**THEATER**

**G31 นั่งดีๆนะ**

**62130500001 กณวรรธน์ มะมุด**

**62130500092 สมิตา พินิจสุพล**

**62130500099 สุพิชญา คณารักษ์สันติ**

|  |  |  |  |
| --- | --- | --- | --- |
| **รหัสนักศึกษา**  **หัวข้อที่รับผิดชอบ** | **62130500001** | **62130500092** | **62130500099** |
| **Insert data** | 100% | - | - |
| **Create table** | - | 50% | 50% |
| **Design Database** | รับผิดชอบร่วมกัน | | |
| **Report** | รับผิดชอบร่วมกัน | | |

**Table of Contents**

เรื่อง หน้า

Business Requirements/Features/Benefits 1

Relational Design 3

SQL statements 4

* Description
* SQL statements
* Query result

Data Dictionary 7

DDL script for creating the database 14

Export Data in the format of INSERT statements 22

**Business Requirements/ Features**

1. User แบ่งเป็น 2 Role ได้แก่ ลูกค้า และ พนักงานโดย

ฝั่งลูกค้า

* + สามารถจองตั๋ว เลือกที่นั่ง เลือกหนังที่ต้องการรับชม เลือกรอบหนัง และแสดงโรงที่ต้องไปรับชม
  + เช็คหนังในอนาคตที่กำลังจะเข้าฉายและเช็คหนังที่ฉายอยู่ในตอนนี้
  + สะสมคะแนนแลกรับรางวัล/ส่วนลดโรงหนัง

ฝั่งพนักงาน

* + สามารถเช็คได้ว่าที่นั่งแต่ละที่สามารถจองได้หรือไม่ หากยังไม่ถูกจองสามารถจองที่นั่งพร้อมเช็ครายละเอียดหนังได้ว่า อยู่โรงไหน, วันและรอบฉาย

1. เมื่อลูกค้าเข้าใช้งานแอปพลิเคชันจะสามารถเช็ค โรงหนังที่เข้าชมและรอบหนังที่กำลังเข้าฉายได้
2. ถ้าลูกค้าต้องการจองตั๋วหนังด้วยตนเอง ลูกค้าต้องทำการสมัครสมาชิกก่อนเป็นอันดับแรกจึงจะสามารถเข้าจองได้
3. ลูกค้าสามารถเลือกประเภทสมาชิกได้ โดยสามารถเลือกระหว่างประเภทธรรมดา หรือแบบที่ อัปเกรดเป็นประเภท VIP โดยสมาชิกทั้ง 2 ประเภทสามารถรับส่วนลด แต่ VIP จะต้องเสียค่าสมัครสมาชิกเพิ่มจากธรรมดา
4. เมื่อลูกค้าสมัครสมาชิกและล็อคอินเข้าใช้งานแล้ว ลูกค้าจะสามารถเลือกหนังพร้อมเลือกเวลาฉายที่ลูกค้าต้องการรับชม
5. หลังจากนั้นลูกค้าต้องเลือกประเภทที่นั่ง
6. ก่อนที่ลูกค้าจะจองหนังที่ต้องการรับชม ลูกค้าสามารถเช็คคะแนนและส่วนลด เพื่อนำไปใช้ในการลดราคาหรือใช้ในการแลกรับรางวัลได้
7. เมื่อลูกค้าจองแล้ว ลูกค้าจะต้องจ่ายเงินผ่านทางออนไลน์เพื่อที่จะจองที่ในโรงหนัง
8. เมื่อลูกค้าซื้อบัตรภาพยนตร์เสร็จแล้ว ระบบจะบันทึกแต้มที่ได้รับ
9. สำหรับลูกค้าที่ไม่ได้สมัครสมาชิกผ่านแอป พนักงานจะสามารถจองหนังให้กับลูกค้าได้
10. เมื่อลูกค้าจองผ่านพนักงาน ลูกค้านั้นจะไม่ได้รับส่วนลด แต่สามารถดูหนังได้ตามปกติ

