ورودی تخفیف محصول به یک تخفیف مرتبط است.

۱۵. نمودار ER نهایی

لینک نمودار: ER diagram



شکل ۱: نمودار ER پایگاه داده فروشگاه آنلاین



۲ فاز دوم: نرمال‌سازی

برای اینکه جداول را به نرمال فرم سوم تبدیل کنیم، ابتدا باید به فرم اول نرمال و سپس به فرم دوم نرمال و در انتها به فرم سوم نرمال تبدیل کنیم:

فرم اول نرمال (1NF)

یک جدول برای قرارگرفتن در فرم اول نرمال باید دارای شروط زیر باشد:

- دارای حداقل یک کلید کاندید باشد.
- همه ستون‌های آن غیر قابل تجزیه باشد.
- همه ستون‌های آن اتمیک باشند.

فرم دوم نرمال (2NF)

یک جدول برای قرارگرفتن در فرم دوم نرمال باید دارای شروط زیر باشد:

- جدول باید در فرم اول نرمال باشد.
- تمام ستون‌های غیرکلیدی به کلید اصلی جدول وابسته باشد.

فرم سوم نرمال (3NF)

یک جدول برای قرار گرفتن در فرم سوم نرمال باید دارای شرایط زیر باشد:

- جدول باید در فرم دوم نرمال باشد.
- جدول باید فاقد وابستگی انتقالی باشد.

در ادامه به نرمال‌سازی هر یک از جداول به صورت جدا می‌پردازیم.



۱. جدول کاربران (Users Table)

فرم اول نرمال (1NF)

- اتمیک بودن: هر ستون حاوی مقادیر اتمیک است.
- یگانگی: هر رکورد منحصر به فرد است و با کلید اصلی `user_id` شناسایی می‌شود.

فرم دوم نرمال (2NF)

- کلید اصلی: کلید اصلی `user_id` است.
- وابستگی کامل: همه ستون‌های غیرکلیدی به طور کامل وابسته به کلید اصلی `user_id` هستند.

فرم سوم نرمال (3NF)

- عدم وابستگی انتقالی: هیچ وابستگی انتقالی بین ستون‌های غیرکلیدی وجود ندارد. هر ستون غیرکلیدی مستقیماً وابسته به کلید اصلی `user_id` است.

۲. جدول مدیران (Managers Table)

فرم اول نرمال (1NF)

- اتمیک بودن: هر ستون حاوی مقادیر اتمیک است.
- یگانگی: هر رکورد منحصر به فرد است و با کلید اصلی `manager_id` شناسایی می‌شود.

فرم دوم نرمال (2NF)

- کلید اصلی: کلید اصلی `manager_id` است.
- وابستگی کامل: همه ستون‌های غیرکلیدی به طور کامل وابسته به کلید اصلی `manager_id` هستند.

فرم سوم نرمال (3NF)

- عدم وابستگی انتقالی: هیچ وابستگی انتقالی بین ستون‌های غیرکلیدی وجود ندارد. هر ستون غیرکلیدی مستقیماً وابسته به کلید اصلی `manager_id` است.



۳. جدول دسته‌بندی‌ها (Categories Table)

فرم اول نرمال (1NF)

- اتمیک بودن: هر ستون حاوی مقادیر اتمیک است.
- یگانگی: هر رکورد منحصر به فرد است و با کلید اصلی category_id شناسایی می‌شود.

فرم دوم نرمال (2NF)

- کلید اصلی: کلید اصلی category_id است.
- وابستگی کامل: ستون غیرکلیدی name به طور کامل وابسته به کلید اصلی category_id است.

فرم سوم نرمال (3NF)

- عدم وابستگی انتقالی: هیچ وابستگی انتقالی وجود ندارد. ستون name مستقیماً وابسته به کلید اصلی category_id است.

