# Oracle database design for e-commerce application

Anna Lihvoinen

Haaga-Helia University of Applied Sciences

Business Information Technology Programme

August, 2009

# Abstract

Published in September, 2009 at Haaga-Helia University of Applied Sciences

The purpose of this thesis is to design a database for e-commerce application which will be further implemented in Oracle Application Express (Apex) by Database Software Horizons.

The design document includes ER diagrams, table descriptions, table source code, and testing results.

Logical and physical database designs for relational modeling methods are applied in this work. The result of the work is documented according to Unified Modeling language notation and implementation is done in Oracle 10g Express Edition database.

The designed database is fully applicable for building an e-commerce application in Apex or any other programming environment.

Keywords: Oracle, database design, e-commerce application

# Table of contents


Abstract ..... 1

Table of contents ..... 2

1 Introduction ..... 1

1.1 About company ..... 1

1.2 Why e-commerce application? ..... 1

1.3 Objectives of thesis work ..... 3

1.4 Tools ..... 3

2 System requirements ..... 4

2.1 Application Usage ..... 4

2.2 Application Users ..... 4

2.3 Product and Product Options ..... 5

2.4 Sales Order and Ordering Process ..... 6

2.5 Shipping and Shipping Process ..... 7

2.6 Payment Systems ..... 7

2.7 Tax System ..... 8

2.8 Configuration of the web-store ..... 8

2.9 Currency ..... 8

2.10 Messages ..... 8

2.11 Newsletters ..... 9

2.12 News ..... 9

2.13 Software requirements ..... 9

3 Logical database design ..... 10

3.1 ER Diagrams ..... 11

3.1.1 Diagram 1 'Main process' ..... 11

3.1.2 Diagram 2. "Product group" ..... 12

3.1.3 Diagram 3 'Sales Order group' ..... 13

3.1.4 Diagram 4 'Shipping group' ..... 14

3.1.5 Diagram 5 'Line Item group' ..... 15

3.1.6 Diagram 6 'Payment group' ..... 16

3.1.7 Diagram 7 'Configuration and additional tables' ..... 17

3.2 Tables 'description ..... 18

3.2.1 Main tables (Diagram 1) ..... 18
CUSTOMER_DETAILS ..... 18
LINE_ITEM ..... 19
PRODUCT ..... 20
SALES_ORDER ..... 21
3.2.2 Product group (see Diagram 2) ..... 22
CATEGORY ..... 22
IMAGE ..... 23
FAVORITES ..... 24
FILE ..... 25
MANUFACTURER ..... 26
PRODUCT_OPTION ..... 27
PRODUCT_TO_PR_OPTION ..... 28
REVIEW ..... 29
TAX_CLASS ..... 30
TAX_RATE ..... 31
3.2.3 Sales Order Group (see Diagram 3) ..... 32
ADDRESS_FORMAT ..... 32
BILLING_DETAILS ..... 33
ORDER_STATUS ..... 34
SHOPPING_CART ..... 35
SHIPPING_DETAILS ..... 36
STATES ..... 37
3.2.4 Shipping Group (see Diagram 4) ..... 38
ESTIMATED_DELIVERY_TIME ..... 38
GEO_ZONE ..... 39
SHIPPING_AMOUNT_RATE ..... 40
SHIPPING_PRICE_TYPE ..... 41
SHIPPING_TYPE ..... 42
ZONE ..... 43
3.2.5 Line Items Group (see Diagram 5) ..... 44
LINE_DOWNLOAD ..... 44
LINE_ITEM_OPTION ..... 45

3.2.6 Payment Group (see Diagram 6) ..... 46
PAYMENT_CC_TYPE ..... 46
PAYMENT_MODE_EX ..... 47
PAYMENT_MODE_IN ..... 48
THESIS_SO_PAY_DETAILS ..... 49
3.2.7 Additional tables (Diagram 7) ..... 50
CONFIGURATION ..... 50
CONFIGURATION_GROUP ..... 51
CURRENCY ..... 52
CURRENCY ..... 52
MESSAGE ..... 53
NEWS ..... 54
NEWSLETTER ..... 55
4 Physical model ..... 56
5 Test Cases and Results ..... 57
Summary ..... 60
Bibliography ..... 61
Appendix A. Create User and Tables source codes ..... 62
Create tables ..... 62
Appendix B. Database Views Source Code ..... 84

# 1 Introduction

## 1.1 About company

"Database Software Horizons" is a privately registered company in Finland specializing in systems design and programming leveraging Oracle and Web technologies. One of the projects is e-commerce application which may be suitable for wide range of customers, mainly small and meddle-sized companies. The application is going to be implemented in Oracle Apex technology which is free. The application objectives are to meet all modern e-commerce requirements but at the same time not to be redundant of unnecessary features, be easily customizable and intuitively clear for end-users.

## 1.2 Why e-commerce application?

Small and medium enterprises (SME's) are increasingly using the Internet to improve efficiency and productivity. Firms move more and more elements of their value chain i.e. their supply networks and sales channels on-line. Direct sales and distribution is expected to play more important role in business practice, particularly where the service involves the supply of information or information surrounding exchange of goods. [Smyk, p. 8]

In order to sell products through the Internet, it is necessary for the company to think about the means needed to bring its products, services, or information to the customers. If the company offers only a few products and has very low order volume, there is no need for a complex shopping system. Once the company begins to offer a wide range of articles, the system becomes difficult for both the shop owner and customers to handle. The shop owner will have difficulties keeping the Web pages up-to-date and consistent, and the customers will have trouble finding a certain products fast. Therefore, a shopping solution is required to handle the increased flow of information that is the basis for the online transaction. Shopping solution software should be easy for the customers to use, for example, it should save the preferences and personal data of the customers and finding of a certain product can be done either by browsing or by searching [Amor, p.276].

The modern online shopping solutions should include at least the following features:

- Database – Product information needs to be stored in a database, separated from the layout.
- Interface to applications – The shopping solutions need to provide interfaces to other applications, such as a payment processor and the ordering system.
- Payment – The shop should support several payment models, for supporting different business models and users preferences.
- Reporting – Thought reports is should be possible to determine what customers really want.
- Search engines – Customer should fine a particular item with one mouse click.
- Shopping basket – The customer's tool for collection the products the want to order.
- Terms and conditions – In order to make contracts legal, it is necessary to display the terms and conditions.
- Web design templates – Use of templates to simplify the design process.

Aside from the business requirements the shopping solution should provide the technical requirements are very important, such as, for instance, application server characteristics (ability to support Internet standards, having a sound foundation and so on). Another very important issue linked closely to the application server platform is the readiness for integration with databases, ERP systems, payment providers, and other system and processes, regardless of who owns them and what operating system and platform they are based on. Significant issue is the ability of shopping solution allows the replacement of parts of the application with some 'in-house' programming of if it is needs to be done by the software vendor. To make things more practical, the user interface should be easily modifiable to allow the marketing department and the graphical artists to change the visuals whenever they need to [Amor, p. 284-286].

## 1.3 Objectives of thesis work

The purposes of this work are to collect and analyze the requirements for modern e-commerce application and design a database based on these. The result of logical design is presented in ER diagrams and tables 'description (tables, attributes, primary and foreign keys, indexes). The database is implemented and tested in Oracle10g Express Edition environment using SQL Developer tool.

Development of application graphical user interface is not included in this thesis work.

## 1.4 Tools

To reach the project goals the following free Oracle tools were used: SQL Developer (version 1.5.0.54.40) and SQL Developer Data Modeler (version 2.0.0.57.0).

SQL Developer is a free Oracle graphical tool for a database to be developed, browsed and maintained. It can be connected to any Oracle database version 9.2.0.1 and later and run on Windows, Linux and Mac operating systems. The thesis database was implemented using SQL Developer, testing SQL queries were also built and run in it. SQL Developer is integrated with Oracle Apex (Application Express Edition) which made it very useful in further application development as well.

SQL Developer Data Modeler is a tool for data and database modeling, including, for instance, Entity Relationship Diagrams (ERD), Relational (database design), forward and reverse engineering and DDL code generation. It is platform independent and connects to any Oracle database.

The thesis database was implemented in Oracle 10 g Express Edition database which is based on Oracle 10g database and free for download, develop and distribute. All Oracle database features are available in Oracle 10 g Express Edition which makes it perfect for studying and developing purposes.

# 2 System requirements

## 2.1 Application Usage

The main idea of any e-commerce application is that the seller places the product catalog in the Internet and the buyers chose the items from it and order them.

The system must be suitable for a small to middle-sized companies which are interested in e-commerce business. The system must meet the general e-commerce application requirements and at the same time can be easily customizable for a particular customer.

## 2.2 Application Users

The application users are web-store customers (buyers) and administrators. Web-store customers are current or potential buyers of products. They are divided into registered and non-registered users.
Non-registered visitors (users) can search the items from the catalog but when they want to place the order the system asks to fulfill his/her personal, billing and delivery information. Once they do that they become registered users and are given a password and a user name for the system.

Registered customers use their credentials to login into the web shop application and not any more needed to enter their personal data. The personal information such as personal, shipping, billing data and etc. can be maintained by the customer. Only user name can not be changed.

The web-store customer can perform the following operations:

- search the products;
- choose the products and make the order;
- pay the order;
- follow the order status;
- place products into list of favorites;
- write a review on product he/she has bought;

- maintain his/her personal data such as, for instance, shipping and billing addresses, contact information and so on;
- subscribe to web-store newsletters.

The system administrator (administrators) has more privileges such as the following:

- adding and maintaining the products in the catalog and product categories;
- changing the products options;
- setting the shipping and payments options;
- changing the order's statuses;
- getting product, order, delivery and payment reports.

## 2.3 Product and Product Options

The product is an item which is sold in electronic store. The product can be a stackable item (for instance food, cloth, tools and etc.) or not-stockable item such as software or music. The stockable items may have the physical characteristics – weight, size, color and etc. The non-stockable items may have file size as an option. All items must to be given a name. They can also have a description (short and long), image (images), price and tax rate and other characteristics. Products can also be marked as hot products (the most popular), promotion products or placed into favorites products related to a particular customer.

In order to simplify the searching of products they may be grouped into categories and sub-categories. Products also may be searched by manufacture (vendor).

In order to make a web shop more attractive to customers the product profile may be provided with images (one or many) and customers review.

Non-stockable products such as music or software can be downloaded after the order is paid. Downloading must have time and click limitations.

## 2.4 Sales Order and Ordering Process

"In business or commerce, an order is a stated intention, either spoken or written, to engage in a commercial transaction for specific products or services. From a buyer's point of view it expresses the intention to buy and is called a purchase order. From a seller's point of view it expresses the intention to sell and is referred to as a sales order." (Wikipedia, Order).

Products chosen by the customer can be preceded to the sales order. Ordering items should be provided with the following information:

- product name;
- short description;
- quantity;
- tax rate;
- price.

Customer choose the delivery type and delivery destination and will be provided with estimated delivery price and shipping conditions.

Non-registered customer is asked to give the personal, billing and delivery information first. Billing information is always related to the particular registered customer but the shipping information can be changed by the customer. By default, it is customer's personal information (name and address).

Sales order have a range of statuses telling about different order processing stages such, for instance, invoiced, delivered and etc. Order statuses can be changed by the application administrator.

Sales order must have the following information:

- order number (given by the system);
- line items;
- items quantity;
- customer details,
- shipping details;

- billing details;
- created date;
- delivery date (estimated and actual);
- order status;
- delivery type;
- order amount (with and without freight charge).

## 2.5 Shipping and Shipping Process

Shipping is a physical process of transporting goods from sellers stock to customer.
During the order processing the delivering information must be given to the buyer depending on order amount, quantity, shipping address, and etc. The following shipping options must be assigned to the order:

- shipping type (for instance; domestic economy, international express, pick-up, air service and etc);
- estimated delivery time (based on shipping type chosen);
- shipping price (based on shipping type, destination, total weight or sales order amount).

The owner of web-store may set a fixed shipping price for any sales order or make it depending on, for instance, sales order amount, order quantity or total weight. It is also possible to calculate the delivery price based on certain variables such as delivery type, destination, total weight, total amount or even delivery time.

## 2.6 Payment Systems

There are a lot of payment possibilities available nowadays such as cash, bank transferring, credit card payments, cheque and others. An owner of web-store may buy an external payment service for providing the customer payments such as, for instance, Paypal or may want to processed payments by itself.

Sales order payment information must be fulfilled with the following data:

- credit card type (if paid by credit card);
- credit card number, expiry date and card verification code;
- check number and date (if paid by cheque);
- demand draft number (if paid by demand draft);
- transaction number (in case of external payment, ex. Paypal).

External payment systems need to be integrated into the web-shop application.

## 2.7 Tax System

Web-shop application must be configurable for any VAT system. The product type and shipping (in some cases billing) address will affect item tax amount, that's why the system must have a flexible algorithm for calculating tax.

## 2.8 Configuration of the web-store

Web shop admin (admins) must have a possibility to configure different options in order to make the application more personal and meeting the needs of the web shop owner. For instance, the admin may want to configure product options, image parameters, set different payment and shipping options to be available or not, to modify the application outlook, and so on. All administrative tasks are done in Web-based graphical user interface.

## 2.9 Currency

Web shop must support all currencies and must have a possibility to convert prices into any currency.