**Benefits**

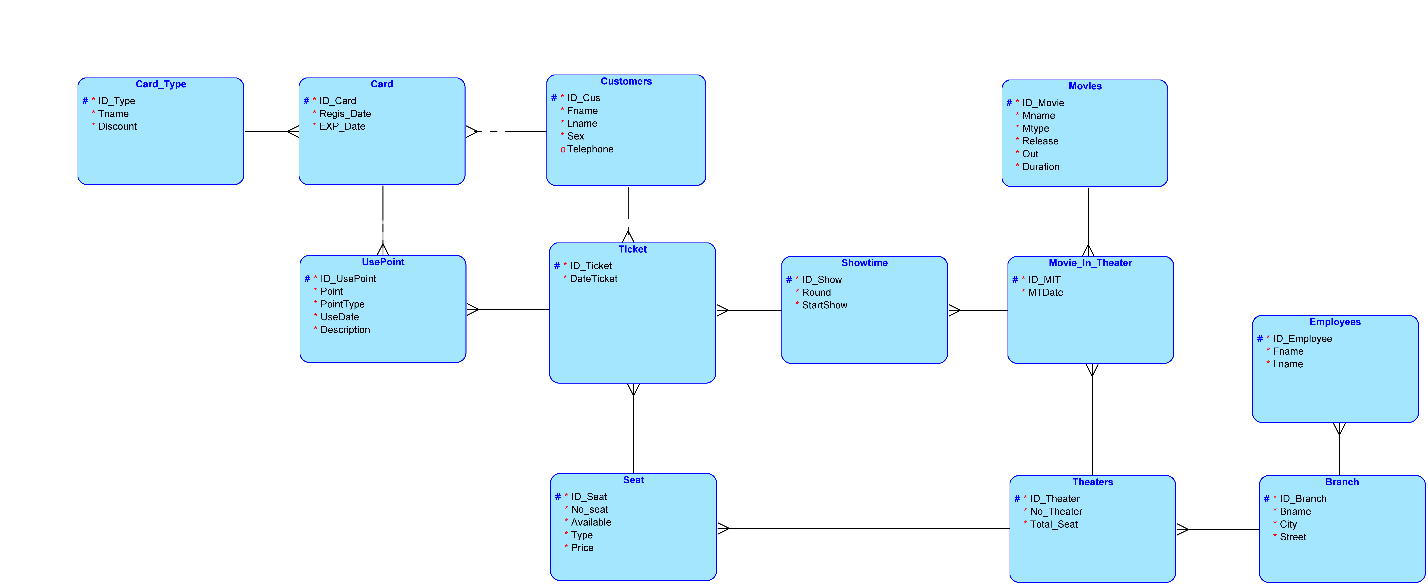
ผู้ใช้ได้อะไรจากระบบ

1. ได้ความสะดวกสบายจากการที่ไม่ต้องไปจองตั๋วหน้าโรงหนังและต่อคิวเป็นระยะเวลานาน
2. ทำให้ลดความหนาแน่นจากการจองหน้าโรงหนัง ทำให้สามารถใช้มาตรการ social distance ได้
3. สามารถได้รับส่วนลดจากการสมัครสมาชิก (ทั้งแบบธรรมดาและVIP) ไม่ว่าจะสมัครสมาชิกจากช่องทางไหนก็ตาม
4. ทำให้ลดการจ้างพนักงานที่ทำหน้าที่รับลูกค้าหน้าโรงหนังได้
5. สามารถเก็บสะสมแต้มเพื่อรับส่วนลดในการดูหนังครั้งต่อไปได้

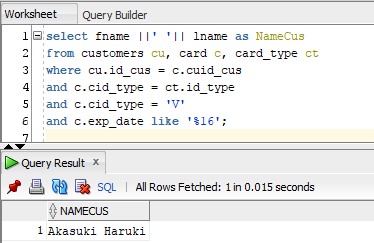
ระบบได้อะไรจากผู้ใช้

1. ระบบจะทราบรายการหนังที่ลูกค้านิยมเข้าชมเพื่อเพิ่มโรงฉายให้มากขึ้นเพื่อรองรับกับความต้องการของผู้ชม
2. ระบบจะทราบประเภทส่วนใหญ่ของลูกค้าที่เข้าใช้บริการในโรงหนัง

**Relational Design**

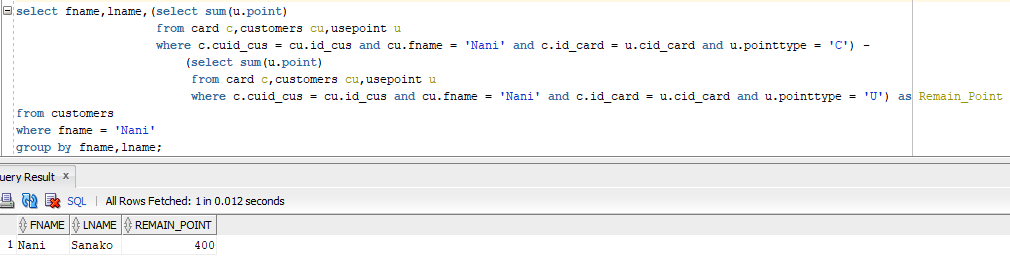
****

**SQL statements**

1.

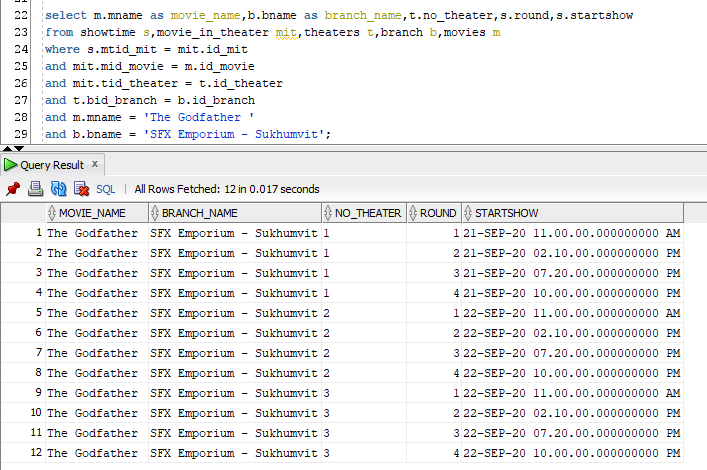
● Description

ต้องการทราบชื่อ – นามสกุลลูกค้าที่บัตรสมาชิกเป็นประเภท VIP และวันหมดอายุของบัตรสมาชิก คือ ปี 2016

2.

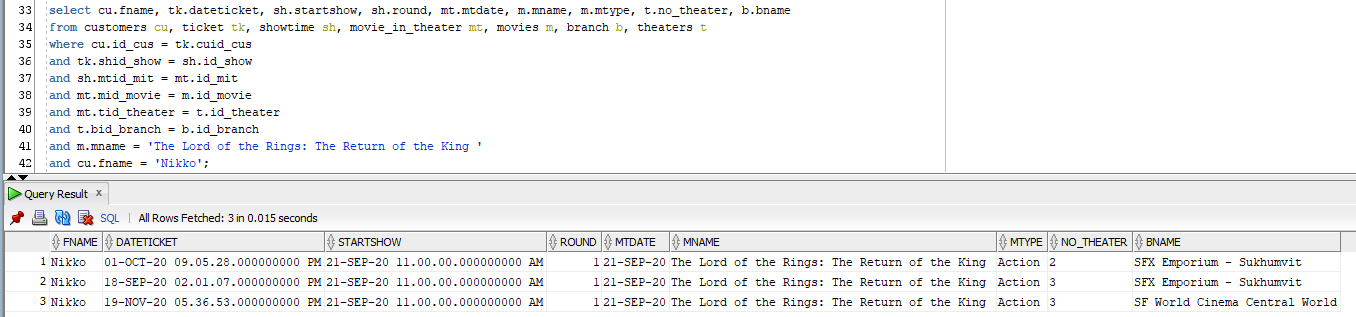
● Description

ต้องการทราบชื่อ – นามสกุล และคะแนนคงเหลือของลูกค้าที่มีชื่อว่า ‘Nani’

3. 

