```
-- MySQL Workbench Forward Engineering

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,N
O_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';

-- -----------------------------------------------------
-- Schema Nike
-- -----------------------------------------------------

-- -----------------------------------------------------
-- Schema Nike
-- -----------------------------------------------------
CREATE SCHEMA IF NOT EXISTS `Nike` DEFAULT CHARACTER SET utf8 ;
USE `Nike` ;

-- -----------------------------------------------------
-- Table `Nike`.`audience`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `Nike`.`audience` (
  `audience_id` INT NOT NULL,
  `audience_type` VARCHAR(20) NOT NULL,
  PRIMARY KEY (`audience_id`),
  UNIQUE INDEX `id_UNIQUE` (`audience_id` ASC) VISIBLE,
  UNIQUE INDEX `audience_type_UNIQUE` (`audience_type` ASC) VISIBLE)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `Nike`.`Customer`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `Nike`.`Customer` (
  `customer_id` INT NOT NULL,
  `first_name` VARCHAR(45) NULL,
  `last_name` VARCHAR(45) NULL,
  `phone_number` VARCHAR(45) NULL,
  PRIMARY KEY (`customer_id`))
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `Nike`.`shoes`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `Nike`.`shoes` (
  `shoes_id` INT NOT NULL AUTO_INCREMENT,
```

```
  `shoe_name` VARCHAR(50) NOT NULL,
  `photo` BLOB NULL,
  `audience_id` INT NOT NULL,
  `customer_id` INT NOT NULL,
  PRIMARY KEY (`shoes_id`),
  UNIQUE INDEX `id_UNIQUE` (`shoes_id` ASC) VISIBLE,
  UNIQUE INDEX `shoe_name_UNIQUE` (`shoe_name` ASC) VISIBLE,
  INDEX `fk_shoes_audience1_idx` (`audience_id` ASC) VISIBLE,
  INDEX `fk_shoes_Customer1_idx` (`customer_id` ASC) VISIBLE,
  CONSTRAINT `fk_shoes_audience1`
    FOREIGN KEY (`audience_id`)
    REFERENCES `Nike`.`audience` (`audience_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_shoes_Customer1`
    FOREIGN KEY (`customer_id`)
    REFERENCES `Nike`.`Customer` (`customer_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `Nike`.`features`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `Nike`.`features` (
  `feature_id` INT NOT NULL,
  `feature_type` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`feature_id`),
  UNIQUE INDEX `id_UNIQUE` (`feature_id` ASC) VISIBLE,
  UNIQUE INDEX `lifestyle_type_UNIQUE` (`feature_type` ASC) VISIBLE)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `Nike`.`shoes_has_feature`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `Nike`.`shoes_has_feature` (
  `shoes_id` INT NOT NULL,
  `feature_id` INT NOT NULL,
  PRIMARY KEY (`shoes_id`, `feature_id`),
  INDEX `fk_shoes_has_lifestyle_lifestyle1_idx` (`feature_id` ASC) VISIBLE,
  INDEX `fk_shoes_has_lifestyle_shoes1_idx` (`shoes_id` ASC) VISIBLE,
  CONSTRAINT `fk_shoes_has_lifestyle_shoes1`
    FOREIGN KEY (`shoes_id`)
    REFERENCES `Nike`.`shoes` (`shoes_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
```

```
  CONSTRAINT `fk_shoes_has_lifestyle_lifestyle1`
    FOREIGN KEY (`feature_id`)
    REFERENCES `Nike`.`features` (`feature_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `Nike`.`city`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `Nike`.`city` (
  `city_id` INT NOT NULL,
  `city_name` VARCHAR(45) NULL,
  `customer_id` INT NOT NULL,
  PRIMARY KEY (`city_id`, `customer_id`),
  INDEX `fk_city_Customer1_idx` (`customer_id` ASC) VISIBLE,
  CONSTRAINT `fk_city_Customer1`
    FOREIGN KEY (`customer_id`)
    REFERENCES `Nike`.`Customer` (`customer_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `Nike`.`address`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `Nike`.`address` (
  `address_id` INT NOT NULL,
  `apt_no` INT NOT NULL,
  `street_name` VARCHAR(45) NULL,
  `postal_code` VARCHAR(45) NULL,
  `city_id` INT NOT NULL,
  `customer_id` INT NOT NULL,
  PRIMARY KEY (`address_id`, `city_id`, `customer_id`),
  INDEX `fk_address_city1_idx` (`city_id` ASC) VISIBLE,
  INDEX `fk_address_Customer1_idx` (`customer_id` ASC) VISIBLE,
  CONSTRAINT `fk_address_city1`
    FOREIGN KEY (`city_id`)
    REFERENCES `Nike`.`city` (`city_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_address_Customer1`
    FOREIGN KEY (`customer_id`)
    REFERENCES `Nike`.`Customer` (`customer_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
```

```
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `Nike`.`Customer_has_audience`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `Nike`.`Customer_has_audience` (
  `customer_id` INT NOT NULL,
  `audience_id` INT NOT NULL,
  PRIMARY KEY (`customer_id`, `audience_id`),
  INDEX `fk_Customer_has_audience_audience1_idx` (`audience_id` ASC) VISIBLE,
  INDEX `fk_Customer_has_audience_Customer1_idx` (`customer_id` ASC) VISIBLE,
  CONSTRAINT `fk_Customer_has_audience_Customer1`
    FOREIGN KEY (`customer_id`)
    REFERENCES `Nike`.`Customer` (`customer_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Customer_has_audience_audience1`
    FOREIGN KEY (`audience_id`)
    REFERENCES `Nike`.`audience` (`audience_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `Nike`.`Payment`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `Nike`.`Payment` (
  `payment_id` INT NOT NULL,
  `payment_amount`INT NOT NULL,
  `customer_id` INT NOT NULL,
  PRIMARY KEY (`payment_id`),
  INDEX `fk_Payment_Customer1_idx` (`customer_id` ASC) VISIBLE,
  CONSTRAINT `fk_Payment_Customer1`
    FOREIGN KEY (`customer_id`)
    REFERENCES `Nike`.`Customer` (`customer_id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

For customer table