## 2.10 Messages

Web-store administration may want to communicate with customers via e-mail (sending the advertisings, answering the questions and so on). All messages needed to be recorded in a database.

### 2.11 Newsletters

Web-store administration may want to send newsletters to customers and keep them recorded in a database.

### 2.12 News

Web-store administrations may want to add news about product or services into web shop.

### 2.13 Software requirements

Web-store application is built and runs on Oracle Express Edition database (Apex 2.1 is included in Oracle Express Edition database package and it can be upgraded to the latest version of Apex) that is free. The installation can be performed by installing Oracle Express Edition database and then importing Web shop application in Apex Web-based Integrated Development Environment.

# 3 Logical database design

Logical database design was made in Oracle SQL Developer Data Modeler and consists of Entity Relationships diagrams and table description.

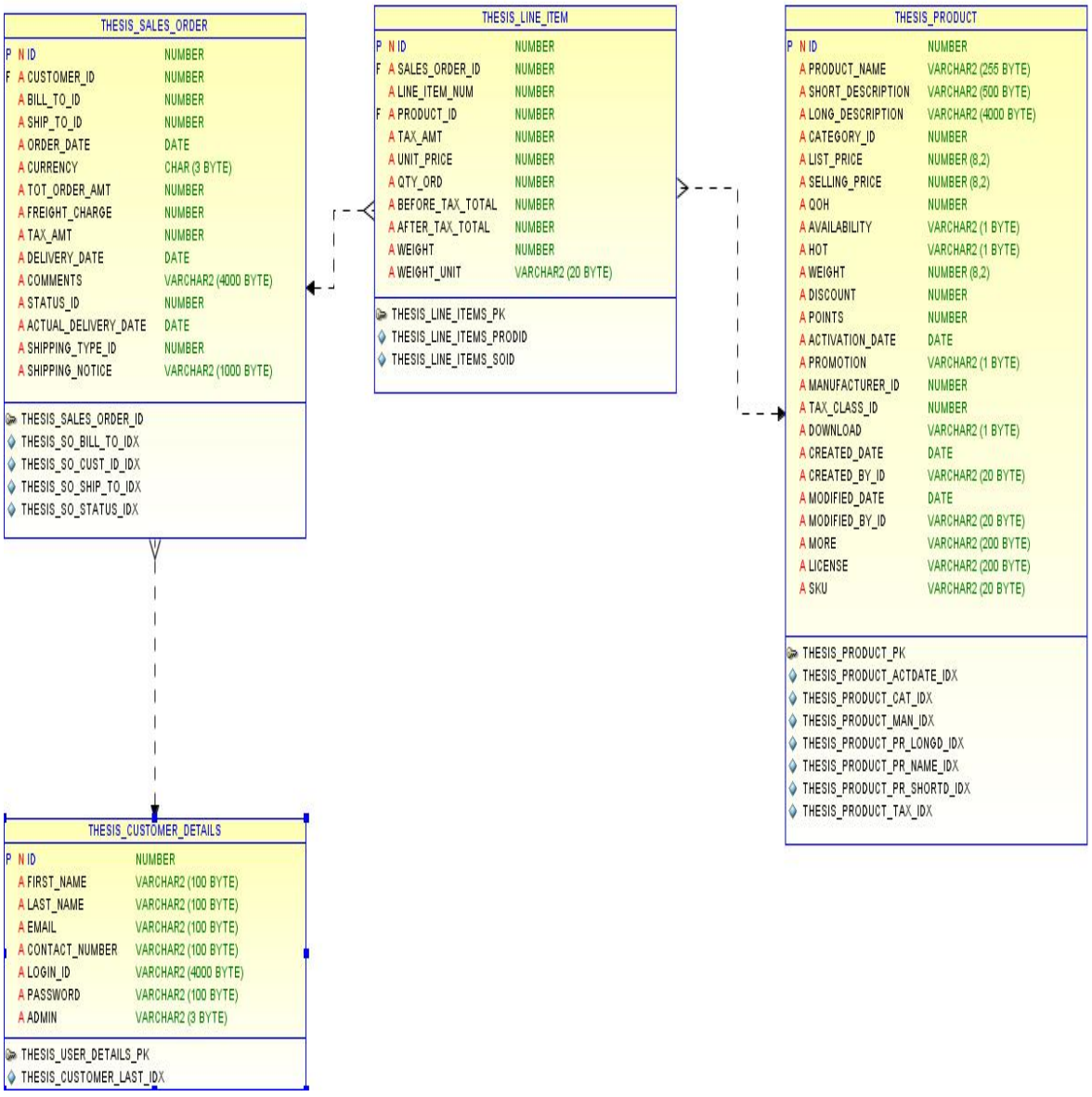
In order to make the diagrams more clear entities were divided into groups relating to main business objects such as in this case Product, Sales order, Shipping (delivery), and Billing (invoicing). The first diagram shows the main business idea of an e-commerce application.

Table descriptions include the following information about database tables:

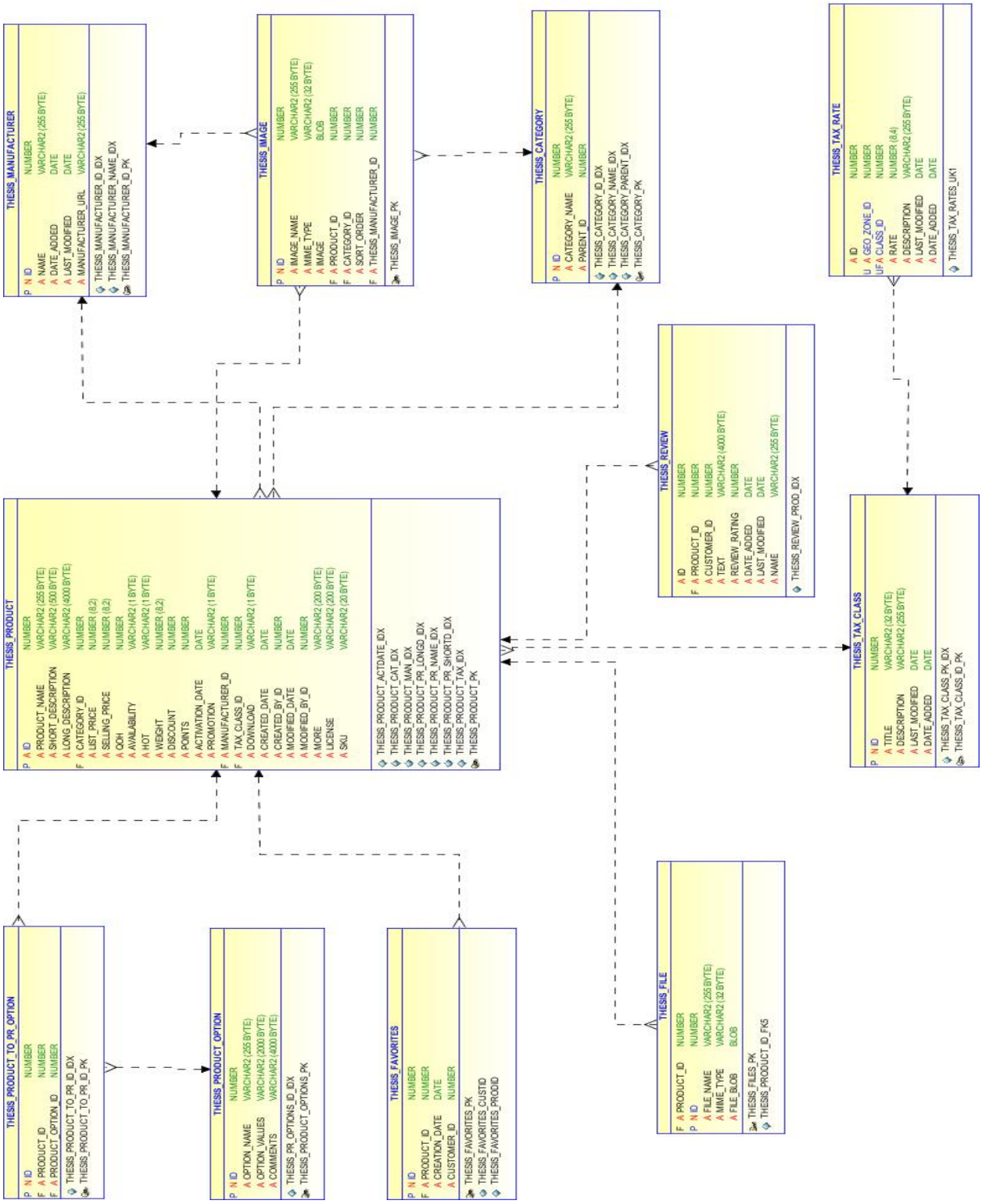
- Entity name
- Meaning of table
- Attributes (primary key and foreign keys are listed first)
- Attributes meaning
- Attributes type and length
- Indexes
- References to other tables

## 3.1 ER Diagrams

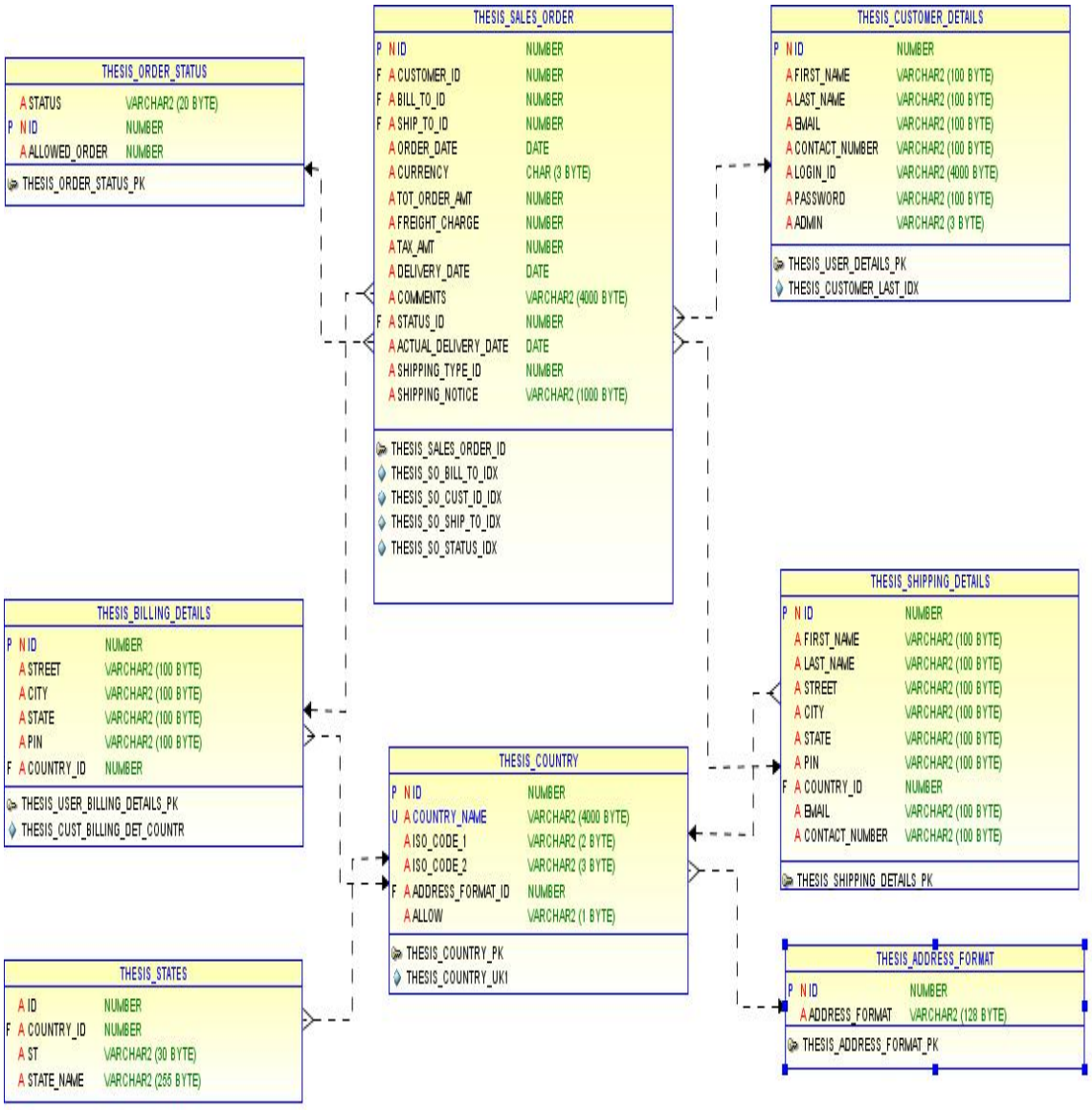
### 3.1.1 Diagram 1 "Main process"



