(01-4) 데이터베이스 설계

Gaillardia / Flo:be(플로비) - 플라워 카페 정보 제공 및 상품 주문/예약 서비스

ERD Diagram 1 (Logical)

seller
- sellerId
- name
- pwd
- phone
- email

cartItem
- cartItemId
- customerId (FK)
- productId (FK)
- quantity

customer
- customerId
- name
- pwd
- phone
- email
- address

reply
- qnapostId (FK)
- sellerId (FK)
- content
- creationDate

qna
- qnaId
- customerId (FK)
- productId (FK)
- title
- content
- creationDate
- viewCount
- pwd

product
- productId
- sellerId (FK)
- name
- price
- description
- type
- category

class
- classId
- sellerId (FK)
- name
- classDate
- maxNum
- currentNum

lineItem
- lineItemId
- orderId (FK)
- productId (FK)
- quantity
- price

review
- reviewId
- customerId (FK)
- productId (FK)
- title
- content
- creationDate
- viewCount
- rate

order
- orderId
- customerId (FK)
- creationDate
- address

reservation
- reservationId
- classId (FK)
- customerId (FK)

Relationship labels: 관련한다 / 관련된다, 작성한다 / 작성된다, 가진다 / 담아둔다, 삭성한다 / 작성된다, 소유한다 / 속한다, 포함된다 / 포함한다, 진행한다 / 진행된다, 가성나 / 작성된다, 가진다 / 작성된다, 실행한나 / 진행된다

ERD Diagram 2 (Physical)

seller
- sellerId: VARCHAR2(30) NOT NULL
- name: VARCHAR2(30) NOT NULL
- pwd: VARCHAR2(30) NOT NULL
- phone: VARCHAR2(30) NOT NULL
- email: VARCHAR2(30) NOT NULL

cartItem
- cartItemId: NUMBER NOT NULL
- customerId: VARCHAR2(30) NOT NULL (FK)
- productId: NUMBER NOT NULL (FK)
- quantity: NUMBER NULL

customer
- customerId: VARCHAR2(30) NOT NULL
- name: VARCHAR2(30) NOT NULL
- pwd: VARCHAR2(30) NOT NULL
- phone: VARCHAR2(30) NOT NULL
- email: VARCHAR2(30) NOT NULL
- address: VARCHAR2(50) NOT NULL

reply
- qnaId: NUMBER NOT NULL (FK)
- sellerId: VARCHAR2(30) NOT NULL (FK)
- content: VARCHAR2(1200) NULL
- creationDate: DATE NULL

qna
- qnaId: NUMBER NOT NULL
- customerId: VARCHAR2(30) NOT NULL (FK)
- productId: NUMBER NOT NULL (FK)
- title: VARCHAR2(90) NULL
- content: VARCHAR2(1200) NULL
- creationDate: DATE NULL
- viewCount: NUMBER NULL
- pwd: VARCHAR2(30) NULL

product
- productId: NUMBER NOT NULL
- sellerId: VARCHAR2(30) NOT NULL (FK)
- name: VARCHAR2(30) NOT NULL
- price: NUMBER NULL
- description: VARCHAR2(1200) NULL
- type: VARCHAR2(30) NULL
- category: VARCHAR2(21) NULL

class
- classId: NUMBER NOT NULL
- sellerId: VARCHAR2(30) NOT NULL (FK)
- name: VARCHAR2(45) NOT NULL
- classDate: DATE NULL
- maxNum: NUMBER NULL
- currrentNum: NUMBER NULL

lineItem
- lineItemId: VARCHAR2(30) NOT NULL
- orderId: NUMBER NOT NULL (FK)
- productId: NUMBER NOT NULL (FK)
- quantity: NUMBER NULL
- price: NUMBER NULL

review
- reviewId: NUMBER NOT NULL
- customerId: VARCHAR2(30) NOT NULL (FK)
- productId: NUMBER NOT NULL (FK)
- title: VARCHAR2(90) NULL
- content: VARCHAR2(1200) NULL
- creationDate: DATE NULL
- viewCount: NUMBER NULL
- rate: NUMBER(1) NULL

order
- orderId: NUMBER NOT NULL
- customerId: VARCHAR2(30) NOT NULL (FK)
- creationDate: DATE NULL
- address: VARCHAR2(30) NULL

reservation
- reservationId: NUMBER NOT NULL
- customerId: VARCHAR2(30) NOT NULL (FK)
- classId: NUMBER NOT NULL (FK)