۴. جدول برندها (Brands Table)

فرم اول نرمال (1NF)

- اتمیک بودن: هر ستون حاوی مقادیر اتمیک است.
- یگانگی: هر رکورد منحصر به فرد است و با کلید اصلی brand_id شناسایی می‌شود.

فرم دوم نرمال (2NF)

- کلید اصلی: کلید اصلی brand_id است.
- وابستگی کامل: همه ستون‌های غیرکلیدی به طور کامل وابسته به کلید اصلی brand_id هستند.

فرم سوم نرمال (3NF)

- عدم وابستگی انتقالی: هیچ وابستگی انتقالی بین ستون‌های غیرکلیدی وجود ندارد. هر ستون غیرکلیدی مستقیماً وابسته به کلید اصلی brand_id است.



۵. جدول محصولات (Products Table)

فرم اول نرمال (1NF)

- اتمیک بودن: هر ستون حاوی مقادیر اتمیک است.
- یگانگی: هر رکورد منحصر به فرد است و با کلید اصلی `product_id` شناسایی می‌شود.

فرم دوم نرمال (2NF)

- کلید اصلی: کلید اصلی `product_id` است.
- وابستگی کامل: همه ستون‌های غیرکلیدی به طور کامل وابسته به کلید اصلی `product_id` هستند.

فرم سوم نرمال (3NF)

- عدم وابستگی انتقالی: هیچ وابستگی انتقالی بین ستون‌های غیرکلیدی وجود ندارد. هر ستون غیرکلیدی مستقیماً وابسته به کلید اصلی `product_id` است.

۶. جدول سفارشات (Orders Table)

فرم اول نرمال (1NF)

- اتمیک بودن: هر ستون حاوی مقادیر اتمیک است.
- یگانگی: هر رکورد منحصر به فرد است و با کلید اصلی `order_id` شناسایی می‌شود.

فرم دوم نرمال (2NF)

- کلید اصلی: کلید اصلی `order_id` است.
- وابستگی کامل: همه ستون‌های غیرکلیدی به طور کامل وابسته به کلید اصلی `order_id` هستند.

فرم سوم نرمال (3NF)

- عدم وابستگی انتقالی: هیچ وابستگی انتقالی بین ستون‌های غیرکلیدی وجود ندارد. هر ستون غیرکلیدی مستقیماً وابسته به کلید اصلی `order_id` است.



۷. جدول اطلاعات حمل و نقل (ShippingInfo Table)

فرم اول نرمال (1NF)

- اتمیک بودن: هر ستون حاوی مقادیر اتمیک است.
- یگانگی: هر رکورد منحصر به فرد است و با کلید اصلی shipping_info_id شناسایی می‌شود.

فرم دوم نرمال (2NF)

- کلید اصلی: کلید اصلی shipping_info_id است.
- وابستگی کامل: همه ستون‌های غیرکلیدی به طور کامل وابسته به کلید اصلی shipping_info_id هستند.

فرم سوم نرمال (3NF)

- عدم وابستگی انتقالی: هیچ وابستگی انتقالی بین ستون‌های غیرکلیدی وجود ندارد. هر ستون غیرکلیدی مستقیماً وابسته به کلید اصلی shipping_info_id است.

۸. جدول جزئیات سفارشات (OrderDetails Table)

فرم اول نرمال (1NF)

- اتمیک بودن: هر ستون حاوی مقادیر اتمیک است.
- یگانگی: هر رکورد منحصر به فرد است و با کلید اصلی order_detail_id شناسایی می‌شود.

فرم دوم نرمال (2NF)

- کلید اصلی: کلید اصلی order_detail_id است.
- وابستگی کامل: همه ستون‌های غیرکلیدی به طور کامل وابسته به کلید اصلی order_detail_id هستند.

فرم سوم نرمال (3NF)

- عدم وابستگی انتقالی: هیچ وابستگی انتقالی بین ستون‌های غیرکلیدی وجود ندارد. هر ستون غیرکلیدی مستقیماً وابسته به کلید اصلی order_detail_id است.