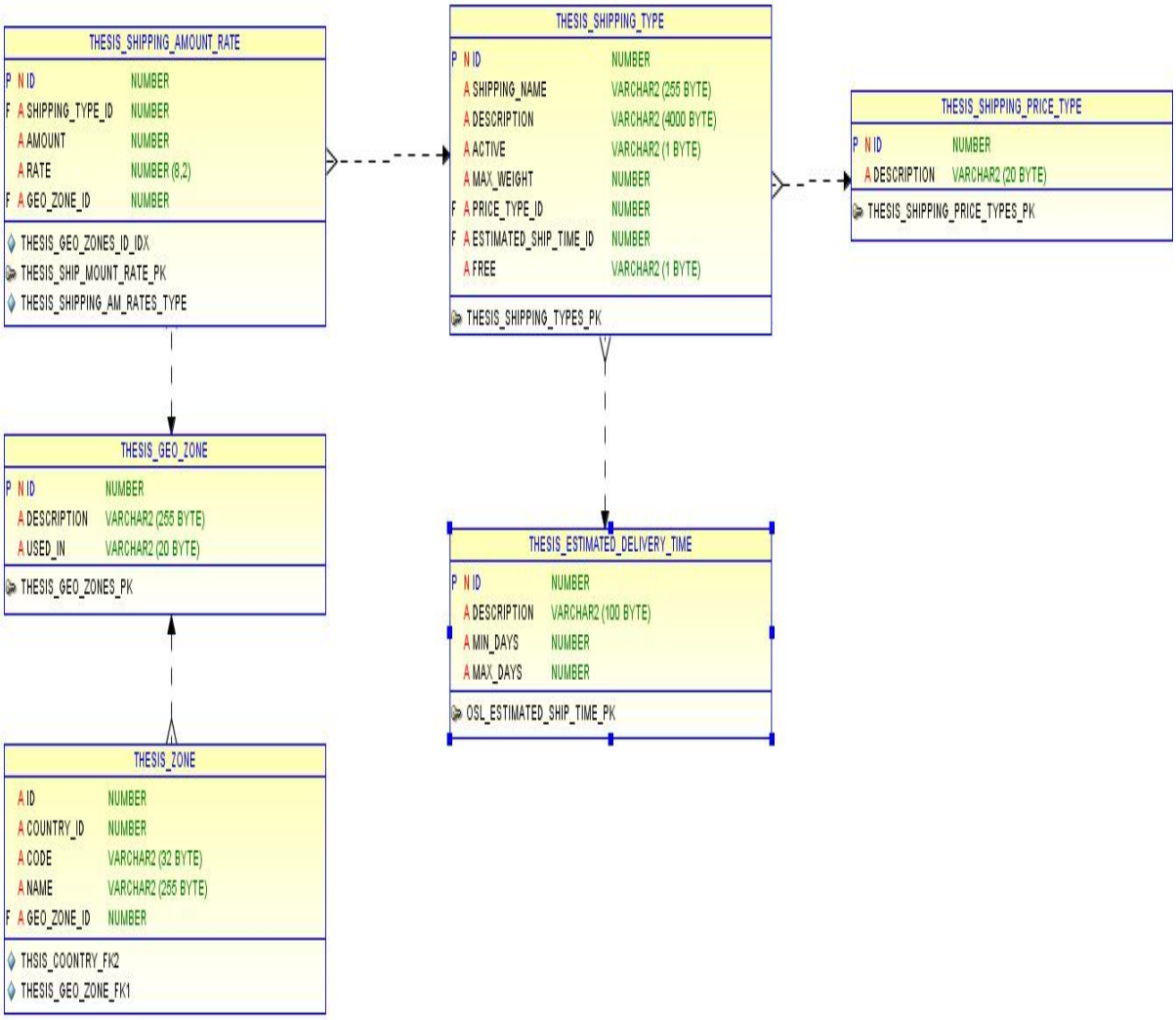
### 3.1.2 Diagram 2. “Product group ”



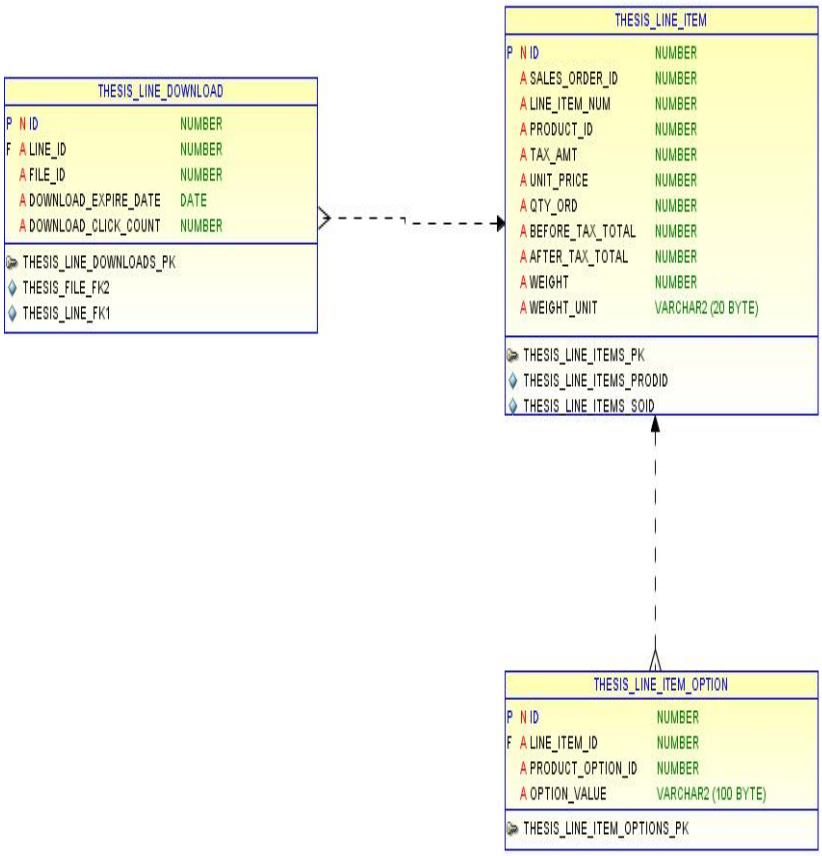
### 3.1.3 Diagram 3 “Sales Order group”



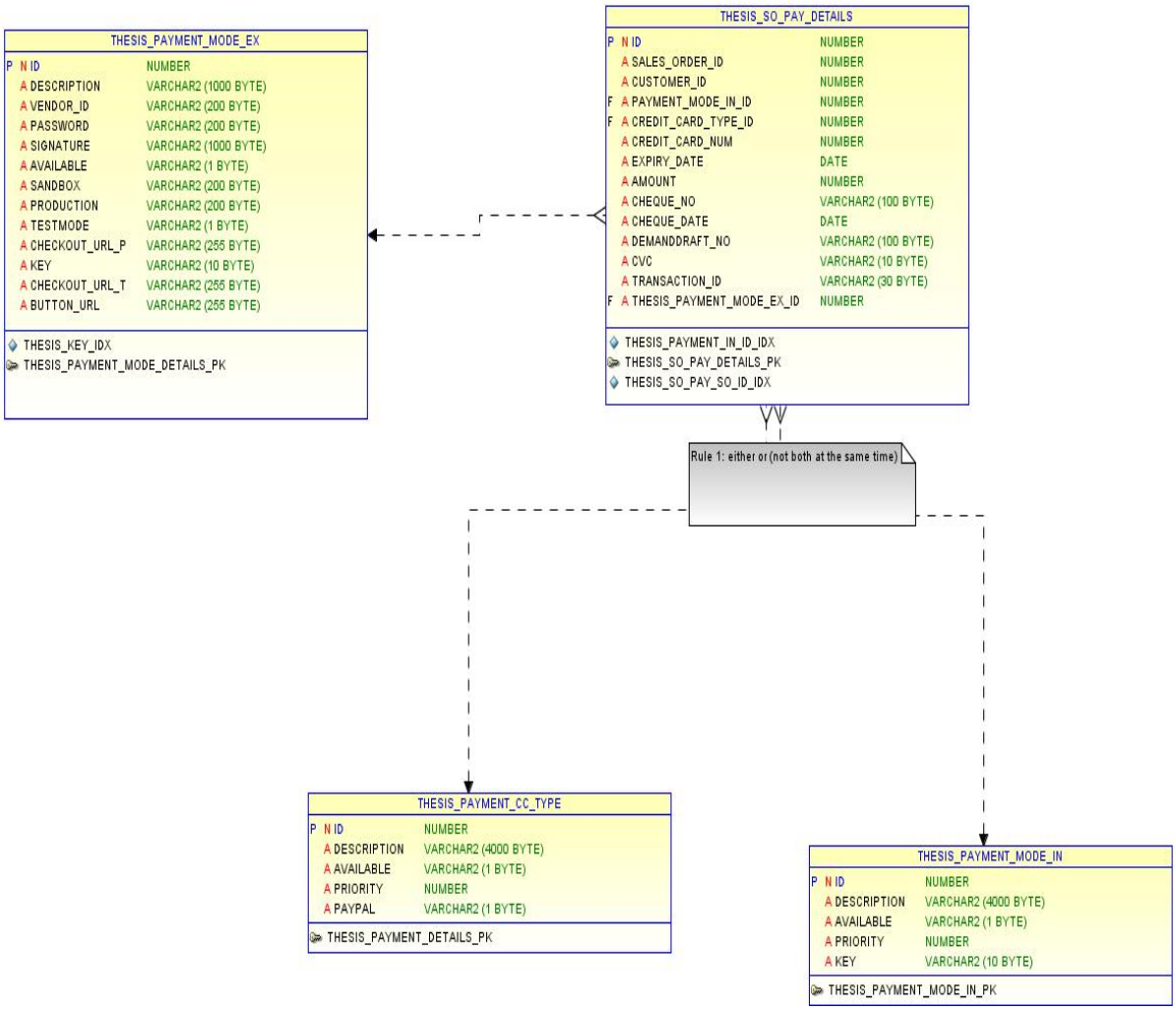
### 3.1.4 Diagram 4 “Shipping group ”



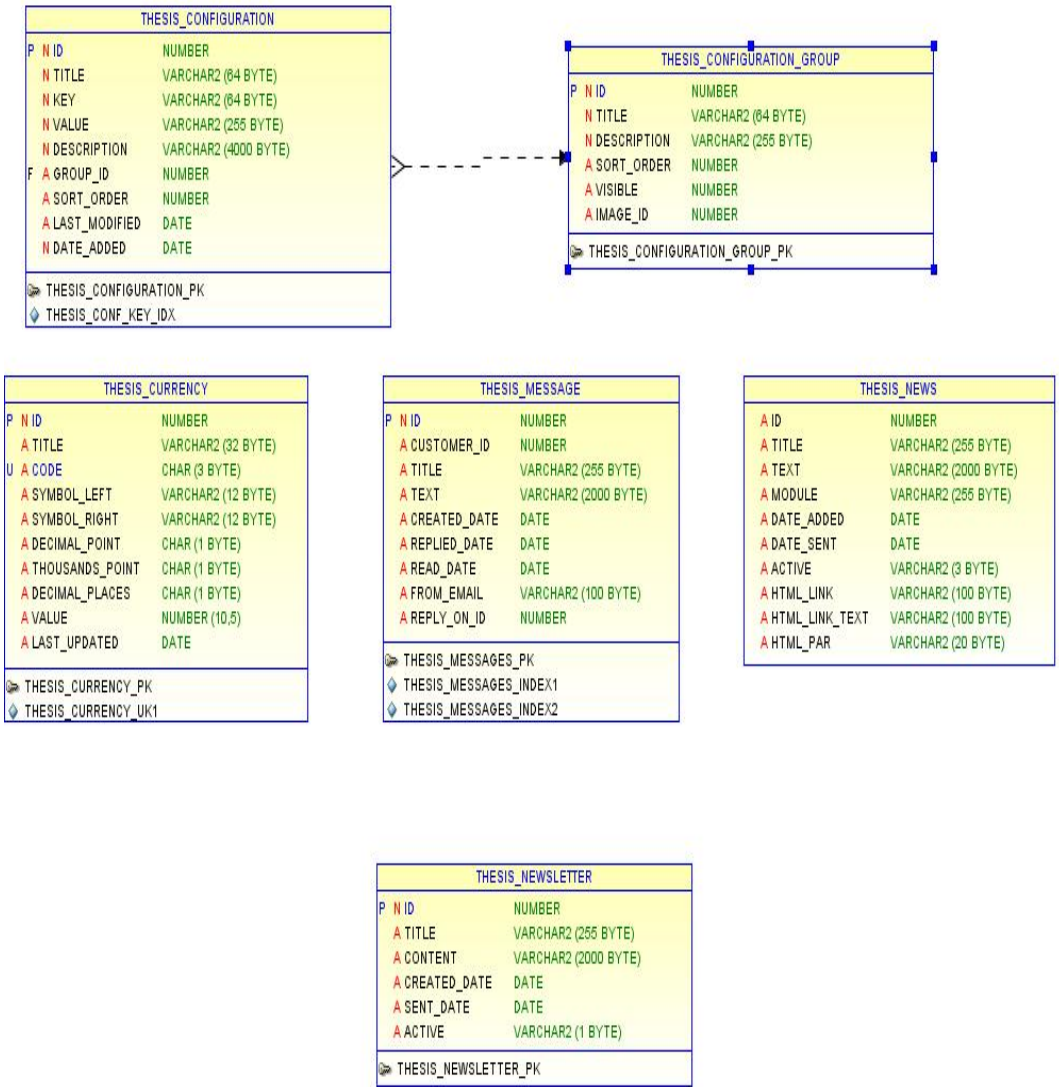
### 3.1.5 Diagram 5 “Line Item group ”