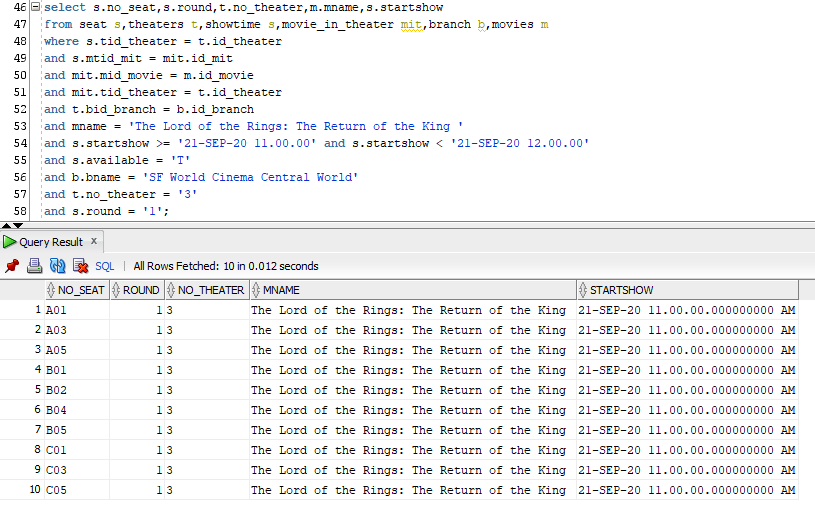
● Description

ต้องการทราบชื่อหนัง, สาขาที่มีหนังที่เลือกไว้, หมายเลขโรงภาพยนตร์, หมายเลขรอบที่เท่าไหร่และเวลาที่หนังฉาย

4. 

● Description

เช็คข้อมูลประวัติทั้งหมดของตั๋วหนังของลูกค้าที่ระบุ ว่าดูเรื่องไหน วันไหน และรอบไหนบ้าง

5. 

● Description

**หาที่นั่งที่ว่างจากภาพยนตร์ที่เลือกในโรงและรอบที่ระบุนั้น ๆ**

**Data Dictionary**

Table: branch

Description: ข้อมูลเกี่ยวกับสาขา

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Description** | **Data Type** | **Constraint** | **Referenced Table** |
| 1 | id\_branch | รหัสสาขา | VARCHAR2(6) | PK |  |
| 2 | bname | ชื่อสาขา | VARCHAR2(50) | NN |  |
| 3 | city | เมืองของสาขา | VARCHAR2(50) | NN |  |
| 4 | street | ถนนของสาขา | VARCHAR2(50) | NN |  |

Table: card

Description: ข้อมูลเกี่ยวกับบัตรสมาชิก

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Description** | **Data Type** | **Constraint** | **Referenced Table** |
| 1 | id\_card | รหัสบัตรสมาชิก | VARCHAR2(10) | PK |  |
| 2 | regis\_date | วันที่สมัครสมาชิก | DATE | NN |  |
| 3 | exp\_date | วันหมดอายุของบัตรสมาชิก | DATE | NN |  |
| 4 | CuID\_cus | รหัสประจำตัวลูกค้า | VARCHAR2(13) | FK | customers |
| 5 | CID\_type | รหัสประเภทบัตรสมาชิก | VARCHAR2(1) | FK | card\_type |

Table: card\_type

Description: ข้อมูลเกี่ยวกับประเภทของบัตรสมาชิก

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Description** | **Data Type** | **Constraint** | **Referenced Table** |
| 1 | id\_type | รหัสประเภทบัตรสมาชิก | VARCHAR2(1) | PK |  |
| 2 | tname | ชื่อประเภทบัตรสมาชิก | VARCHAR2(10) | NN |  |
| 3 | discount | ส่วนลดราคา | NUMBER(3) | NN,  CK(discount in (‘0.1’, '0.3')) |  |

Table: customers

Description: ข้อมูลเกี่ยวกับลูกค้า

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Description** | **Data Type** | **Constraint** | **Referenced Table** |
| 1 | id\_cus | รหัสประจำตัวลูกค้า | VARCHAR2(13) | PK |  |
| 2 | fname | ชื่อจริง | VARCHAR2(100) | NN |  |
| 3 | lname | นามสกุล | VARCHAR2(100) | NN |  |
| 4 | sex | เพศ | VARCHAR2(1) | NN, CK(sex in('M', 'F', 'O')) |  |
| 5 | telephone | เบอร์โทรศัพท์ | VARCHAR2(15) |  |  |

Table: employees

Description: ข้อมูลเกี่ยวกับพนักงาน

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Description** | **Data Type** | **Constraint** | **Referenced Table** |
| 1 | id\_employee | รหัสประจำตัวพนักงาน | VARCHAR2(10) | PK |  |
| 2 | fname | ชื่อจริง | VARCHAR2(50) | NN |  |
| 3 | lname | นามสกุล | VARCHAR2(50) | NN |  |
| 4 | BID\_branch | รหัสสาขา | VARCHAR2(6) | FK | branch |

Table: movie\_in\_theater

Description: ข้อมูลเกี่ยวกับภาพยนตร์ที่จะฉายในโรงภาพยนตร์

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Description** | **Data Type** | **Constraint** | **Referenced Table** |
| 1 | id\_mit | รหัสภาพยนตร์ที่จะฉายในโรงภาพยนตร์ | VARCHAR2(5) | PK |  |
| 2 | MTDate | วันที่ภาพยนตร์ฉายในวันนั้น ๆ | DATE | NN |  |
| 3 | MID\_movie | รหัสภาพยนตร์ | VARCHAR2(6) | FK | movies |
| 4 | TID\_theater | รหัสโรงภาพยนตร์ | VARCHAR2(6) | FK | theaters |

Table: movies

Description: ข้อมูลเกี่ยวกับภาพยนตร์

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Description** | **Data Type** | **Constraint** | **Referenced Table** |
| 1 | id\_movie | รหัสภาพยนตร์ | VARCHAR2(6) | PK |  |
| 2 | mname | ชื่อภาพยนตร์ | VARCHAR2(6) | NN |  |
| 3 | mtype | ประเภทภาพยนตร์ | VARCHAR2(6) | NN, CK(mtype in ('Horror', 'Romantic', 'Comedy', 'Drama', 'Action', 'Crime', 'Cartoon')) |  |
| 4 | release | วันที่ฉาย | DATE | NN |  |
| 5 | out | วันที่ออก | DATE | NN |  |
| 6 | duration | ความยาวของภาพยนตร์ | NUMBER(3) | NN |  |

Table: seat

Description: ข้อมูลเกี่ยวกับที่นั่ง

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Description** | **Data Type** | **Constraint** | **Referenced Table** |
| 1 | id\_seat | รหัสที่นั่ง | NUMBER(5) | PK |  |
| 2 | no\_seat | หมายเลขที่นั่ง | VARCHAR2(5) | NN |  |
| 3 | available | สถานะที่นั่ง | VARCHAR2(1) | NN, (available in ('T', 'F')) |  |
| 4 | type | ประเภทที่นั่ง | VARCHAR2(50) | NN, CK(type in ('RegularSeat', 'HoneyMoon', 'VIP Seat')) |  |
| 5 | price | ราคาที่นั่ง | NUMBER(4) | NN, CK(price in ('300', '600', '900')) |  |
| 6 | TID\_theater | รหัสโรงภาพยนตร์ | VARCHAR2(6) | FK | theaters |