```
INSERT INTO customer
(customer_id,
first_name,
last_name,
phone_number)
VALUES
(1,
'Jon',
'Snow',
'08044599495');

INSERT INTO customer
(customer_id,
first_name,
last_name,
phone_number)
VALUES
(2,
'James',
'Bond',
'090436789');

INSERT INTO customer
(customer_id,
first_name,
last_name,
phone_number)
VALUES
(3,
'Okpa',
'Okonta',
'090136781');
```

For audience table

```
INSERT INTO audience
(audience_id,
audience_type)
VALUES
(1,
'Male');

INSERT INTO audience
(audience_id,
audience_type)
```

```
VALUES
(2,
'Female');
INSERT INTO audience
(audience_id,
audience_type)
VALUES
(3,
'Kids');
```

For shoes table

```
INSERT INTO shoes
(shoes_id,
shoe_name, audience_id, customer_id)
VALUES
(1,
'Air',
1,
1);


INSERT INTO shoes
(shoes_id,
shoe_name, audience_id, customer_id)
VALUES
(2,
'Jordan',
2,
3);

INSERT INTO shoes
(shoes_id,
shoe_name, audience_id, customer_id)
VALUES
(3,
'React',
3,
2);
```

For features table

```
INSERT INTO features
(feature_id,
```

```
feature_type)
VALUES
(1,
'running');

INSERT INTO features
(feature_id,
feature_type)
VALUES
(2,
'basketball');

INSERT INTO features
(feature_id,
feature_type)
VALUES
(3,
'soccer');
```

For city table

```
INSERT INTO city
(city_id,
city_name, customer_id)
VALUES
(1,
'Dallas', 1);

INSERT INTO city
(city_id,
city_name, customer_id)
VALUES
(2,
'Rexburg', 2);

INSERT INTO city
(city_id,
city_name, customer_id)
VALUES
(3,
'Rexburg', 2);
```

For shoes_has_feature table

```
INSERT INTO shoes_has_feature
(shoes_id, feature_id)
VALUES
(1,1);

INSERT INTO shoes_has_feature
(shoes_id, feature_id)
VALUES
(2,2);


INSERT INTO shoes_has_feature
(shoes_id, feature_id)
VALUES
(3,3);
```

For payment table

```
INSERT INTO payment
(customer_id, payment_id,
payment_amount)
VALUES
(1, 1, 180);

INSERT INTO payment
(payment_id,
payment_amount, customer_id)
VALUES
(2,
200, 2);

INSERT INTO payment
(payment_id,
payment_amount, customer_id)
VALUES
(3,
140, 3);
```

For customer_has_audience table

```
INSERT INTO customer_has_audience
(customer_id, audience_id)
VALUES
(1,1);
```

```
INSERT INTO customer_has_audience
(customer_id, audience_id)
VALUES
(2,1);

INSERT INTO customer_has_audience
(customer_id, audience_id)
VALUES
(3,1);
```

For address table

```
INSERT INTO address
(address_id, apt_no, street_name, postal_code, city_id, customer_id)
VALUES
(1, 304, "1st Avenue", 75001, 1, 1);

INSERT INTO address
(address_id, apt_no, street_name, postal_code, city_id, customer_id)
VALUES
(2, 3, "2nd Avenue", 75001, 2, 2);

INSERT INTO address
(address_id, apt_no, street_name, postal_code, city_id, customer_id)
VALUES
(3, 101, "3rd Avenue", 90001, 3, 3);
```