۹. جدول سبد خرید (ShoppingCart Table)

فرم اول نرمال (1NF)

- اتمیک بودن: هر ستون حاوی مقادیر اتمیک است.
- یگانگی: هر رکورد منحصر به فرد است و با کلید اصلی cart_id شناسایی می‌شود.

فرم دوم نرمال (2NF)

- کلید اصلی: کلید اصلی cart_id است.
- وابستگی کامل: همه ستون‌های غیرکلیدی به طور کامل وابسته به کلید اصلی cart_id هستند.

فرم سوم نرمال (3NF)

- عدم وابستگی انتقالی: هیچ وابستگی انتقالی بین ستون‌های غیرکلیدی وجود ندارد. هر ستون غیرکلیدی مستقیماً وابسته به کلید اصلی cart_id است.

۱۰. جدول اقلام سبد خرید (CartItems Table)

فرم اول نرمال (1NF)

- اتمیک بودن: هر ستون حاوی مقادیر اتمیک است.
- یگانگی: هر رکورد منحصر به فرد است و با کلید اصلی cart_item_id شناسایی می‌شود.

فرم دوم نرمال (2NF)

- کلید اصلی: کلید اصلی cart_item_id است.
- وابستگی کامل: همه ستون‌های غیرکلیدی به طور کامل وابسته به کلید اصلی cart_item_id هستند.

فرم سوم نرمال (3NF)

- عدم وابستگی انتقالی: هیچ وابستگی انتقالی بین ستون‌های غیرکلیدی وجود ندارد. هر ستون غیرکلیدی مستقیماً وابسته به کلید اصلی cart_item_id است.



۱۱. جدول تاریخچه خرید (PurchaseHistory Table)

فرم اول نرمال (1NF)

- اتمیک بودن: هر ستون حاوی مقادیر اتمیک است (مثلاً، `purchase_date`, `order_id`, `user_id`).
- یگانگی: هر رکورد منحصر به فرد است و با کلید اصلی `history_id` شناسایی می‌شود.

فرم دوم نرمال (2NF)

- کلید اصلی: کلید اصلی `history_id` است.
- وابستگی کامل: همه ستون‌های غیرکلیدی (`purchase_date`, `order_id`, `user_id`) به طور کامل وابسته به کلید اصلی `history_id` هستند.

فرم سوم نرمال (3NF)

- عدم وابستگی انتقالی: هیچ وابستگی انتقالی بین ستون‌های غیرکلیدی وجود ندارد. هر ستون غیرکلیدی مستقیماً وابسته به کلید اصلی `history_id` است.

۱۲. جدول نظرات (Comments Table)

فرم اول نرمال (1NF)

- اتمیک بودن: هر ستون حاوی مقادیر اتمیک است.
- یگانگی: هر رکورد منحصر به فرد است و با کلید اصلی `comment_id` شناسایی می‌شود.

فرم دوم نرمال (2NF)

- کلید اصلی: کلید اصلی `comment_id` است.
- وابستگی کامل: همه ستون‌های غیرکلیدی به طور کامل وابسته به کلید اصلی `comment_id` هستند.

فرم سوم نرمال (3NF)

- عدم وابستگی انتقالی: هیچ وابستگی انتقالی بین ستون‌های غیرکلیدی وجود ندارد. هر ستون غیرکلیدی مستقیماً وابسته به کلید اصلی `comment_id` است.



۱۳. جدول تخفیف‌ها (Discounts Table)

فرم اول نرمال (1NF)

- اتمیک بودن: هر ستون حاوی مقادیر اتمیک است.
- یگانگی: هر رکورد منحصر به فرد است و با کلید اصلی discount_id شناسایی می‌شود.

فرم دوم نرمال (2NF)

- کلید اصلی: کلید اصلی discount_id است.
- وابستگی کامل: همه ستون‌های غیرکلیدی به طور کامل وابسته به کلید اصلی discount_id هستند.