Table: showtime

Description: ข้อมูลเกี่ยวกับรอบฉายของภาพยนตร์

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Description** | **Data Type** | **Constraint** | **Referenced Table** |
| 1 | id\_show | รหัสรอบฉาย | VARCHAR2(6) | PK |  |
| 2 | round | รอบที่ฉาย | NUMBER(1) | NN |  |
| 3 | startShow | วันและเวลาที่ฉาย | TIMESTAMP | NN |  |
| 4 | MTID\_MIT | รหัสภาพยนตร์ที่จะฉายในโรงภาพยนตร์ | VARCHAR2(5) | FK | movie\_in\_theater |

Table: theaters

Description: ข้อมูลเกี่ยวกับโรงภาพยนตร์

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Description** | **Data Type** | **Constraint** | **Referenced Table** |
| 1 | id\_theater | รหัสโรงภาพยนตร์ | VARCHAR2(6) | PK |  |
| 2 | no\_theater | หมายเลขโรงภาพยนตร์ | VARCHAR2(2) | NN |  |
| 3 | total\_seat | จำนวนที่นั่งทั้งหมดในโรงภาพยนตร์ | NUMBER(3) | NN, CK(total\_seat >= 0) |  |
| 4 | BID\_branch | รหัสสาขา | VARCHAR2(6) | FK | branch |

Table: ticket

Description: ข้อมูลเกี่ยวกับตั๋วภาพยนตร์

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Description** | **Data Type** | **Constraint** | **Referenced Table** |
| 1 | id\_ticket | รหัสตั๋วภาพยนตร์ | VARCHAR2(6) | PK |  |
| 2 | dateTicket | วันที่และเวลาที่ออกตั๋วภาพยนตร์ | TIMESTAMP | NN |  |
| 3 | CuID\_cus | รหัสประจำตัวลูกค้า | VARCHAR2(13) | FK | customers |
| 4 | ShID\_show | รหัสรอบฉาย | VARCHAR2(6) | FK | showtime |
| 5 | SID\_seat | รหัสที่นั่ง | NUMBER(5) | FK | seat |

Table: usepoint

Description: ข้อมูลเกี่ยวกับการใช้คะแนนของบัตรสมาชิก

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Description** | **Data Type** | **Constraint** | **Referenced Table** |
| 1 | id\_usePoint | รหัสการใช้คะแนน | VARCHAR2(10) | PK |  |
| 2 | point | คะแนนต่อรายการ | NUMBER(6) | NN, CK(point >= 0) |  |
| 3 | pointType | ประเภทคะแนน | VARCHAR2(1) | NN, CK(pointType in('U', 'C')) |  |
| 4 | useDate | วันที่ใช้คะแนน | DATE | NN |  |
| 5 | description | คำอธิบายคะแนน | VARCHAR2(100) | NN |  |
| 6 | CID\_card | รหัสบัตรสมาชิก | VARCHAR2(10) | FK | card |
| 7 | TID\_ticket | รหัสตั๋วภาพยนตร์ | VARCHAR2(6) | FK | ticket |

**DDL script for creating the database**

DROP TABLE branch CASCADE CONSTRAINTS;

DROP TABLE card CASCADE CONSTRAINTS;

DROP TABLE card\_type CASCADE CONSTRAINTS;

DROP TABLE customers CASCADE CONSTRAINTS;

DROP TABLE employees CASCADE CONSTRAINTS;

DROP TABLE movie\_in\_theater CASCADE CONSTRAINTS;

DROP TABLE movies CASCADE CONSTRAINTS;

DROP TABLE seat CASCADE CONSTRAINTS;

DROP TABLE showtime CASCADE CONSTRAINTS;

DROP TABLE theaters CASCADE CONSTRAINTS;

DROP TABLE ticket CASCADE CONSTRAINTS;

DROP TABLE usepoint CASCADE CONSTRAINTS;

CREATE TABLE branch (

id\_branch VARCHAR2(6) NOT NULL,

bname VARCHAR2(50) NOT NULL,

city VARCHAR2(50) NOT NULL,

street VARCHAR2(50) NOT NULL

);

ALTER TABLE branch ADD CONSTRAINT branch\_pk PRIMARY KEY ( id\_branch );

CREATE TABLE card (

id\_card VARCHAR2(10) NOT NULL,

regis\_date DATE NOT NULL,

exp\_date DATE NOT NULL,

CuID\_cus VARCHAR2(13),

CID\_type VARCHAR2(1) NOT NULL

);

ALTER TABLE card ADD CONSTRAINT card\_pk PRIMARY KEY ( id\_card );

CREATE TABLE card\_type (

id\_type VARCHAR2(1) NOT NULL,

tname VARCHAR2(10) NOT NULL,

discount NUMBER(3,2) NOT NULL

);

ALTER TABLE card\_type ADD CONSTRAINT card\_type\_pk PRIMARY KEY ( id\_type );

CREATE TABLE customers (

id\_cus VARCHAR2(13) NOT NULL,

fname VARCHAR2(100) NOT NULL,

lname VARCHAR2(100) NOT NULL,

sex VARCHAR2(1) DEFAULT 'O' NOT NULL

constraint cu\_sex\_ck check(sex in('M','F','O')),

telephone VARCHAR2(15)

);

ALTER TABLE customers ADD CONSTRAINT customers\_pk PRIMARY KEY ( id\_cus );

CREATE TABLE employees (

id\_employee VARCHAR2(10) NOT NULL,

fname VARCHAR2(50) NOT NULL,

lname VARCHAR2(50) NOT NULL,

BID\_branch VARCHAR2(6) NOT NULL

);

ALTER TABLE employees ADD CONSTRAINT employees\_pk PRIMARY KEY ( id\_employee );

CREATE TABLE movie\_in\_theater (

id\_mit VARCHAR2(5) NOT NULL,

MTDate DATE NOT NULL,

MID\_movie VARCHAR2(6) NOT NULL,

TID\_theater VARCHAR2(6) NOT NULL

);