### 3.1.6 Diagram 6 “Payment group ”



### 3.1.7 Diagram 7 "Configuration and additional tables"



## 3.2 Tables ’description

### 3.2.1 Main tables (Diagram 1)

#### CUSTOMER_DETAILS

Contains customer information

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| FIRST_NAME | Customer ’s first name | Varchar2 (100) |
| LAST_NAME | Customer ’s last name | Varchar2 (100) |
| EMAIL | Customer ’s email address | Varchar2 (100) |
| CONTACT_NUMBER | A phone number of a customer | Varchar2 (100) |
| LOGIN_ID | Customer ’s login ID, FK, unique | Number |
| PASSWORD | Password for login to the system | Varchar2 (4000) |
| ADMIN | Admin flag, default ’N’ | Varchar2 (1) |

Indexes on ID, LAST_NAME

## LINE_ITEM

Product details of orders placed by the customer

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| SALES_ORDER_ID | Sales order ID, FK | Number |
| PRODUCT_ID | Product ID, FK | Number |
| LINE_ITEM_NUM | Line item number in sales order | Number |
| TAX_AMT | Tax amount of the product | Number |
| UNIT_PRICE | Price for the product | Number |
| QTY_ORD | Quantity of the product | Number |
| BEFORE_TAX_TOTAL | Before tax total of each product | Number |
| AFTER_TAX_TOTAL | After tax total of each product | Number |
| WEIGHT | Weight of ordered product | Varchar2 (20) |
| WEIGHT_UNIT | Weight of each product | Varchar2 (20) |

Indexes on ID, PRODUCT_ID, SALES_ORDER_ID

References to SALES_ORDER, PRODUCT

## PRODUCT

Contains the details of all the products

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| CATEGORY_ID | Product category ID, FK | Number |
| MANUFACTURER_ID | Manufacturer ID, FK | Number |
| TAX_CLASS_ID | Tax class ID, FK | Number |
| PRODUCT_NAME | Product name | Varchar2 (255) |
| SHORT_DESCRIPTION | Product short description | Varchar2 (500) |
| LONG_DESCRIPTION | Product detailed description | Varchar2 (4000) |
| LIST_PRICE | Price excluding tax, default 0 | Number (8,2) |
| SELLING_PRICE | Price including tax, default 0 | Number (8,2) |
| QOH | Quantity on hand, default 0 | Number |
| AVAILABILITY | Availability for sale, default ' N ' | Varchar2 (1) |
| HOT | Hot product, default ' N ' | Varchar2 (1) |
| WEIGHT | Product weight, default 0 | Number (8,2) |
| DISCOUNT | Discount percent, default 0 | Number (8,2) |
| POINTS | Product rate, default 0 | Number |
| ACTIVATION_DATE | Date to be activated | Data |
| PROMOTION | Promotion product, default ' N ' | Varchar2 (1) |
| DOWNLOAD | Download product | Varchar2 (1) |
| CREATED_DATE | Created date | Date |
| CREATED_BY_ID | Created by person ID | Number |
| MODIFIED_DATE | Modification date | Date |
| MODIFIED_BY_ID | Modified by person ID | Number |
| MORE | Additional information | Varchar2 (200) |
| LICENSE | Product license | Varchar2 (200) |
| SKU | Stock-keeping unit | Varchar2 (2) |

Indexes on ID, ACTIVATION_DATE, CATEGORY_ID, MANUFACTURER_ID, LONG_DESCRIPTION, PRODUCT_NAME, SHORT_DESCRIPTION, TAX_CLASS_ID

References to CATEGORY, MANUFACTURER, TAX_CLASS

## SALES_ORDER

Sales orders 'information

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| CUSTOMER_ID | Customer ID, FK | Number |
| BILL_TO_ID | Billing address ID, FK | Number |
| SHIP_TO_ID | Shipping address ID, FK | Number |
| STATUS_ID | Order status ID, FK | Number |
| SHIPPING_TYPE_ID | Shipping type ID, FK | Number |
| ORDER_DATE | Order date | Date |
| CURRENCY | Order currency | Char |
| TOT_ORDER_AMT | Total order amount | Number |
| FREIGHT_CHARGE | Freight charge for shipping | Number |
| TAX_AMT | Tax amount | Number |
| DELIVERY_DATE | Estimated delivery date | Date |
| COMMENTS | Additional information | Varchar2 (4000) |
| ACTUAL_DELIVERY_DATE | Actual delivery date | Date |
| SHIPPING_NOTICE | Shipping notice | Varchar2 (4000) |

Indexes on ID, BILL_TO_ID, CUSTOMER_ID, SHIP_TO_ID, STATUS_ID

References to CUSTOMER_DETAILS, ORDER_STATUS, BILLING_DETAILS, SHIPPING_DETAILS, SHIPPING_TYPES

### 3.2.2 Product group (see Diagram 2)

CATEGORY

List of product groups (categories)

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| CATEGORY_NAME | Category name | Varchar2 (255) |
| PARENT_ID | Parent category ID | Number |

Indexes on ID, CATEGORY_NAME, PARENT_ID
Reference to CATEGORY

## IMAGE

Contains images used in web-shop

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| PRODUCT_ID | Product ID, FK | Number |
| CATEGORY_ID | Category ID, FK | Number |
| MAUFACTURE_ID | Manufacture ID, FK | Number |
| IMAGE_NAME | Image name | Varchar2 (255) |
| MIME_TYPE | Image type | Varchar2 (32) |
| IMAGE | Image in the blob file | Blob |
| SORT_ORDER | The order in which images are shown in GUI | Number |

Index on ID

References to PRODUCT, CATEGORY, MANUFACTURE

## FAVORITES

Products added to the favorites

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| PRODUCT_ID | Product ID, FK | Number |
| CUSTOMER_ID | Customer who added product to the favorites | Number |
| CREATION_DATE | Created date | Date |

Indexes on ID

References to PRODUCT, CUSTOMER_DETAILS

## FILE

Additional file details of the product added.

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| PRODUCT_ID | Product ID, FK | Number |
| FILE_NAME | Name of the file | Varchar2 (255) |
| MIME_TYPE | Mime type of the file | Varchar2 (32) |
| FILE_BLOB | Blob file | Blob |

Indexes on ID, PRODUCT_ID
Reference to PRODUCT

## MANUFACTURER

List of vendors (manufacturers) of products

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| NAME | Manufacture name, not null | Varchar2 (255) |
| DATE_ADDED | Date it is added | Date |
| LAST_MODIFIED | Date it is modified | Date |
| MANUFACTURER_URL | Manufacture URL | Varchar2 (255) |

Indexes on ID, NAME

## PRODUCT_OPTION

Options (such as size, color etc.) available for any product

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| OPTION_NAME | Name for the option | Varchar2 (255) |
| OPTION_VALUES | Option value | Varchar2 (2000) |
| COMMENTS | Comments | Varchar2 (4000) |

Index on ID

## PRODUCT_TO_PR_OPTION

Product options assigned to particular products

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| PRODUCT_ID | Product ID, FK | Number |
| PRODUCT_OPTION_ID | Product option ID, FK | Number |

Indexes on ID

References to PRODUCT, PRODUCT_OPTION

## REVIEW

Product reviews given by customers

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| PRODUCT_ID | Product Id, FK | Number |
| CUSTOMER_ID | Customer Id, FK | Number |
| TEXT | Text of the review | Varchar (4000) |
| REVIEW_RATING | Review rating | Number |
| DATE_ADDED | Added date | Date |
| LAST_MODIFIED | Last modified date | Date |
| NAME | Review name | Varchar (255) |

Indexes on ID, PRODUCT_ID

References to CUSTOMER_DETAILS, PRODUCT

## TAX_CLASS

Tax classes used in web-shop (for instance, Standard Tax, Reduced Tax etc.)

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| TITLE | Tax title | Varchar2 (32) |
| DESCRIPTION | Description | Varchar2 (255) |
| LAST_MODIFIED | Last modified date | Date |
| DATE_ADDED | Date added | Date |

Index on ID

## TAX_RATE

Taxation rates used in web-shop

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| GEO_ZONE_ID | Geo zone ID, FK | Number |
| CLASS_ID | Tax class ID, FK | Number |
| RATE | Tax rate | Number (8.2) |
| DESCRIPTION | Description | Varchar2 (255) |
| LAST_MODIFIED | Last modified date | Date |
| DATE_ADDED | Date added | Date |

Indexes on ID, GEO_ZONE_ID, CLASS_ID

References to GEO_ZONE, TAX_CLASS

### 3.2.3 Sales Order Group (see Diagram 3)

#### ADDRESS_FORMAT

Presents different standards of address formatting

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| ADDRESS FORMAT | Address format type | Varchar2 (128) |

Index on ID

## BILLING_DETAILS

Customer bill to address information

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| COUNTRY_ID | Country ID, FK | Number |
| STREET | Stores the street address of the customer | Varchar2 (100) |
| CITY | Stores the city name of the customer | Varchar2 (100) |
| STATE | Stores the state name of the customer | Varchar2 (100) |
| PIN | Stores the zip code of the customer | Varchar2 (100) |

Indexes on ID, COUNTRY_ID

Reference to COUNTRY

## ORDER_STATUS

Statuses of the sales order

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| STATUS | Status of sales order | Varchar2 (20) |
| ALLOWED_ORDER | Allowed to be changed | Number |

Index on ID

## SHOPPING_CART

Temporary table: list of products chosen by a customer, but not ordered yet.

| | | |
|---|---|---|
| SESSION_ID | Session Id | VARCHAR2(100) |
| PRODUCT_ID | Product Id, FK | NUMBER |
| QUANTITY | Quantity of product | NUMBER |
| OPTIONS | Product options | VARCHAR2(4000) |
| OPTION_VALUES | Option values | VARCHAR2(4000) |
| CREATED | Created date | DATE |

Reference to PRODUCT

## SHIPPING_DETAILS

Customer ship to address information

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| COUNTRY_ID | Country ID, FK | Number |
| FIRST_NAME | Customer first name | Varchar2 (100) |
| LAST_NAME | Customer last name | Varchar2 (100) |
| STREET | Street address of delivery | Varchar2 (100) |
| CITY | City address of delivery | Varchar2 (100) |
| STATE | State name | Varchar2 (100) |
| PIN | Zip cod | Varchar2 (100) |
| EMAIL | E-mail of the contact person | Varchar2 (100) |
| CONTACT_NUMBER | Phone number of the contact person | Varchar2 (100) |

Indexes on ID

Reference to COUNTRY

## STATES

List of states

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| COUNTRY_ID | Country ID, FK | Number |
| ST | State code | Varchar2 (30) |
| STATE_NAME | State name | Varchar2 (255) |

Index on ID

Reference to COUNTRY

### 3.2.4 Shipping Group (see Diagram 4)

#### ESTIMATED_DELIVERY_TIME

Estimated delivery time

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| DESCRIPTION | Delivery description to be shown in GUI | Varchar2 (100) |
| MIN_DAYS | Minimum business days | Number |
| MAX_DAYS | Maximum business days | Number |

Index on ID

## GEO_ZONE

Geo zones is used in tax and shipping calculations

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| DESCRIPTION | Description for the geo zone | Varchar2 (255) |

Index on ID

## SHIPPING_AMOUNT_RATE

Shipping rate depending on different shipping types

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| SHIPPING_TYPE_ID | Shipping type ID, FK | Number |
| GEO_ZONE_ID | Geo zones ID, FK | Number |
| AMOUNT | Sales order amount on which the shipping rate is based | Number |
| RATE | Shipping price rate | Number |

Indexes on ID, SHIPPING_TYPE_ID, GEO_ZONE_ID

References to SHIPPING_TYPE, GEO_ZONE

## SHIPPING_PRICE_TYPE

List of shipping price types such as Total Amount, Weight, and Quantity

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| DESCRIPTION | Description | Varchar2 (20) |

Index on ID

## SHIPPING_TYPE

Available shipping types

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| PRICE_TYPE_ID | Based on shipping price type, FK | Number |
| ESTIMATED_SHIP_TIME_ID | Estimated shipping time ID, FK | Number |
| SHIPPING_NAME | Shipping name, not null | Varchar2 (255) |
| DESCRIPTION | Description | Varchar2 (4000) |
| ACTIVE | Is the type active or not, default 'Y' | Varchar2 (1) |
| MAX_WEIGHT | Maximum weight allowed shipping | Number |
| FREE | Shipping free of charge, default 'N' | Varchar2 (1) |

Index on ID

References to SHIPPING_PRICE_TYPE, ESTIMATED_SHIPPING_TIME

## ZONE

Geo zones grouped into zones depending on tax percentage or delivery charge

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| COUNTRY_ID | Country ID, FK | Number |
| GEO_ZONE_ID | Shipping geo zone ID, FK | Number |
| CODE | Zone code | Varchar2 (32) |
| NAME | Zone name | Varchar2 (255) |

Indexes on ID, COUNTRY_ID, GEO_ZONE_ID

References to COUNTRY, GEO_ZONE

### 3.2.5 Line Items Group (see Diagram 5)

#### LINE_DOWNLOAD

Keeps the information about downloading of the files

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| LINE_ID | Item to be downloaded ID, FK | Number |
| FILE_ID | File to be downloaded ID, FK | Number |
| DOWNLOAD_EXPIRE_DATE | Date when downloading is expired | Date |
| DOWNLOAD_CLICK_COUNT | Count of downloads | Number |