فرم سوم نرمال (3NF)

- عدم وابستگی انتقالی: هیچ وابستگی انتقالی بین ستون‌های غیرکلیدی وجود ندارد. هر ستون غیرکلیدی مستقیماً وابسته به کلید اصلی discount_id است.

۱۴. جدول تخفیف‌های محصولات (ProductDiscounts Table)

فرم اول نرمال (1NF)

- اتمیک بودن: هر ستون حاوی مقادیر اتمیک است.
- یگانگی: هر رکورد منحصر به فرد است و با کلید اصلی product_discount_id شناسایی می‌شود.

فرم دوم نرمال (2NF)

- کلید اصلی: کلید اصلی product_discount_id است.
- وابستگی کامل: همه ستون‌های غیرکلیدی به طور کامل وابسته به کلید اصلی product_discount_id هستند.

فرم سوم نرمال (3NF)

- عدم وابستگی انتقالی: هیچ وابستگی انتقالی بین ستون‌های غیرکلیدی وجود ندارد. هر ستون غیرکلیدی مستقیماً وابسته به کلید اصلی product_discount_id است.

با توجه به توضیحات ارائه شده، همگی جداول تا سطح سوم نرمال می‌باشند.



۳ فاز سوم: ایجاد جداول و اضافه کردن دیتا

برای اضافه کردن داده به جداول از اسکریپتی استفاده شده است که در ادامه هر تابع آن توضیح داده شده است. این اسکریپت از کتابخانه Faker برای تولید داده‌های واقع‌گرایانه و از کتابخانه mysql.connector برای ارتباط با یک پایگاه داده MySQL استفاده می‌کند.

۱. جدول کاربران (Users Table)

- تولید نام کاربری، رمز عبور، نام، ایمیل، شماره تماس و آدرس با استفاده از کتابخانه Faker.

۲. جدول مدیران (Managers Table)

- تولید نام کاربری، رمز عبور و ایمیل با استفاده از کتابخانه Faker.

۳. جدول دسته‌بندی‌ها (Categories Table)

- وارد کردن دسته‌بندی‌های از پیش تعریف شده مانند الکترونیک، پوشاک و غیره به جدول.

۴. جدول برندها (Brands Table)

- وارد کردن برندهای از پیش تعریف شده برای هر دسته با وضعیت تصادفی (فعال، غیرفعال، قدیمی).

۵. جدول محصولات (Products Table)

- واکنشی^۱ شناسه‌های دسته‌بندی و برند.
- تولید نام محصول و توضیحات با استفاده از داده‌های از پیش تعریف شده.
- تولید قیمت، مقادیر موجودی و وضعیت به صورت تصادفی.

۶. جدول اطلاعات حمل و نقل (ShippingInfo Table)

- تولید شماره پیگیری، نام حامل، تاریخ حمل و تاریخ تحویل و وضعیت با استفاده از کتابخانه Faker.

۷. جدول سفارشات (Orders Table)

- استفاده از شناسه‌های کاربر و اطلاعات حمل و نقل تصادفی.
- تولید وضعیت و مقادیر کل به صورت تصادفی.

^۱ Fetch



۸. جدول جزئیات سفارشات (OrderDetails Table)

- استفاده از شناسه‌های سفارش و محصول تصادفی.
- تولید مقادیر و قیمت‌ها به صورت تصادفی.

۹. جدول سبد خرید (ShoppingCart Table)

- تولید سبدهای خرید برای کاربران به صورت تصادفی.

۱۰. جدول اقلام سبد خرید (CartItem Table)

- استفاده از شناسه‌های سبد و محصول تصادفی.
- تولید مقادیر به صورت تصادفی.

۱۱. جدول تاریخچه خرید (PurchaseHistory Table)

- استفاده از شناسه‌های کاربر و سفارش تصادفی.