ALTER TABLE movie\_in\_theater ADD CONSTRAINT movie\_in\_theater\_pk PRIMARY KEY ( id\_mit );

CREATE TABLE movies (

id\_movie VARCHAR2(6) NOT NULL,

mname VARCHAR2(100) NOT NULL,

mtype VARCHAR2(50) NOT NULL

constraint movies\_mtype\_ck check(mtype in ('Horror', 'Romantic', 'Comedy', 'Drama', 'Action', 'Crime', 'Cartoon')),

release DATE NOT NULL,

out DATE NOT NULL,

duration NUMBER(3) NOT NULL

);

ALTER TABLE movies ADD CONSTRAINT movies\_pk PRIMARY KEY ( id\_movie );

CREATE TABLE seat (

id\_seat NUMBER(5) NOT NULL,

no\_seat VARCHAR2(5) NOT NULL,

available VARCHAR2(1) NOT NULL

constraint seat\_available\_ck check (available in ('T', 'F')),

type VARCHAR2(50) NOT NULL

constraint seat\_type\_ck check (type in ('RegularSeat', 'HoneyMoon', 'VIP Seat')),

price NUMBER(4) NOT NULL

constraint seat\_price\_ck check (price in ('300', '600', '900')),

TID\_theater VARCHAR2(6) NOT NULL

);

ALTER TABLE seat ADD CONSTRAINT seat\_pk PRIMARY KEY ( id\_seat );

CREATE TABLE showtime (

id\_show VARCHAR2(6) NOT NULL,

round NUMBER(1) NOT NULL,

startShow TIMESTAMP NOT NULL,

MTID\_MIT VARCHAR2(5) NOT NULL

);

ALTER TABLE showtime ADD CONSTRAINT showtime\_pk PRIMARY KEY ( id\_show );

CREATE TABLE theaters (

id\_theater VARCHAR2(6) NOT NULL,

no\_theater VARCHAR2(2) NOT NULL,

total\_seat NUMBER(3) NOT NULL

constraint theaters\_totalSeat\_ck check (total\_seat >= 0),

BID\_branch VARCHAR2(6) NOT NULL

);

ALTER TABLE theaters ADD CONSTRAINT theaters\_pk PRIMARY KEY ( id\_theater );

CREATE TABLE ticket (

id\_ticket VARCHAR2(6) NOT NULL,

dateTicket TIMESTAMP NOT NULL,

CuID\_cus VARCHAR2(13) NOT NULL,

ShID\_show VARCHAR2(6) NOT NULL,

SID\_seat NUMBER(5) NOT NULL

);

CREATE UNIQUE INDEX ticket\_\_idx

ON ticket (SID\_seat ASC);

ALTER TABLE ticket ADD CONSTRAINT ticket\_pk PRIMARY KEY ( id\_ticket );

CREATE TABLE usepoint (

id\_usePoint VARCHAR2(10)

CONSTRAINT nnc\_usepoint\_id\_card NOT NULL,

point NUMBER(6)

CONSTRAINT nnc\_usepoint\_point NOT NULL

constraint usepoint\_point\_ck check (point>=0),

pointType VARCHAR2(1)

CONSTRAINT nnc\_usepoint\_pointtype NOT NULL

constraint usepoint\_pointType\_ck check (pointType in('U','C')),

useDate DATE NOT NULL,

description VARCHAR2(100)

CONSTRAINT nnc\_usepoint\_description NOT NULL,

CID\_card VARCHAR2(10) NOT NULL,

TID\_ticket VARCHAR2(6) NOT NULL

);

ALTER TABLE usepoint ADD CONSTRAINT usepoint\_pk PRIMARY KEY ( id\_usePoint );

ALTER TABLE card

ADD CONSTRAINT card\_cidType\_fk FOREIGN KEY ( CID\_type )

REFERENCES card\_type ( id\_type );

ALTER TABLE card

ADD CONSTRAINT card\_customers\_fk FOREIGN KEY ( CuID\_cus )

REFERENCES customers ( id\_cus );

ALTER TABLE employees

ADD CONSTRAINT employees\_branch\_fk FOREIGN KEY ( BID\_branch )

REFERENCES branch ( id\_branch );

ALTER TABLE movie\_in\_theater

ADD CONSTRAINT movie\_in\_theater\_movies\_fk FOREIGN KEY ( MID\_movie )

REFERENCES movies ( id\_movie );

ALTER TABLE movie\_in\_theater

ADD CONSTRAINT movie\_in\_theater\_theaters\_fk FOREIGN KEY ( TID\_theater )

REFERENCES theaters ( id\_theater );

ALTER TABLE seat

ADD CONSTRAINT seat\_theaters\_fk FOREIGN KEY ( TID\_theater )

REFERENCES theaters ( id\_theater );

ALTER TABLE showtime

ADD CONSTRAINT showtime\_movie\_in\_theater\_fk FOREIGN KEY ( MTID\_MIT )

REFERENCES movie\_in\_theater ( id\_mit );

ALTER TABLE theaters

ADD CONSTRAINT theaters\_branch\_fk FOREIGN KEY ( BID\_branch )

REFERENCES branch ( id\_branch );

ALTER TABLE ticket

ADD CONSTRAINT ticket\_customers\_fk FOREIGN KEY ( CuID\_cus )

REFERENCES customers ( id\_cus );

ALTER TABLE ticket

ADD CONSTRAINT ticket\_showtime\_fk FOREIGN KEY ( ShID\_show )

REFERENCES showtime ( id\_show );

ALTER TABLE ticket

ADD CONSTRAINT ticket\_seat\_fk FOREIGN KEY ( SID\_seat )

REFERENCES seat ( id\_seat );

ALTER TABLE usepoint

ADD CONSTRAINT usepoint\_card\_fk FOREIGN KEY ( CID\_card )

REFERENCES card ( id\_card );