Indexes on ID, LINE_ITEM_ID, FILE_ID

References to LINE_ITEM, FILE

## LINE_ITEM_OPTION

Product options assigned to the particular line item

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| LINE_ITEM_ID | Line item ID, FK | Number |
| PRODUCT_OPTION_ID | Product option ID, FK | Number |
| OPTION_VALUE | Option value | Varchar2 (100) |

Index on ID

References to PRODUCT_OPTION_ID, LINE_ITEM

### 3.2.6 Payment Group (see Diagram 6)

#### PAYMENT_CC_TYPE

Contains details of various credit cards accepted for payment in the store

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| DESCRIPTION | Credit card name | Varchar2 (4000) |
| AVAILABLE | To be shown or not in the web-shop GUI, default 'N' | Varchar2 (1) |
| PRIORITY | The order in which the credit cards are shown in the GUI | Number |
| PAYPAL | Acceptable of Paypal, default 'N' | Varchar2 (1) |

Index on ID

## PAYMENT_MODE_EX

Details of external payments accepted in the web-shop

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| DESCRIPTION | Description of payment mode | Varchar2 (1000) |
| VENDOR_ID | Vendor or merchant ID | Varchar2 (200) |
| PASSWORD | Password or secret key | Varchar2 (200) |
| SIGNATURE | Signature, default null | Varchar2 (1000) |
| AVAILABLE | Availability in the web-shop, default N' | Varchar2 (1) |
| SANDBOX | Test system address | Varchar2 (200) |
| PRODUCTION | Production system address | Varchar2 (200) |
| TESTMODE | Sandbox should be used, default 'Y' | Varchar2 (1) |
| CHECKOUT_URL_P | Checkout production URL | Varchar2 (255) |
| CHECKOUT_URL_T | Checkout test URL | Varchar2 (255) |
| BUTTON_URL | Button URL | Varchar2 (255) |

Indexes on ID, KEY

## PAYMENT_MODE_IN

Details of internal payments accepted in the web-shop

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| DESCRIPTION | Payment mode description | Varchar2 (4000) |
| AVAILABLE | Accepted or not, default 'N' | Varchar2 (1) |
| PRIORITY | Sorting order | Number |
| KEY | Key used in the code to fetch payment info | Varchar2 (10) |

Index on ID

## THESIS_SO_PAY_DETAILS

Payment details of each sales order placed by the customer

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| SALES_ORDER_ID | Sales order ID, FK | Number |
| PAYMENT_MODE_IN_ID | Internal payment ID, FK | Number |
| CREDIT_CARD_TYPE_ID | Credit card ID (if paid by credit card), FK | Number |
| CREDIT_CARD_NUM | Credit card number (if paid by credit card) | Number |
| EXPIRY_DATE | Credit card expiry date (if paid by credit card) Date | |
| AMOUNT | Sales order total amount | Number |
| CHEQUE_NO | Cheque number (if paid by cheque) | Number |
| CHEQUE_DATE | Cheque date (if paid by cheque) | Date |
| DEMANDDRAFT_NO | Demand draft number (if paid by demand draft) | Varchar2 (100) |
| CVC | Card verification code | Varchar2 (10) |
| TRANSACTION_ID | External system transaction reference | Varchar2 (30) |

Indexes on ID, SALES_ORDER_ID, PAYMENT_MODE_IN_ID

References to SALES_ORDER, PAYMENT_MODE_IN, PAYMENT_MODE_EX, PAYMENT_CC_TYPE

### 3.2.7 Additional tables (Diagram 7)

#### CONFIGURATION

The configuration table for the whole system: the web-shop administrator can set the options to be displayed in the customers.

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| GROUP_ID | Group of settings ID, FK | Number |
| TITLE | Setting title, not null | Varchar2 (64) |
| KEY | Key for value to be fetched | Varchar2 (64) |
| VALUE | Value of the setting, not null | Varchar2 (255) |
| DESCRIPTION | Description for the setting | Varchar2 (4000) |
| SORT_ORDER | Sorting order is used for GUI | Number |
| LAST_MODIFIED | Date of last modification | Date |
| DATE_ADDED | Date of addition | Date |

References to CONFIGURATION_GROUP

Indexes on KEY

## CONFIGURATION_GROUP

Configurations (settings) are grouped into groups

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| IMAGE_ID | Image Id, FK | Number |
| TITLE | Title for the group, not null | Varchar2 (64) |
| DESCRIPTION | Description, not null | Varchar2 (255) |
| SORT_ORDER | Sorted order | Number |
| VISIBLE | Visible or not in GUI | Varchar2 (1) |

Index on ID

References to IMAGE

## CURRENCY

Currency codes and their information

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| TITLE | Currency title | Varchar2 (32) |
| CODE | Code for the currency | Char |
| SYMBOL_LEFT | Left symbol | Varchar2 (12) |
| SYMBOL_RIGHT | Right symbol | Varchar2 (12) |
| DECIMAL_POINT | Decimal separator | Char |
| THOUSANDS_POINT | Thousand separator | Char |
| DECIMAL_PLACES | Number of decimals | Char |
| VALUE | Currency exchange rate | Number (10,5) |
| LAST_UPDATED | Last updated date | Date |

Indexes on ID, CODE

## MESSAGE

Massages sent to the customers or got from the customers

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| CUSTOMER_ID | Customer Id, FK | Number |
| REPLY_ON_ID | Reply on message Id | Number |
| TITLE | Title of the message | Varchar2 (255 BYTE) |
| TEXT | Text of the message | Varchar2 (2000 BYTE) |
| CREATED_DATE | Created date | Date |
| REPLIED_DATE | Replied date | Date |
| READ_DATE | Read date | Date |
| FROM_EMAIL | Got from e-mail | Varchar2 (100) |

Indexes on ID, CUSTOMER_ID, REPLY_ON_ID

References to CUSTOMER_DETAILS

## NEWS

News to be displayed on the GUI

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| TITLE | Title | Varchar2 (255) |
| TEXT | Text | Varchar2 (2000) |
| MODULE | News module | Varchar2 (255) |
| DATE_ADDED | Added date | Date |
| DATE_SENT | Sent date | Date |
| ACTIVE | Active or not, default 'N' | Varchar2 (1) |
| HTML_LINK | HTML link | Varchar2 (100) |
| HTML_LINK_TEXT | Text for HTML link | Varchar2 (100) |

Index on ID

## NEWSLETTER

Newsletters sent to the customers

| | | |
|---|---|---|
| ID | Unique number, PK | Number |
| TITLE | Title | Varchar2 (255) |
| CREATED_DATE | Created date | Date |
| SENT_DATE | Sent date | Date |
| ACTIVE | Is the newsletter active or not, default 'N' | Varchar2 (1) |

Index on ID

# 4 Physical model

In order to test the solution the physical database was implemented in Oracle 10g Express Edition using SQL Developer tool (see Appendix A Create database script). Physical model includes tables with primary and foreign keys and indexes (see Appendix A). The physical characteristics of the tables such as storage and table space were set by Oracle automatically.

Sample data were inserted for testing purposes and views were created (see Appendix B).

# 5 Test Cases and Results

To test that the database design meets the business requirements the test cases are created and the appropriate views or SQL statements are built in database.

Case 1

See Insert data script in Appendix B

Web-store administrator inserts the information about a product and assigns the product to product category, to product manufacture class and tax class.

Case 2

See vw_ product_in_cutegory in Appendix B.

A web-store customer wants to find out all products available in particular product category. The result is shown bellow.

| *Product category* | *Product subcategory* | *Product name* |
|---|---|---|
| Books for kids | Novels for kids | L.Carroll 'Alice in Wonderland' |
| Books for kids | Novels for kids | A.Milne 'Winnie-the-Pooh' |
| Professional books | SAP | ABAP Basics |
| Professional books | SAP | Discover ABAP |
| Books for teens | Novels | J.Rowling 'Harry Potter and Philosopher's ´ Stone' |
| Books for teens | Novels | J.Rowling 'Harry Potter and Half-Blood Prince' |
| Suomenkielinen kirjallisuus | Romaanit | M.Waltari 'Sinuhe egyptiläinen' |

Case 3

See vw_ ordered_products in Appendix B.

Web-store administrator wants to find out who has ordered items from his web-store, what products and how many pieces were ordered. The result is shown bellow.

| *SO num* | *Customer* | *Product* | *Qty* |
|---|---|---|---|
| 1020 | Marina Ivanova | J.Rowling 'Harry Potter and Philosopher's Stone' | 2 |
| 1020 | Marina Ivanova | I.Carroll 'Alice in Wonderland' | 2 |
| 1020 | Marina Ivanova | A.Milne 'Winnie-the-Pooh' | 1 |
| 1010 | Pekka Juhani Aalto | Discover ABAP | 1 |
| 1010 | Pekka Juhani Aalto | ABAP Basics | 1 |

Case 4

See vw_sales_order_delivery_details in Appendix B.

Administrator wishes to know delivery details of all existing sales orders: shipping address and delivery type. See result bellow.

| *SO num* | *Customer* | *Address* | *Delivery type* | *Order status* |
|---|---|---|---|---|
| 1020 | Marina Ivanova | Moskovskaja, 56 - 7 145001 Novosibirsk Russian Federation | Air Service | Delivered |
| 1010 | Pekka Juhani Aalto | Vuosaarentie 6 A 24 00960 Helsinki Finland | Economy | Invoiced |

Case 5

See vw_sales_order_payment_details in Appendix B.

Web-store administrator is interested in sales orders which were paid by credit cards and the amount of them.

| *SO num* | *Customer* | *Order status* | *Credit card type* | *Amount paid* |
|---|---|---|---|---|
| 1010 | Aalto Pekka Juhani | Invoiced | Amex | 153 |

Case 6

See vw_product_details in Appendix B.

The web-shop user makes an order of products and gets price and tax information about items he/she is ordering. The result is shown bellow.

| *Product* | *Short description* | *Category* | *Manufacture* | *Price before tax* | *Tax* | *Total price* |
|---|---|---|---|---|---|---|
| L. Carroll 'Alice in Wonderland' | Story about Alice | Novels | WSOY | 14,02 | Books, Finland,7% | 15 |
| J.Rowling 'Harry Potter and Philosopher's Stone' | Story about Harry Potter | Books for teens | WSOY | 25,25 | Books, Finland,7% | 27 |
| ABAP Basics | ABAP guide | SAP | SAP-PRESS | 44,85 | Books, Finland,7% | 48 |

# Summary

The objectives of the thesis project were reached and presented in the thesis document. The database for e/commerce application in designed, documented, implemented and tested and is ready for further utilization.

The designed database is essential part of e-commerce application. It presents all basic features required for modern e-commerce application but not redundant of unnecessary ones. It can be easily maintained in the future in case of changes of business requirements or customized in order to meet the specific requirements of a particular customer. It may be applicable for any development environment compatible with Oracle but we recommend using Apex technology because of its significant benefits such as low cost, flexibility, professional support and others.

# Bibliography


- Amor D.The E-business (R) evolution, PH PTR, 2002

- Oracle White pages

  http://www.oracle.com/technology/products/database/sql_developer/index.html

  Visited 25.08.2009

  http://www.oracle.com/technology/products/database/sql_developer/index.html

  Visited 13.08.2009

- Smyk Michal The impact of E-Business on small and medium sized enterprises (SME´s). The development of E-Business strategy for Travelon (USA) –case study. Bachelor´s Thesis. Helsinki Business Politechnic, 2001

- Wikipedia. www.wikipedia.org

# Appendix A. Create User and Tables source codes

## Create Database user

```
-- run by SYSTEM
CREATE USER thesis IDENTIFIED BY password;
ALTER USER thesis DEFAULT TABLESPACE USERS TEMPORARY TABLESPACE TEMP
ACCOUNT UNLOCK;
GRANT "RESOURCE","CONNECT" to thesis;
```