۱۲. جدول نظرات (Comments Table)

- استفاده از شناسه‌های محصول و کاربر تصادفی.
- تولید نظرات با محتوای تصادفی.
- تعیین وضعیت (تأیید شده یا نامناسب) و مدیریت توسط شناسه‌های مدیر تصادفی.

۱۳. جدول تخفیف‌ها (Discounts Table)

- تولید نام‌ها و توضیحات تخفیف بر اساس تعطیلات از پیش تعریف شده.
- تعیین درصد‌های تخفیف، تاریخ‌های شروع و پایان به صورت تصادفی.

۱۴. جدول تخفیف‌های محصولات (ProductDiscounts Table)

- استفاده از شناسه‌های محصول و تخفیف تصادفی.



۴ فاز چهارم: طراحی پرس و جو

پرس و جو ۱: نام کلیه مشتریانی که خریدی در یک ماه گذشته داشته‌اند

توضیح: این پرس و جو نام، ایمیل و شماره تماس تمام کاربرانی را که حداقل یک خرید در یک ماه گذشته داشته‌اند، بازایی می‌کند. این پرس و جو با استفاده از جدول Orders، کاربرانی را که سفارشات در یک ماه گذشته ثبت کرده‌اند، فیلتر می‌کند.

```
SELECT name, email, contact_number
FROM Users
WHERE user_id IN (
    SELECT DISTINCT user_id
    FROM Orders
    WHERE order_date >= DATE_SUB(CURDATE(), INTERVAL 1 MONTH)
);
```

پرس و جو ۲: مقدار کل فروش هر دسته‌بندی محصول در یک ماه گذشته

توضیح: این پرس و جو مقدار کل فروش را برای هر دسته‌بندی محصول در یک ماه گذشته محاسبه می‌کند. این پرس و جو با اتصال جداول OrderDetails، Products، Categories و Orders، فروش را برای هر دسته‌بندی جمع می‌کند و نتایج را بر اساس نام دسته‌بندی گروه‌بندی می‌کند.

```
SELECT c.name AS category, SUM(od.price * od.quantity) AS
    total_sales
FROM OrderDetails od
JOIN Products p ON od.product_id = p.product_id
JOIN Categories c ON p.category_id = c.category_id
JOIN Orders o ON od.order_id = o.order_id
WHERE o.order_date >= DATE_SUB(CURDATE(), INTERVAL 1 MONTH)
GROUP BY c.name;
```



پرس و جو ۳: جزئیات سفارشات معلق

توضیح: این پرس و جو جزئیات سفارشات را که هنوز در حالت معلق هستند، بازیابی می‌کند. این پرس و جو شناسه سفارش، نام کاربر، تاریخ سفارش و مبلغ کل سفارشات را از جداول Orders و Users انتخاب می‌کند که وضعیت آنها pending است.

```
SELECT o.order_id, u.name AS user_name, o.order_date,
       o.total_amount
FROM Orders o
JOIN Users u ON o.user_id = u.user_id
WHERE o.status = 'pending';
```

پرس و جو ۴: لیست محصولات با موجودی کم

توضیح: این پرس و جو محصولاتی را که موجودی کمی دارند (کمتر از ۱۰ آیتم) شناسایی می‌کند. این پرس و جو نام محصول و مقدار موجودی را از جدول Products انتخاب می‌کند که موجودی آنها کمتر از مقدار مشخص شده است.

```
SELECT name, stock
FROM Products
WHERE stock < %s;
```

پرس و جو ۵: کاربران با هزینه کل بیشتر از یک میلیون

توضیح: این پرس و جو کاربرانی را که مجموع هزینه‌های آنها بیش از یک میلیون است شناسایی می‌کند. این پرس و جو نام، ایمیل و مبلغ کل هزینه شده را از جداول Orders و Users انتخاب می‌کند، نتایج را بر اساس شناسه کاربر گروه‌بندی می‌کند و کاربرانی را فیلتر می‌کند که مجموع هزینه‌های آنها بیش از یک میلیون است.

```
SELECT u.name, u.email, SUM(o.total_amount) AS total_spent
FROM Orders o
JOIN Users u ON o.user_id = u.user_id
GROUP BY u.user_id
HAVING total_spent > %s;
```