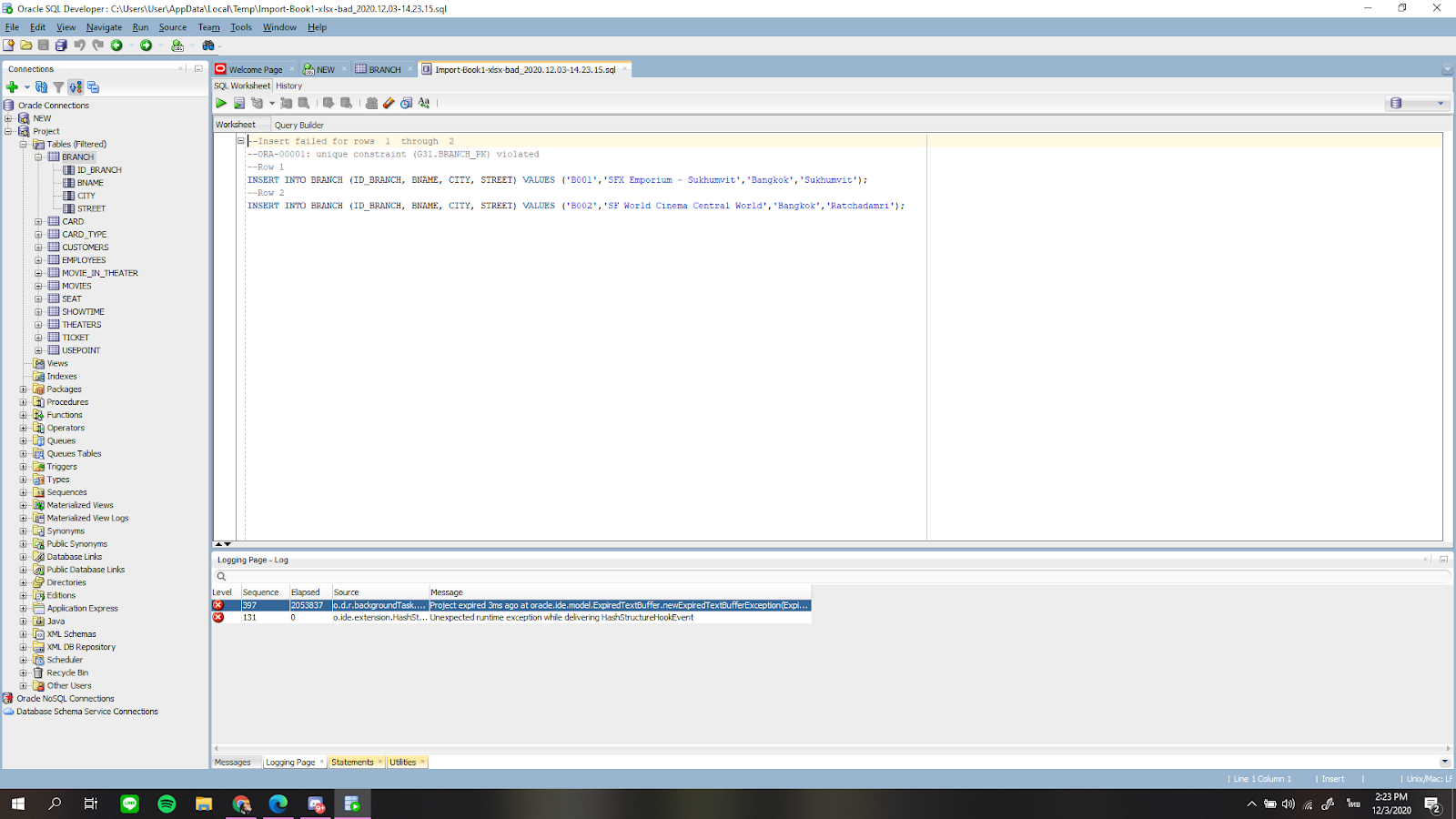
ALTER TABLE usepoint

ADD CONSTRAINT usepoint\_ticket\_fk FOREIGN KEY ( TID\_ticket )

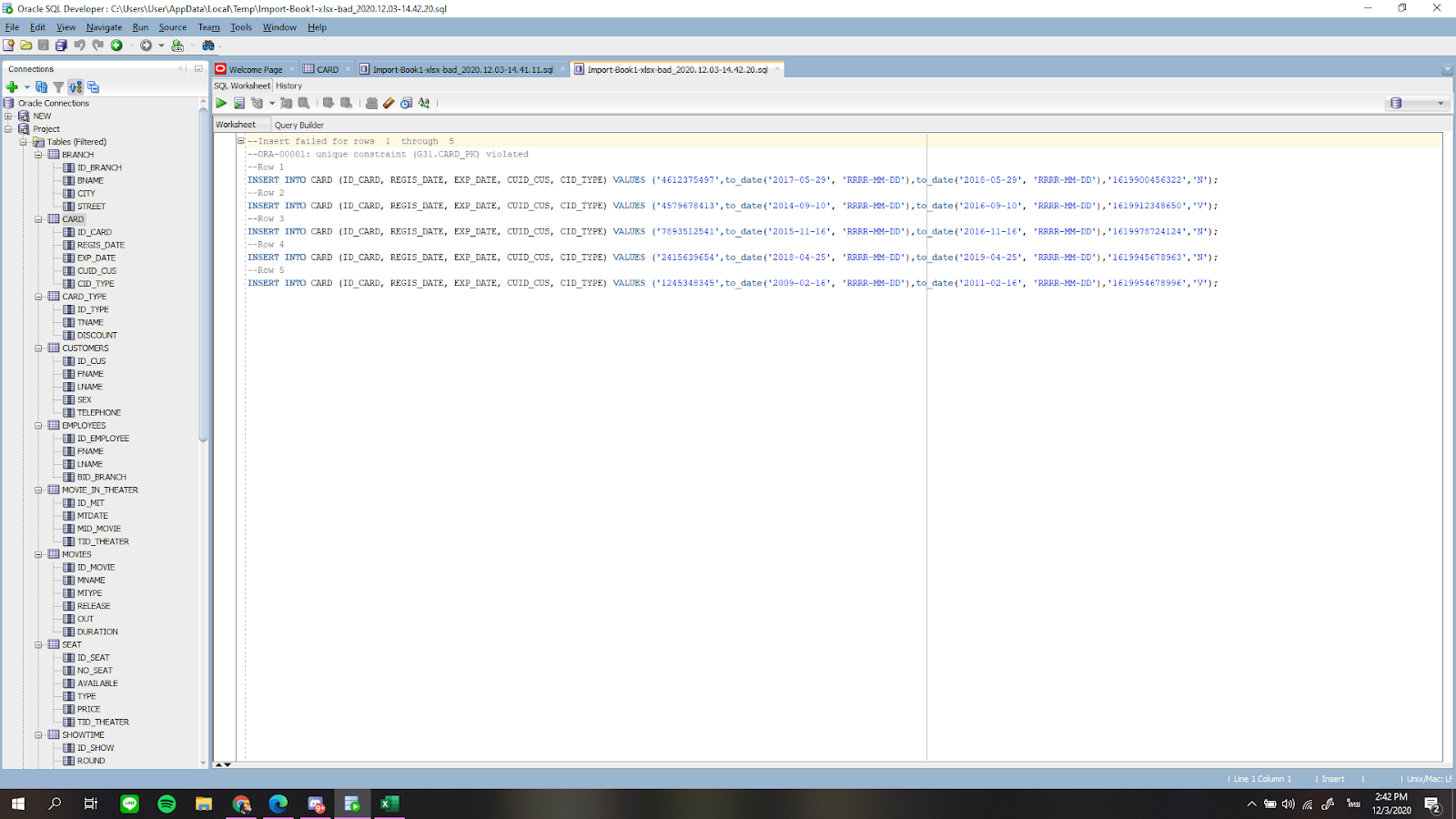
REFERENCES ticket ( id\_ticket );