[DDL]

CREATE SEQUENCE Sequence_cartItem;

CREATE SEQUENCE Sequence_class;

CREATE SEQUENCE Sequence_lineItem;

CREATE SEQUENCE Sequence_order;

CREATE SEQUENCE Sequence_product;

CREATE SEQUENCE Sequence_qna;

CREATE SEQUENCE Sequence_reply;

CREATE SEQUENCE Sequence_reservation;

CREATE SEQUENCE Sequence_review;

CREATE TABLE seller
(
        sellerId   VARCHAR2(30) NOT NULL ,
        name   VARCHAR2(30) NOT NULL ,
        pwd   VARCHAR2(30) NOT NULL ,
        phone   VARCHAR2(30) NOT NULL ,
        email   VARCHAR2(30) NOT NULL
);

CREATE UNIQUE INDEX XPKseller ON seller
(sellerId   ASC);

ALTER TABLE seller
        ADD CONSTRAINT XPKseller PRIMARY KEY (sellerId);

CREATE TABLE class
(
        classId   NUMBER NOT NULL ,
        name   VARCHAR2(45) NOT NULL ,
        classDate   DATE   NULL ,
        maxNum   NUMBER NULL ,
        currrentNum   NUMBER NULL ,
        sellerId   VARCHAR2(30) NOT NULL
);

CREATE UNIQUE INDEX XPKclass ON class
(classId   ASC);

ALTER TABLE class
        ADD CONSTRAINT XPKclass PRIMARY KEY (classId);

CREATE TABLE customer
(
        customerId   VARCHAR2(30) NOT NULL ,
        name   VARCHAR2(30) NOT NULL ,
        pwd   VARCHAR2(30) NOT NULL ,
        phone   VARCHAR2(30) NOT NULL ,
        email   VARCHAR2(30) NOT NULL ,
        address   VARCHAR2(50) NOT NULL
);

CREATE UNIQUE INDEX XPKcustomer ON customer
(customerId   ASC);

ALTER TABLE customer
        ADD   CONSTRAINT   XPKcustomer PRIMARY KEY
(customerId);

CREATE TABLE order
(
        orderId   NUMBER NOT NULL ,
        creationDate   DATE NULL ,
        customerId   VARCHAR2(30) NOT NULL ,
        address   VARCHAR2(30) NULL
);

CREATE UNIQUE INDEX XPKorder ON order
(orderId   ASC);

ALTER TABLE order
        ADD CONSTRAINT XPKorder PRIMARY KEY (orderId);

CREATE TABLE reservation
(
        reservationId   NUMBER NOT NULL ,
        classId   NUMBER NOT NULL ,
        customerId   VARCHAR2(30) NOT NULL
);

CREATE UNIQUE INDEX XPKreservation ON reservation
(reservationId   ASC);

ALTER TABLE reservation
        ADD CONSTRAINT   XPKreservation PRIMARY KEY
(reservationId);

CREATE TABLE product
(
        productId   NUMBER NOT NULL ,
        price   NUMBER NULL ,
        description   VARCHAR2(1200) NULL ,
        type   VARCHAR2(30) NULL ,
        category   VARCHAR2(21) NULL ,
        sellerId   VARCHAR2(30) NOT NULL ,
        name   VARCHAR2(30) NOT NULL
);

CREATE UNIQUE INDEX XPKproduct ON product
(productId   ASC);

ALTER TABLE product
        ADD   CONSTRAINT   XPKproduct PRIMARY   KEY
(productId);

CREATE TABLE review
(
        reviewId   NUMBER NOT NULL ,
        title   VARCHAR2(90) NULL ,
        content   VARCHAR2(1200) NULL ,
        creationDate   DATE NULL ,
        viewCount   NUMBER NULL ,
        rate   NUMBER(1) NULL ,
        customerId   VARCHAR2(30) NOT NULL ,
        productId   NUMBER NOT NULL
);

CREATE UNIQUE INDEX XPKreview ON review
(reviewId   ASC);

ALTER TABLE review
        ADD CONSTRAINT XPKreview PRIMARY KEY (reviewId);