**پرس و جو ۶: اضافه کردن تخفیف به لیست تخفیف‌ها**

توضیح: این پرس و جو به مدیر اجازه می‌دهد یک تخفیف جدید به لیست تخفیف‌ها اضافه کند. این پرس و جو یک رکورد جدید را با جزئیات تخفیف به جدول Discounts وارد می‌کند.

```
INSERT INTO Discounts (name, description, discount_percentage,
                        start_date, end_date)
VALUES (%s, %s, %s, %s, %s);
```

پرس و جو ۷: برچسب زدن یک نظر به عنوان نامناسب

توضیح: این پرس و جو به مدیر اجازه می‌دهد یک نظر کاربر را به عنوان نامناسب برچسب بزند. این پرس و جو وضعیت نظر مشخص شده را در جدول Comments به inappropriate به‌روزرسانی کرده و شناسه مدیر را که این تغییر را انجام داده است، ثبت می‌کند.

```
UPDATE Comments
SET status = 'inappropriate', moderated_by = %s
WHERE comment_id = %s;
```

پرس و جو ۸: محصولات پرفروش در ماه گذشته

توضیح: این پرس و جو ۵ محصول پرفروش را در ماه گذشته شناسایی می‌کند. این پرس و جو با اتصال جداول OrderDetails، Orders و Products، فروش را برای هر محصول جمع می‌کند و نتایج را بر اساس نام محصول گروه‌بندی کرده و بر اساس فروش کل به ترتیب نزولی مرتب می‌کند و به ۵ محصول برتر محدود می‌کند.

```
SELECT p.name AS product, SUM(od.price * od.quantity) AS
      total_sales
FROM OrderDetails od
JOIN Products p ON od.product_id = p.product_id
JOIN Orders o ON od.order_id = o.order_id
WHERE o.order_date >= DATE_SUB(CURDATE(), INTERVAL 1 MONTH)
GROUP BY p.name
ORDER BY total_sales DESC
LIMIT %s;
```



پرس و جو ۹: لیست برندهای غیرفعال و تعداد محصولات آنها

توضیح: این پرس و جو برندهای غیرفعال و تعداد محصولات مربوط به هر یک را بازایی می‌کند. این پرس و جو جداول Brands و Products را اتصال داده، نتایج را بر اساس نام برند گروه‌بندی کرده و تعداد محصولات را برای هر برند غیرفعال شمارش می‌کند.

```
SELECT b.name AS brand, COUNT(p.product_id) AS product_count
FROM Brands b
JOIN Products p ON b.brand_id = p.brand_id
WHERE b.status = 'inactive'
GROUP BY b.name;
```

پرس و جو ۱۰: مقدار کل تخفیف اعمال شده برای هر محصول در ماه گذشته

توضیح: این پرس و جو مقدار کل تخفیف اعمال شده برای هر محصول در ماه گذشته را محاسبه می‌کند. این پرس و جو با اتصال جداول Products, Discounts, ProductDiscounts, OrderDetails و Orders، تخفیف‌ها را برای هر محصول جمع می‌کند و نتایج را بر اساس نام محصول گروه‌بندی می‌کند.

```
SELECT p.name AS product, SUM(d.discount_percentage * od.price
    * od.quantity / 100) AS total_discount
FROM ProductDiscounts pd
JOIN Discounts d ON pd.discount_id = d.discount_id
JOIN Products p ON pd.product_id = p.product_id
JOIN OrderDetails od ON p.product_id = od.product_id
JOIN Orders o ON od.order_id = o.order_id
WHERE o.order_date >= DATE_SUB(CURDATE(), INTERVAL 1 MONTH)
GROUP BY p.name;
```