**Export Data in the format of INSERT statements**

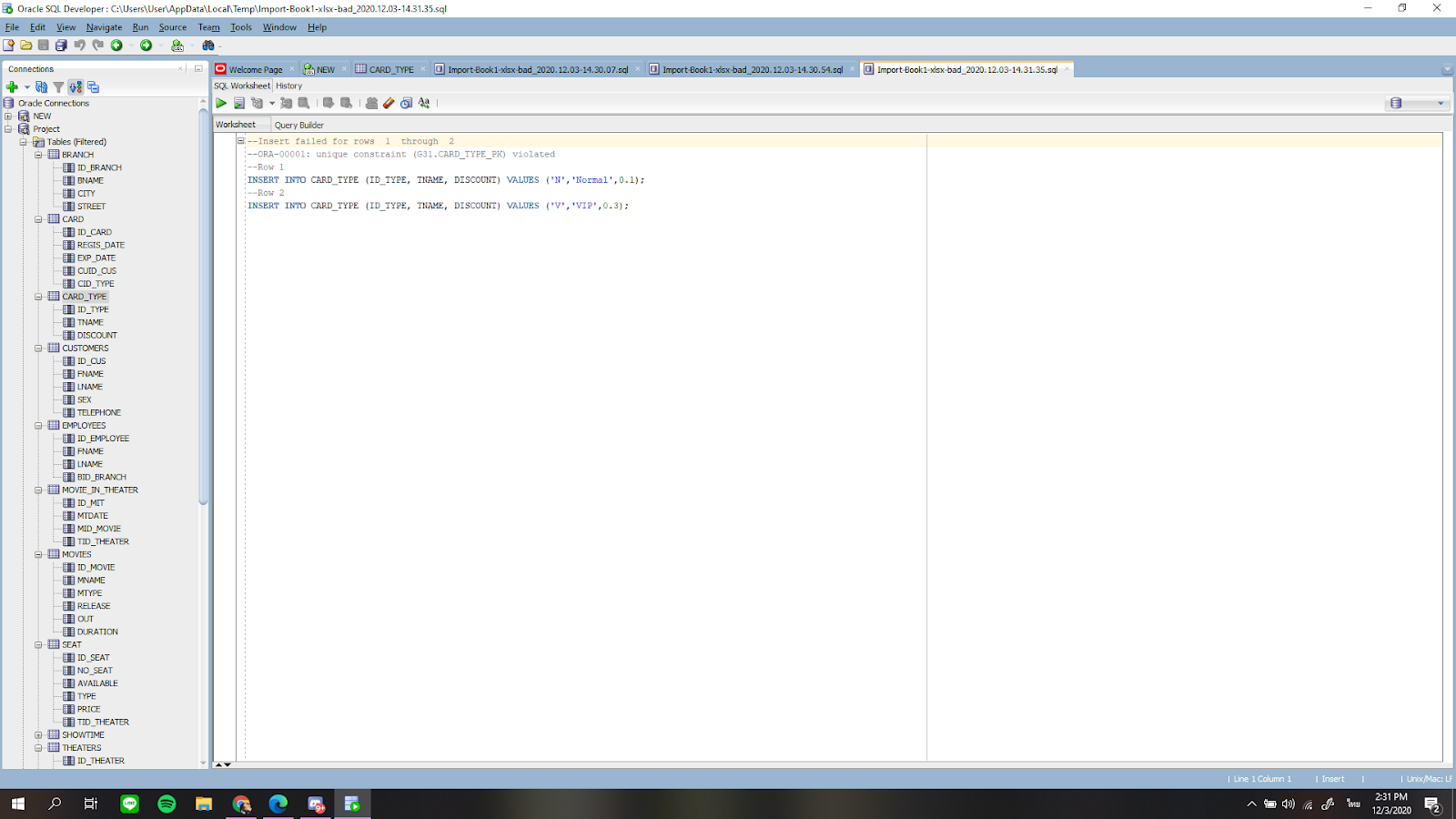
1. Table: branch