CREATE TABLE cartItem
(
        cartItemId   NUMBER NOT NULL ,
        quantity   NUMBER NULL ,

```sql
          customerId              VARCHAR2(30)  NOT NULL ,
          productId               NUMBER  NOT NULL
);

CREATE UNIQUE INDEX XPKcartItem ON cartItem
(cartItemId     ASC);

ALTER TABLE cartItem
          ADD     CONSTRAINT     XPKcartItem   PRIMARY   KEY
(cartItemId);

CREATE TABLE lineItem
(
          lineItemId              VARCHAR2(30)  NOT NULL ,
          productId               NUMBER  NOT NULL ,
          orderId                 NUMBER  NOT NULL ,
          quantity                NUMBER  NULL ,
          price                   NUMBER  NULL
);

CREATE UNIQUE INDEX XPKlineItem ON lineItem
(lineItemId     ASC);

ALTER TABLE lineItem
          ADD     CONSTRAINT     XPKlineItem   PRIMARY   KEY
(lineItemId);

CREATE TABLE qna
(
          qnaId                   NUMBER  NOT NULL ,
          title                   VARCHAR2(90)  NULL ,
          content                 VARCHAR2(1200)  NULL ,
          creationDate            DATE  NULL ,
          viewCount               NUMBER  NULL ,
          pwd                     VARCHAR2(30)  NULL ,
          customerId              VARCHAR2(30)  NOT NULL ,
          productId               NUMBER  NOT NULL
);

CREATE UNIQUE INDEX XPKqna ON qna
(qnaId     ASC);

ALTER TABLE qna
          ADD CONSTRAINT  XPKqna PRIMARY KEY (qnaId);

CREATE TABLE reply
(
          content                 VARCHAR2(1200)  NULL ,
          creationDate            DATE  NULL ,
          sellerId                VARCHAR2(30)  NOT NULL ,
          qnaId                   NUMBER  NOT NULL
);

CREATE UNIQUE INDEX XPKreply ON reply
(qnaId     ASC);

ALTER TABLE reply
          ADD CONSTRAINT  XPKreply PRIMARY KEY (qnaId);

ALTER TABLE class
          ADD (
CONSTRAINT   R_13   FOREIGN   KEY   (sellerId)   REFERENCES   seller
(sellerId));

ALTER TABLE order
          ADD (
CONSTRAINT R_21 FOREIGN KEY (customerId) REFERENCES customer
(customerId));

ALTER TABLE reservation
          ADD (
CONSTRAINT R_16 FOREIGN KEY (classId) REFERENCES class (classId));

ALTER TABLE reservation
          ADD (
CONSTRAINT R_17 FOREIGN KEY (customerId) REFERENCES customer
(customerId));

ALTER TABLE product
          ADD (
CONSTRAINT   R_12   FOREIGN   KEY   (sellerId)   REFERENCES   seller
(sellerId));

ALTER TABLE review
          ADD (
CONSTRAINT R_14 FOREIGN KEY (customerId) REFERENCES customer
(customerId));

ALTER TABLE review
          ADD (
CONSTRAINT   R_15   FOREIGN   KEY   (productId)   REFERENCES   product
(productId));

ALTER TABLE cartItem
          ADD (
CONSTRAINT R_18 FOREIGN KEY (customerId) REFERENCES customer
(customerId));

ALTER TABLE cartItem
          ADD (
CONSTRAINT   R_19   FOREIGN   KEY   (productId)   REFERENCES   product
(productId));

ALTER TABLE lineItem
          ADD (
CONSTRAINT   R_29   FOREIGN   KEY   (productId)   REFERENCES   product
(productId));

ALTER TABLE lineItem
          ADD (
CONSTRAINT   R_30   FOREIGN   KEY   (orderId)   REFERENCES   order
(orderId));

ALTER TABLE qna
          ADD (
CONSTRAINT R_7 FOREIGN KEY (customerId) REFERENCES customer
(customerId));

ALTER TABLE qna
          ADD (
CONSTRAINT   R_10   FOREIGN   KEY   (productId)   REFERENCES   product
(productId));

ALTER TABLE reply
          ADD (
CONSTRAINT R_2 FOREIGN KEY (sellerId) REFERENCES seller (sellerId));

ALTER TABLE reply
          ADD (
CONSTRAINT R_4 FOREIGN KEY (qnaId) REFERENCES qna (qnaId));
```