## Create tables

```
  CREATE TABLE "THESIS"."ADDRESS_FORMAT"
   (  "ID" NUMBER,
      "ADDRESS_FORMAT" VARCHAR2(128 BYTE),
       CONSTRAINT "ADDRESS_FORMAT_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS";


  CREATE TABLE "THESIS"."BILLING_DETAILS"
   (  "ID" NUMBER,
      "STREET" VARCHAR2(100 BYTE),
      "CITY" VARCHAR2(100 BYTE),
      "STATE" VARCHAR2(100 BYTE),
      "PIN" VARCHAR2(100 BYTE),
      "COUNTRY_ID" NUMBER,
       CONSTRAINT "USER_BILLING_DETAILS_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "BILLING_COUNTRY_FK" FOREIGN KEY ("COUNTRY_ID")
        REFERENCES "THESIS"."COUNTRY" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."CUST_BILLING_DET_COUNTR" ON
"THESIS"."BILLING_DETAILS" ("COUNTRY_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE UNIQUE INDEX "THESIS"."USER_BILLING_DETAILS_PK" ON
"THESIS"."BILLING_DETAILS" ("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
```

```
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."CATEGORY"
   (  "ID" NUMBER,
      "CATEGORY_NAME" VARCHAR2(255 BYTE),
      "PARENT_ID" NUMBER,
       CONSTRAINT "CATEGORY_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE UNIQUE INDEX "THESIS"."CATEGORY_ID_IDX" ON "THESIS"."CATEGORY"
("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."CATEGORY_NAME_IDX" ON "THESIS"."CATEGORY"
("CATEGORY_NAME")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."CATEGORY_PARENT_IDX" ON "THESIS"."CATEGORY"
("PARENT_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."CONFIGURATION"
   (  "ID" NUMBER NOT NULL ENABLE,
      "TITLE" VARCHAR2(64 BYTE) NOT NULL ENABLE,
      "KEY" VARCHAR2(64 BYTE) NOT NULL ENABLE,
      "VALUE" VARCHAR2(255 BYTE) NOT NULL ENABLE,
      "DESCRIPTION" VARCHAR2(4000 BYTE) NOT NULL ENABLE,
      "GROUP_ID" NUMBER,
      "SORT_ORDER" NUMBER,
      "LAST_MODIFIED" DATE,
      "DATE_ADDED" DATE NOT NULL ENABLE,
       CONSTRAINT "CONFIGURATION_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "GROUP_ID_FK" FOREIGN KEY ("GROUP_ID")
        REFERENCES "THESIS"."CONFIGURATION_GROUP" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
```

```
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


   CREATE UNIQUE INDEX "THESIS"."CONFIGURATION_PK" ON
"THESIS"."CONFIGURATION" ("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE UNIQUE INDEX "THESIS"."CONF_KEY_IDX" ON "THESIS"."CONFIGURATION"
("KEY")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."CONFIGURATION_GROUP"
   (  "ID" NUMBER NOT NULL ENABLE,
      "TITLE" VARCHAR2(64 BYTE) NOT NULL ENABLE,
      "DESCRIPTION" VARCHAR2(255 BYTE) NOT NULL ENABLE,
      "SORT_ORDER" NUMBER,
      "VISIBLE" NUMBER,
      "IMAGE_ID" NUMBER,
       CONSTRAINT "CONFIGURATION_GROUP_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."COUNTRY"
   (  "ID" NUMBER,
      "COUNTRY_NAME" VARCHAR2(4000 BYTE),
      "ISO_CODE_1" VARCHAR2(2 BYTE),
      "ISO_CODE_2" VARCHAR2(3 BYTE),
      "ADDRESS_FORMAT_ID" NUMBER,
      "ALLOW" VARCHAR2(1 BYTE) DEFAULT 'Y',
       CONSTRAINT "COUNTRY_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "COUNTRY_UK1" UNIQUE ("COUNTRY_NAME")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "ADDRESS_FORMAT_FK" FOREIGN KEY ("ADDRESS_FORMAT_ID")
        REFERENCES "THESIS"."ADDRESS_FORMAT" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
```

```
  TABLESPACE "USERS" ;


  CREATE UNIQUE INDEX "THESIS"."COUNTRY_PK" ON "THESIS"."COUNTRY" ("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE UNIQUE INDEX "THESIS"."COUNTRY_UK1" ON "THESIS"."COUNTRY"
("COUNTRY_NAME")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;



  CREATE TABLE "THESIS"."CURRENCY"
   (  "ID" NUMBER,
      "TITLE" VARCHAR2(32 BYTE),
      "CODE" CHAR(3 BYTE),
      "SYMBOL_LEFT" VARCHAR2(12 BYTE),
      "SYMBOL_RIGHT" VARCHAR2(12 BYTE),
      "DECIMAL_POINT" CHAR(1 BYTE),
      "THOUSANDS_POINT" CHAR(1 BYTE),
      "DECIMAL_PLACES" CHAR(1 BYTE),
      "VALUE" NUMBER(10,5),
      "LAST_UPDATED" DATE,
       CONSTRAINT "CURRENCY_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "CURRENCY_UK1" UNIQUE ("CODE")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE UNIQUE INDEX "THESIS"."CURRENCY_PK" ON "THESIS"."CURRENCY" ("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE UNIQUE INDEX "THESIS"."CURRENCY_UK1" ON "THESIS"."CURRENCY"
("CODE")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;
```

```
  CREATE TABLE "THESIS"."CUSTOMER_DETAILS"
   (  "ID" NUMBER,
      "FIRST_NAME" VARCHAR2(100 BYTE),
      "LAST_NAME" VARCHAR2(100 BYTE),
      "EMAIL" VARCHAR2(100 BYTE),
      "CONTACT_NUMBER" VARCHAR2(100 BYTE),
      "LOGIN_ID" VARCHAR2(4000 BYTE),
      "PASSWORD" VARCHAR2(100 BYTE),
      "ADMIN" VARCHAR2(3 BYTE) DEFAULT 'No',
       CONSTRAINT "USER_DETAILS_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."CUSTOMER_LAST_IDX" ON "THESIS"."CUSTOMER_DETAILS"
("LAST_NAME")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE UNIQUE INDEX "THESIS"."USER_DETAILS_PK" ON
"THESIS"."CUSTOMER_DETAILS" ("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;



     CREATE TABLE "THESIS"."ESTIMATED_DELIVERY_TIME"
   (  "ID" NUMBER,
      "DESCRIPTION" VARCHAR2(100 BYTE),
      "MIN_DAYS" NUMBER,
      "MAX_DAYS" NUMBER,
       CONSTRAINT "OSL_ESTIMATED_SHIP_TIME_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."FAVORITES"
   (  "ID" NUMBER,
      "PRODUCT_ID" NUMBER,
      "CREATION_DATE" DATE,
      "CUSTOMER_ID" NUMBER,
       CONSTRAINT "FAVORITES_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
```

```
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "PRODUCT_ID_FK1" FOREIGN KEY ("PRODUCT_ID")
        REFERENCES "THESIS"."PRODUCT" ("ID") DISABLE,
       CONSTRAINT "CUSTOMER_ID_FK1" FOREIGN KEY ("CUSTOMER_ID")
        REFERENCES "THESIS"."CUSTOMER_DETAILS" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."FAVORITES_CUSTID" ON "THESIS"."FAVORITES"
("CUSTOMER_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE UNIQUE INDEX "THESIS"."FAVORITES_PK" ON "THESIS"."FAVORITES"
("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."FAVORITES_PRODID" ON "THESIS"."FAVORITES"
("PRODUCT_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;



  CREATE TABLE "THESIS"."FILE"
   (  "PRODUCT_ID" NUMBER,
      "ID" NUMBER,
      "FILE_NAME" VARCHAR2(255 BYTE),
      "MIME_TYPE" VARCHAR2(32 BYTE),
      "FILE_BLOB" BLOB,
       CONSTRAINT "FILES_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "PRODUCT_FK2" FOREIGN KEY ("PRODUCT_ID")
        REFERENCES "THESIS"."PRODUCT" ("ID") DISABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"
 LOB ("FILE_BLOB") STORE AS (
  TABLESPACE "USERS" ENABLE STORAGE IN ROW CHUNK 8192 PCTVERSION 10
  NOCACHE LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)) ;

  CREATE UNIQUE INDEX "THESIS"."FILES_PK" ON "THESIS"."FILE" ("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
```

```
 STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
 TABLESPACE "USERS" ;

 CREATE INDEX "THESIS"."PRODUCT_ID_FK5" ON "THESIS"."FILE" ("PRODUCT_ID")
 PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
 STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
 TABLESPACE "USERS" ;


 CREATE TABLE "THESIS"."GEO_ZONE"
  (  "ID" NUMBER,
     "DESCRIPTION" VARCHAR2(255 BYTE),
     "USED_IN" VARCHAR2(20 BYTE),
      CONSTRAINT "GEO_ZONES_PK" PRIMARY KEY ("ID")
 USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
 STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
 TABLESPACE "USERS"  ENABLE
  ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
 STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
 TABLESPACE "USERS" ;

 CREATE TABLE "THESIS"."IMAGE"
  (  "ID" NUMBER,
     "IMAGE_NAME" VARCHAR2(255 BYTE),
     "MIME_TYPE" VARCHAR2(32 BYTE),
     "IMAGE" BLOB,
     "PRODUCT_ID" NUMBER,
     "CATEGORY_ID" NUMBER,
     "SORT_ORDER" NUMBER,
     "MANUFACTURER_ID" NUMBER,
      CONSTRAINT "IMAGE_PK" PRIMARY KEY ("ID")
 USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
 STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
 TABLESPACE "USERS"  ENABLE,
      CONSTRAINT "PRODUCT_FK_2" FOREIGN KEY ("PRODUCT_ID")
       REFERENCES "THESIS"."PRODUCT" ("ID") DISABLE,
      CONSTRAINT "CATEGORY_FK_2" FOREIGN KEY ("CATEGORY_ID")
       REFERENCES "THESIS"."CATEGORY" ("ID") ENABLE,
      CONSTRAINT "MANUFACTURER_ID_FK_2" FOREIGN KEY ("MANUFACTURER_ID")
       REFERENCES "THESIS"."MANUFACTURER" ("ID") ENABLE
  ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
 STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
 TABLESPACE "USERS"
LOB ("IMAGE") STORE AS (
 TABLESPACE "USERS" ENABLE STORAGE IN ROW CHUNK 8192 PCTVERSION 10
 NOCACHE LOGGING
 STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)) ;


 CREATE TABLE "THESIS"."LINE_DOWNLOAD"
  (  "ID" NUMBER,
     "LINE_ID" NUMBER,
```

```
    "FILE_ID" NUMBER,
    "DOWNLOAD_EXPIRE_DATE" DATE,
    "DOWNLOAD_CLICK_COUNT" NUMBER,
     CONSTRAINT "LINE_DOWNLOADS_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
     CONSTRAINT "LINE_ID_FK" FOREIGN KEY ("LINE_ID")
      REFERENCES "THESIS"."LINE_ITEM" ("ID") ENABLE,
     CONSTRAINT "FILE_ID_FK1" FOREIGN KEY ("FILE_ID")
      REFERENCES "THESIS"."FILE" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE INDEX "THESIS"."FILE_FK2" ON "THESIS"."LINE_DOWNLOAD" ("FILE_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE UNIQUE INDEX "THESIS"."LINE_DOWNLOADS_PK" ON
"THESIS"."LINE_DOWNLOAD" ("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."LINE_FK1" ON "THESIS"."LINE_DOWNLOAD" ("LINE_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."LINE_ITEM"
   (  "ID" NUMBER,
    "SALES_ORDER_ID" NUMBER,
    "LINE_ITEM_NUM" NUMBER,
    "PRODUCT_ID" NUMBER,
    "TAX_AMT" NUMBER,
    "UNIT_PRICE" NUMBER,
    "QTY_ORD" NUMBER,
    "BEFORE_TAX_TOTAL" NUMBER,
    "AFTER_TAX_TOTAL" NUMBER,
    "WEIGHT" NUMBER,
    "WEIGHT_UNIT" VARCHAR2(20 BYTE),
     CONSTRAINT "LINE_ITEMS_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
     CONSTRAINT "SALES_ORDER_ID_FK" FOREIGN KEY ("SALES_ORDER_ID")
      REFERENCES "THESIS"."SALES_ORDER" ("ID") ENABLE,
     CONSTRAINT "PRODUCT_ID_FK3" FOREIGN KEY ("PRODUCT_ID")
      REFERENCES "THESIS"."PRODUCT" ("ID") DISABLE
```

```
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


   CREATE UNIQUE INDEX "THESIS"."LINE_ITEMS_PK" ON "THESIS"."LINE_ITEM"
("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."LINE_ITEMS_PRODID" ON "THESIS"."LINE_ITEM"
("PRODUCT_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."LINE_ITEMS_SOID" ON "THESIS"."LINE_ITEM"
("SALES_ORDER_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."LINE_ITEM_OPTION"
   (  "ID" NUMBER,
      "LINE_ITEM_ID" NUMBER,
      "PRODUCT_OPTION_ID" NUMBER,
      "OPTION_VALUE" VARCHAR2(100 BYTE),
       CONSTRAINT "LINE_ITEM_OPTIONS_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "LINE_ITEM_ID_FK" FOREIGN KEY ("LINE_ITEM_ID")
        REFERENCES "THESIS"."LINE_ITEM" ("ID") ENABLE,
       CONSTRAINT "PRODUCT_OPTION_ID_FK" FOREIGN KEY ("PRODUCT_OPTION_ID")
        REFERENCES "THESIS"."PRODUCT_OPTION" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE TABLE "THESIS"."MANUFACTURER"
   (  "ID" NUMBER,
      "NAME" VARCHAR2(255 BYTE),
      "DATE_ADDED" DATE,
      "LAST_MODIFIED" DATE,
      "MANUFACTURER_URL" VARCHAR2(255 BYTE),
       CONSTRAINT "MANUFACTURER_ID_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
```

```
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


   CREATE UNIQUE INDEX "THESIS"."MANUFACTURER_ID_IDX" ON
"THESIS"."MANUFACTURER" ("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."MANUFACTURER_NAME_IDX" ON "THESIS"."MANUFACTURER"
("NAME")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;



  CREATE TABLE "THESIS"."MESSAGE"
   (  "ID" NUMBER,
      "CUSTOMER_ID" NUMBER,
      "TITLE" VARCHAR2(255 BYTE),
      "TEXT" VARCHAR2(2000 BYTE),
      "CREATED_DATE" DATE,
      "REPLIED_DATE" DATE,
      "READ_DATE" DATE,
      "FROM_EMAIL" VARCHAR2(100 BYTE),
      "REPLY_ON_ID" NUMBER,
       CONSTRAINT "MESSAGES_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "MESSAGES_CUST_FK" FOREIGN KEY ("CUSTOMER_ID")
        REFERENCES "THESIS"."CUSTOMER_DETAILS" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE INDEX "THESIS"."MESSAGES_INDEX1" ON "THESIS"."MESSAGE"
("CUSTOMER_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."MESSAGES_INDEX2" ON "THESIS"."MESSAGE"
("REPLY_ON_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE UNIQUE INDEX "THESIS"."MESSAGES_PK" ON "THESIS"."MESSAGE" ("ID")
```

```
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."NEWS"
   (  "ID" NUMBER,
      "TITLE" VARCHAR2(255 BYTE),
      "TEXT" VARCHAR2(2000 BYTE),
      "MODULE" VARCHAR2(255 BYTE),
      "DATE_ADDED" DATE,
      "DATE_SENT" DATE,
      "ACTIVE" VARCHAR2(3 BYTE),
      "HTML_LINK" VARCHAR2(100 BYTE),
      "HTML_LINK_TEXT" VARCHAR2(100 BYTE),
      "HTML_PAR" VARCHAR2(20 BYTE),
       CONSTRAINT "NEWS_CHK1" CHECK (
active in ('Yes','No')
) ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."NEWSLETTER"
   (  "ID" NUMBER,
      "TITLE" VARCHAR2(255 BYTE),
      "CONTENT" VARCHAR2(2000 BYTE),
      "CREATED_DATE" DATE,
      "SENT_DATE" DATE,
      "ACTIVE" VARCHAR2(1 BYTE) DEFAULT 'N',
       CONSTRAINT "NEWSLETTER_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;



  CREATE TABLE "THESIS"."ORDER_STATUS"
   (  "STATUS" VARCHAR2(20 BYTE),
      "ID" NUMBER,
      "ALLOWED_ORDER" NUMBER,
       CONSTRAINT "ORDER_STATUS_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;
```

```
  CREATE TABLE "THESIS"."PAYMENT_CC_TYPE"
   (  "ID" NUMBER,
      "DESCRIPTION" VARCHAR2(4000 BYTE),
      "AVAILABLE" VARCHAR2(1 BYTE) DEFAULT 'N',
      "PRIORITY" NUMBER DEFAULT 1,
      "PAYPAL" VARCHAR2(1 BYTE) DEFAULT 'N',
       CONSTRAINT "PAYMENT_DETAILS_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."PAYMENT_MODE_EX"
   (  "ID" NUMBER,
      "DESCRIPTION" VARCHAR2(1000 BYTE),
      "VENDOR_ID" VARCHAR2(200 BYTE),
      "PASSWORD" VARCHAR2(200 BYTE),
      "SIGNATURE" VARCHAR2(1000 BYTE) DEFAULT NULL,
      "AVAILABLE" VARCHAR2(1 BYTE) DEFAULT 'N',
      "SANDBOX" VARCHAR2(200 BYTE),
      "PRODUCTION" VARCHAR2(200 BYTE),
      "TESTMODE" VARCHAR2(1 BYTE) DEFAULT 'Y',
      "CHECKOUT_URL_P" VARCHAR2(255 BYTE),
      "KEY" VARCHAR2(10 BYTE),
      "CHECKOUT_URL_T" VARCHAR2(255 BYTE),
      "BUTTON_URL" VARCHAR2(255 BYTE),
       CONSTRAINT "PAYMENT_MODE_DETAILS_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."KEY_IDX" ON "THESIS"."PAYMENT_MODE_EX" ("KEY")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE UNIQUE INDEX "THESIS"."PAYMENT_MODE_DETAILS_PK" ON
"THESIS"."PAYMENT_MODE_EX" ("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."PAYMENT_MODE_IN"
   (  "ID" NUMBER DEFAULT 1,
      "DESCRIPTION" VARCHAR2(4000 BYTE),
```

```
    "AVAILABLE" VARCHAR2(1 BYTE) DEFAULT 'N',
    "PRIORITY" NUMBER DEFAULT 1,
    "KEY" VARCHAR2(10 BYTE),
     CONSTRAINT "PAYMENT_MODE_IN_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."PRODUCT"
   (  "ID" NUMBER,
    "PRODUCT_NAME" VARCHAR2(255 BYTE),
    "SHORT_DESCRIPTION" VARCHAR2(500 BYTE),
    "LONG_DESCRIPTION" VARCHAR2(4000 BYTE),
    "CATEGORY_ID" NUMBER,
    "LIST_PRICE" NUMBER(8,2) DEFAULT 0,
    "SELLING_PRICE" NUMBER(8,2) DEFAULT 0,
    "QOH" NUMBER DEFAULT 0,
    "AVAILABILITY" VARCHAR2(1 BYTE) DEFAULT 'N',
    "HOT" VARCHAR2(1 BYTE) DEFAULT 'N',
    "WEIGHT" NUMBER(8,2) DEFAULT 0,
    "DISCOUNT" NUMBER DEFAULT 0,
    "POINTS" NUMBER,
    "ACTIVATION_DATE" DATE,
    "PROMOTION" VARCHAR2(1 BYTE) DEFAULT 'N',
    "MANUFACTURER_ID" NUMBER,
    "TAX_CLASS_ID" NUMBER,
    "DOWNLOAD" VARCHAR2(1 BYTE),
    "CREATED_DATE" DATE,
    "CREATED_BY_ID" VARCHAR2(20 BYTE),
    "MODIFIED_DATE" DATE,
    "MODIFIED_BY_ID" VARCHAR2(20 BYTE),
    "MORE" VARCHAR2(200 BYTE),
    "LICENSE" VARCHAR2(200 BYTE),
    "SKU" VARCHAR2(20 BYTE),
     CONSTRAINT "PRODUCT_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
     CONSTRAINT "PRODUCT_CATEGORY_FK" FOREIGN KEY ("CATEGORY_ID")
      REFERENCES "THESIS"."CATEGORY" ("ID") ENABLE,
     CONSTRAINT "MANUFACTURER_FK" FOREIGN KEY ("MANUFACTURER_ID")
      REFERENCES "THESIS"."MANUFACTURER" ("ID") ENABLE,
     CONSTRAINT "TAX_CLASS_FK" FOREIGN KEY ("TAX_CLASS_ID")
      REFERENCES "THESIS"."TAX_CLASS" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE INDEX "THESIS"."PRODUCT_ACTDATE_IDX" ON "THESIS"."PRODUCT"
("ACTIVATION_DATE")
```

```
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."PRODUCT_CAT_IDX" ON "THESIS"."PRODUCT"
("CATEGORY_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."PRODUCT_MAN_IDX" ON "THESIS"."PRODUCT"
("MANUFACTURER_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE UNIQUE INDEX "THESIS"."PRODUCT_PK" ON "THESIS"."PRODUCT" ("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."PRODUCT_PR_LONGD_IDX" ON "THESIS"."PRODUCT"
("LONG_DESCRIPTION")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."PRODUCT_PR_NAME_IDX" ON "THESIS"."PRODUCT"
("PRODUCT_NAME")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."PRODUCT_PR_SHORTD_IDX" ON "THESIS"."PRODUCT"
("SHORT_DESCRIPTION")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."PRODUCT_TAX_IDX" ON "THESIS"."PRODUCT"
("TAX_CLASS_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."PRODUCT_OPTION"
   (  "ID" NUMBER,
      "OPTION_NAME" VARCHAR2(255 BYTE),
      "OPTION_VALUES" VARCHAR2(2000 BYTE),
      "COMMENTS" VARCHAR2(4000 BYTE),
```

```
       CONSTRAINT "PRODUCT_OPTIONS_PK" PRIMARY KEY ("ID")
    USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
    STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
    PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
    TABLESPACE "USERS"  ENABLE
     ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
    STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
    PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
    TABLESPACE "USERS" ;

    CREATE UNIQUE INDEX "THESIS"."PR_OPTIONS_ID_IDX" ON
  "THESIS"."PRODUCT_OPTION" ("ID")
    PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
    STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
    PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
    TABLESPACE "USERS" ;


    CREATE TABLE "THESIS"."PRODUCT_TO_PR_OPTION"
     (  "ID" NUMBER,
        "PRODUCT_ID" NUMBER,
        "PRODUCT_OPTION_ID" NUMBER,
         CONSTRAINT "PRODUCT_TO_PR_ID_PK" PRIMARY KEY ("ID")
    USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
    STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
    PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
    TABLESPACE "USERS"  ENABLE,
         CONSTRAINT "PRODUCT_ID_FK2" FOREIGN KEY ("PRODUCT_ID")
          REFERENCES "THESIS"."PRODUCT" ("ID") DISABLE,
         CONSTRAINT "PRODUCT_OPTION_ID_FK1" FOREIGN KEY ("PRODUCT_OPTION_ID")
          REFERENCES "THESIS"."PRODUCT_OPTION" ("ID") ENABLE
     ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
    STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
    PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
    TABLESPACE "USERS" ;


    CREATE UNIQUE INDEX "THESIS"."PRODUCT_TO_PR_ID_IDX" ON
  "THESIS"."PRODUCT_TO_PR_OPTION" ("ID")
    PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
    STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
    PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
    TABLESPACE "USERS" ;


    CREATE TABLE "THESIS"."REVIEW"
     (  "ID" NUMBER,
        "PRODUCT_ID" NUMBER,
        "CUSTOMER_ID" NUMBER,
        "TEXT" VARCHAR2(4000 BYTE),
        "REVIEW_RATING" NUMBER,
        "DATE_ADDED" DATE,
        "LAST_MODIFIED" DATE,
        "NAME" VARCHAR2(255 BYTE),
         CONSTRAINT "REVIEW_PROD_FK" FOREIGN KEY ("PRODUCT_ID")
          REFERENCES "THESIS"."PRODUCT" ("ID") DISABLE,
         CONSTRAINT "REVIEWS_CUSTOMER_FK" FOREIGN KEY ("CUSTOMER_ID")
          REFERENCES "THESIS"."CUSTOMER_DETAILS" ("ID") ENABLE
     ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
```

```
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE INDEX "THESIS"."REVIEW_PROD_IDX" ON "THESIS"."REVIEW"
("PRODUCT_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."SALES_ORDER"
   (  "ID" NUMBER,
      "CUSTOMER_ID" NUMBER,
      "BILL_TO_ID" NUMBER,
      "SHIP_TO_ID" NUMBER,
      "ORDER_DATE" DATE,
      "CURRENCY" CHAR(3 BYTE),
      "TOT_ORDER_AMT" NUMBER,
      "FREIGHT_CHARGE" NUMBER,
      "TAX_AMT" NUMBER,
      "DELIVERY_DATE" DATE,
      "COMMENTS" VARCHAR2(4000 BYTE),
      "STATUS_ID" NUMBER,
      "ACTUAL_DELIVERY_DATE" DATE,
      "SHIPPING_TYPE_ID" NUMBER,
      "SHIPPING_NOTICE" VARCHAR2(1000 BYTE),
       CONSTRAINT "SALES_ORDER_ID" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "CUSTOMER_DETAILS_ID_FK" FOREIGN KEY ("CUSTOMER_ID")
        REFERENCES "THESIS"."CUSTOMER_DETAILS" ("ID") ENABLE,
       CONSTRAINT "SALES_ORDER_STATUS_FK1" FOREIGN KEY ("STATUS_ID")
        REFERENCES "THESIS"."ORDER_STATUS" ("ID") ENABLE,
       CONSTRAINT "SALES_BILL_TO_FK" FOREIGN KEY ("BILL_TO_ID")
        REFERENCES "THESIS"."BILLING_DETAILS" ("ID") ENABLE,
       CONSTRAINT "SALES_SHIP_TO_FK" FOREIGN KEY ("SHIP_TO_ID")
        REFERENCES "THESIS"."SHIPPING_DETAILS" ("ID") ENABLE,
       CONSTRAINT "SHIP_TYPE_ID_FK" FOREIGN KEY ("SHIPPING_TYPE_ID")
        REFERENCES "THESIS"."SHIPPING_TYPE" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE UNIQUE INDEX "THESIS"."SALES_ORDER_ID" ON "THESIS"."SALES_ORDER"
("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."SO_BILL_TO_IDX" ON "THESIS"."SALES_ORDER"
("BILL_TO_ID")
```

```
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."SO_CUST_ID_IDX" ON "THESIS"."SALES_ORDER"
("CUSTOMER_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."SO_SHIP_TO_IDX" ON "THESIS"."SALES_ORDER"
("SHIP_TO_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."SO_STATUS_IDX" ON "THESIS"."SALES_ORDER"
("STATUS_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."SHIPPING_AMOUNT_RATE"
   (  "ID" NUMBER,
      "SHIPPING_TYPE_ID" NUMBER,
      "AMOUNT" NUMBER,
      "RATE" NUMBER(8,2),
      "GEO_ZONE_ID" NUMBER,
       CONSTRAINT "SHIP_MOUNT_RATE_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "SHIPPING_TYPE_ID_FK" FOREIGN KEY ("SHIPPING_TYPE_ID")
        REFERENCES "THESIS"."SHIPPING_TYPE" ("ID") ENABLE,
       CONSTRAINT "GEO_ZONES_ID1_FK" FOREIGN KEY ("GEO_ZONE_ID")
        REFERENCES "THESIS"."GEO_ZONE" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE INDEX "THESIS"."GEO_ZONES_ID_IDX" ON
"THESIS"."SHIPPING_AMOUNT_RATE" ("GEO_ZONE_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."SHIPPING_AM_RATES_TYPE" ON
"THESIS"."SHIPPING_AMOUNT_RATE" ("SHIPPING_TYPE_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
```

```
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE UNIQUE INDEX "THESIS"."SHIP_MOUNT_RATE_PK" ON
"THESIS"."SHIPPING_AMOUNT_RATE" ("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."SHIPPING_DETAILS"
   (  "ID" NUMBER,
      "FIRST_NAME" VARCHAR2(100 BYTE),
      "LAST_NAME" VARCHAR2(100 BYTE),
      "STREET" VARCHAR2(100 BYTE),
      "CITY" VARCHAR2(100 BYTE),
      "STATE" VARCHAR2(100 BYTE),
      "PIN" VARCHAR2(100 BYTE),
      "COUNTRY_ID" NUMBER,
      "EMAIL" VARCHAR2(100 BYTE),
      "CONTACT_NUMBER" VARCHAR2(100 BYTE),
       CONSTRAINT "SHIPPING_DETAILS_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "SHIP_COUNTRY_FK" FOREIGN KEY ("COUNTRY_ID")
        REFERENCES "THESIS"."COUNTRY" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."SHIPPING_PRICE_TYPE"
   (  "ID" NUMBER,
      "DESCRIPTION" VARCHAR2(20 BYTE),
       CONSTRAINT "SHIPPING_PRICE_TYPES_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."SHIPPING_TYPE"
   (  "ID" NUMBER,
      "SHIPPING_NAME" VARCHAR2(255 BYTE),
      "DESCRIPTION" VARCHAR2(4000 BYTE),
      "ACTIVE" VARCHAR2(1 BYTE) DEFAULT 'Y',
      "MAX_WEIGHT" NUMBER,
      "PRICE_TYPE_ID" NUMBER DEFAULT 1,
      "ESTIMATED_SHIP_TIME_ID" NUMBER DEFAULT 1,
      "FREE" VARCHAR2(1 BYTE) DEFAULT 'N',
       CONSTRAINT "SHIPPING_TYPES_PK" PRIMARY KEY ("ID")
```

```
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "EST_SHIP_TIME_FK" FOREIGN KEY ("ESTIMATED_SHIP_TIME_ID")
        REFERENCES "THESIS"."ESTIMATED_DELIVERY_TIME" ("ID") ENABLE,
       CONSTRAINT "PRICE_TYPE_FK" FOREIGN KEY ("PRICE_TYPE_ID")
        REFERENCES "THESIS"."SHIPPING_PRICE_TYPE" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."SHOPPING_CART"
   (  "SESSION_ID" VARCHAR2(100 BYTE),
      "PRODUCT_ID" NUMBER,
      "QUANTITY" NUMBER,
      "OPTIONS" VARCHAR2(4000 BYTE),
      "OPTION_VALUES" VARCHAR2(4000 BYTE),
      "CREATED" DATE,
       CONSTRAINT "SHOPPING_CART_PK" PRIMARY KEY ("SESSION_ID",
"PRODUCT_ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "PRODUCT_ID_FK4" FOREIGN KEY ("PRODUCT_ID")
        REFERENCES "THESIS"."PRODUCT" ("ID") DISABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."SO_PAY_DETAILS"
   (  "ID" NUMBER,
      "SALES_ORDER_ID" NUMBER,
      "CUSTOMER_ID" NUMBER,
      "PAYMENT_MODE_IN_ID" NUMBER,
      "CREDIT_CARD_TYPE_ID" NUMBER,
      "CREDIT_CARD_NUM" NUMBER,
      "EXPIRY_DATE" DATE,
      "AMOUNT" NUMBER,
      "CHEQUE_NO" VARCHAR2(100 BYTE),
      "CHEQUE_DATE" DATE,
      "DEMANDDRAFT_NO" VARCHAR2(100 BYTE),
      "CVC" VARCHAR2(10 BYTE),
      "TRANSACTION_ID" VARCHAR2(30 BYTE),
      "PAYMENT_MODE_EX_ID" NUMBER,
       CONSTRAINT "SO_PAY_DETAILS_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "PAYMENT_MODE_IN_FK" FOREIGN KEY ("PAYMENT_MODE_IN_ID")
        REFERENCES "THESIS"."PAYMENT_MODE_IN" ("ID") ENABLE,
       CONSTRAINT "PAYMENT_MODE_EX_FK" FOREIGN KEY ("PAYMENT_MODE_EX_ID")
        REFERENCES "THESIS"."PAYMENT_MODE_EX" ("ID") ENABLE,
```

```
      CONSTRAINT "CC_TYPE_ID_FK" FOREIGN KEY ("CREDIT_CARD_TYPE_ID")
       REFERENCES "THESIS"."PAYMENT_CC_TYPE" ("ID") ENABLE,
      CONSTRAINT "SALES_ORDER_FK3" FOREIGN KEY ("SALES_ORDER_ID")
       REFERENCES "THESIS"."SALES_ORDER" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE INDEX "THESIS"."PAYMENT_IN_ID_IDX" ON "THESIS"."SO_PAY_DETAILS"
("PAYMENT_MODE_IN_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE UNIQUE INDEX "THESIS"."SO_PAY_DETAILS_PK" ON
"THESIS"."SO_PAY_DETAILS" ("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."SO_PAY_SO_ID_IDX" ON "THESIS"."SO_PAY_DETAILS"
("SALES_ORDER_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."STATES"
   (  "ID" NUMBER,
      "COUNTRY_ID" NUMBER,
      "ST" VARCHAR2(30 BYTE),
      "STATE_NAME" VARCHAR2(255 BYTE),
       CONSTRAINT "COUNTRY_ID_FK" FOREIGN KEY ("COUNTRY_ID")
        REFERENCES "THESIS"."COUNTRY" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."TAX_CLASS"
   (  "ID" NUMBER,
      "TITLE" VARCHAR2(32 BYTE),
      "DESCRIPTION" VARCHAR2(255 BYTE),
      "LAST_MODIFIED" DATE,
      "DATE_ADDED" DATE,
       CONSTRAINT "TAX_CLASS_ID_PK" PRIMARY KEY ("ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
```

```
  TABLESPACE "USERS" ;


  CREATE UNIQUE INDEX "THESIS"."TAX_CLASS_ID_PK" ON "THESIS"."TAX_CLASS"
("ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE UNIQUE INDEX "THESIS"."TAX_CLASS_PK_IDX" ON "THESIS"."TAX_CLASS"
("TITLE", "ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."TAX_RATE"
   (  "ID" NUMBER,
      "GEO_ZONE_ID" NUMBER,
      "CLASS_ID" NUMBER,
      "RATE" NUMBER(8,4),
      "DESCRIPTION" VARCHAR2(255 BYTE),
      "LAST_MODIFIED" DATE,
      "DATE_ADDED" DATE,
       CONSTRAINT "TAX_RATES_UK1" UNIQUE ("GEO_ZONE_ID", "CLASS_ID")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS"  ENABLE,
       CONSTRAINT "TAX_RATES_GEO_ZONES_FK1" FOREIGN KEY ("GEO_ZONE_ID")
        REFERENCES "THESIS"."GEO_ZONE" ("ID") ENABLE,
       CONSTRAINT "TAX_RATES_TAX_CLASS_FK1" FOREIGN KEY ("CLASS_ID")
        REFERENCES "THESIS"."TAX_CLASS" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE UNIQUE INDEX "THESIS"."TAX_RATES_UK1" ON "THESIS"."TAX_RATE"
("GEO_ZONE_ID", "CLASS_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;


  CREATE TABLE "THESIS"."ZONE"
   (  "ID" NUMBER,
      "COUNTRY_ID" NUMBER,
      "CODE" VARCHAR2(32 BYTE),
      "NAME" VARCHAR2(255 BYTE),
      "GEO_ZONE_ID" NUMBER,
       CONSTRAINT "COUNTRY_3_FK" FOREIGN KEY ("COUNTRY_ID")
        REFERENCES "THESIS"."COUNTRY" ("ID") ENABLE,
       CONSTRAINT "GEO_ZONE_2_FK" FOREIGN KEY ("GEO_ZONE_ID")
        REFERENCES "THESIS"."GEO_ZONE" ("ID") ENABLE
   ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
```

```
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."GEO_ZONE_FK1" ON "THESIS"."ZONE" ("GEO_ZONE_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;

  CREATE INDEX "THESIS"."THSIS_COONTRY_FK2" ON "THESIS"."THESIS_ZONE"
("COUNTRY_ID")
  PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
  TABLESPACE "USERS" ;
```

# Appendix B. Database Views Source Code

```
INSERT INTO THESIS_PRODUCT (product_name, shoort_description ,
long_description ,    list_price ,  selling_price ,  qoh,availability,
activation_date, manufacturer_id,category_id, tax_class_id)
SELECT 'M. Waltari ´Sinuhe egyptiläinen’', 'Historiallinen romaani',
'Maailman kuuluisa historiallinen romaani',
25, 28, 105, 'Y', sysdate, m.id, c.id, t.id
FROM thesis_manufacturer m, thesis_category c, thesis_tax_class t
WHERE m.name = 'WSOY'
AND c.category_name = 'Suomenkielinen kirjallisuus'
AND t.description = 'VAT on books in Finland'
;

  CREATE OR REPLACE FORCE VIEW "THESIS"."V_ORDERED_PRODUCTS" ("SO number",
"Customer name", "Product", "Quantity")
AS select thesis_line_item.sales_order_id "SO number", first_name||'
'||last_name "Customer name",
product_name "Product", thesis_line_item.qty_ord "Quantity"
from thesis_customer_details, thesis_product, thesis_line_item,
thesis_sales_order
where thesis_line_item.product_id = thesis_product.id
and thesis_line_item.sales_order_id = thesis_sales_order.id
and thesis_sales_order.customer_id = thesis_customer_details.id
order by thesis_line_item.qty_ord desc;


  CREATE OR REPLACE FORCE VIEW "THESIS"."VW_PRODUCT_DETAILS" ("Product
name", "Short description", "Category", "Vendor", "Price before tax", "Tax
class and rate", "Price after tax")
AS select p.product_name "Product name", p.short_description "Short
description",
c.category_name "Category", m.name "Vendor", p.list_price "Price before
tax",
t.title ||','|| r.rate||'%' "Tax class and rate",
p.selling_price "Price after tax"
from thesis_product p
join thesis_category c
on p.category_id = c.id
join thesis_manufacturer m
on p.manufacturer_id = m.id
join thesis_tax_class t
on p.tax_class_id = t.id
join thesis_tax_rate r
on t.id = r.class_id;


  CREATE OR REPLACE FORCE VIEW "THESIS"."VW_SALES_ORDER_DELIVERY_DETAILS"
("Sales Order number", "Buyer", "Shipping address", "Delivery type",
"Status")
AS select DISTINCT thesis_sales_order.id "Sales Order number",
thesis_customer_details.first_name||' '||thesis_customer_details.last_name
"Buyer",
thesis_shipping_details.street||' '||thesis_shipping_details.pin||' '||
thesis_shipping_details.city||' '||thesis_country.country_name "Shipping
address",
```

```
thesis_shipping_type.shipping_name "Delivery type",
thesis_order_status.status "Status"
from thesis_sales_order
inner join thesis_customer_details
on thesis_sales_order.customer_id = thesis_customer_details.id
inner join thesis_shipping_details
on thesis_sales_order.ship_to_id = thesis_shipping_details.id
inner join thesis_country
on thesis_shipping_details.country_id = thesis_country.id
inner join thesis_shipping_type
on thesis_shipping_type.id = thesis_sales_order.shipping_type_id
inner  join thesis_shipping_amount_rate
on thesis_shipping_amount_rate.shipping_type_id =
thesis_sales_order.shipping_type_id
inner join thesis_order_status
on thesis_sales_order.status_id = thesis_order_status.id;


  CREATE OR REPLACE FORCE VIEW "THESIS"."V_SALES_ORDER_PAYMENT_DETAILS"
("Sales Order number", "Customer", "Order status", "Credit card type",
"Amount paid")
AS select thesis_sales_order.id "Sales Order number",
thesis_customer_details.last_name||' '||thesis_customer_details.first_name
"Customer",
thesis_order_status.status "Order status",
thesis_payment_cc_type.description "Credit card type",
thesis_so_pay_details.amount "Amount paid"
from thesis_sales_order
join thesis_customer_details
on thesis_sales_order.customer_id = thesis_customer_details.id
join thesis_order_status
on thesis_sales_order.status_id = thesis_order_status.id
join thesis_so_pay_details
on thesis_so_pay_details.sales_order_id = thesis_sales_order.id
join thesis_payment_cc_type
on thesis_so_pay_details.credit_card_type_id = thesis_payment_cc_type.id;
```