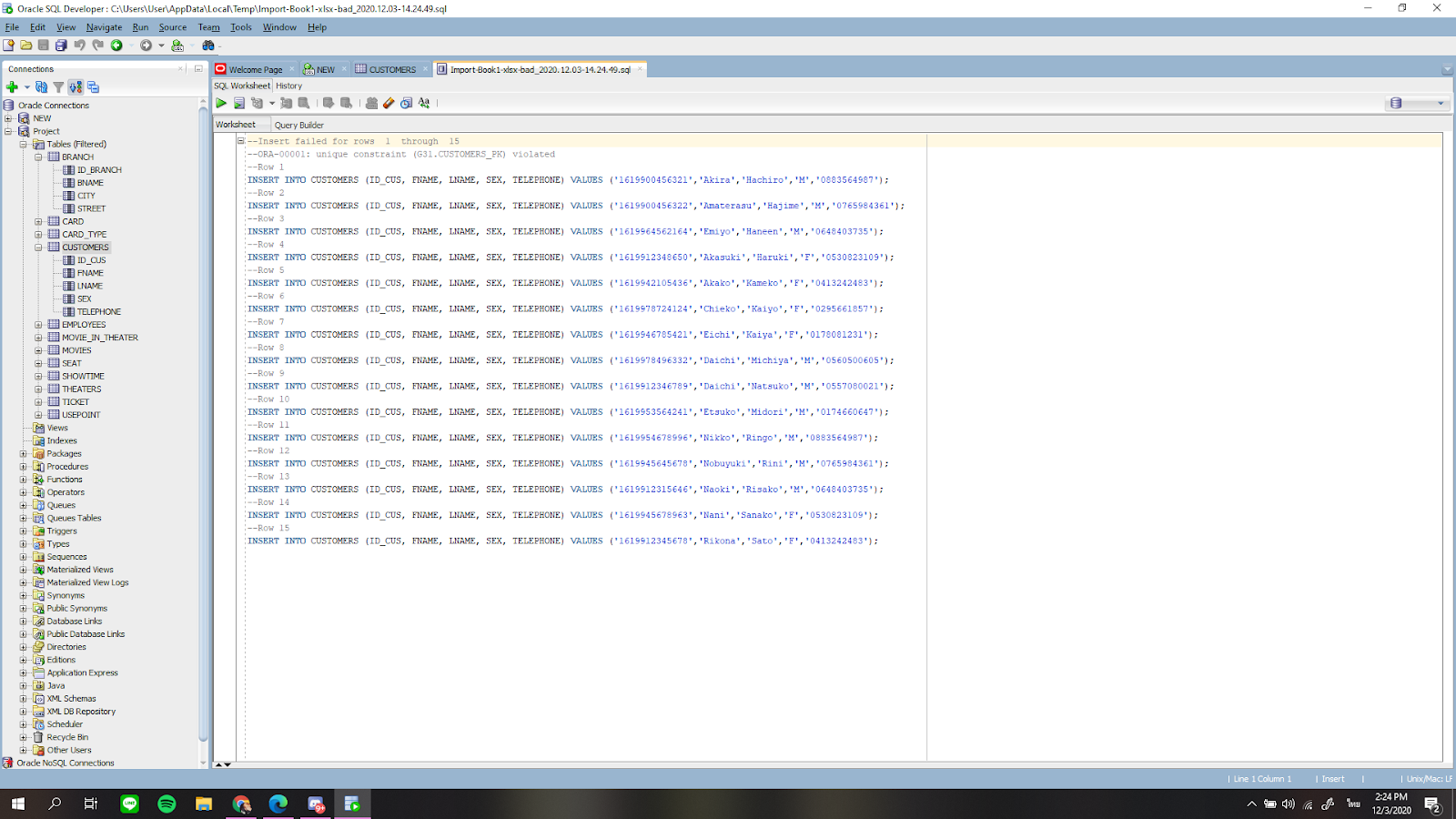


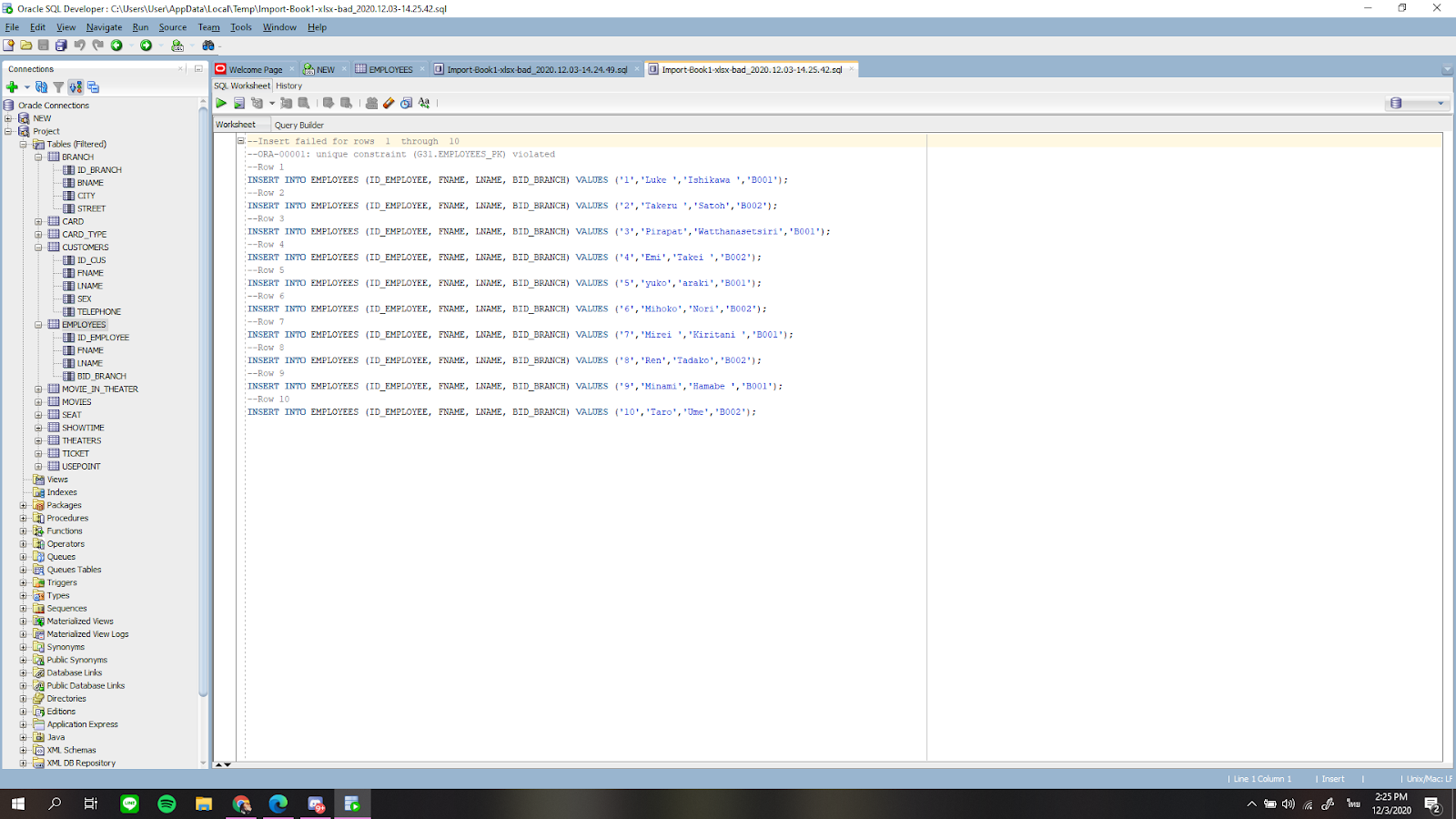
1. Table: card

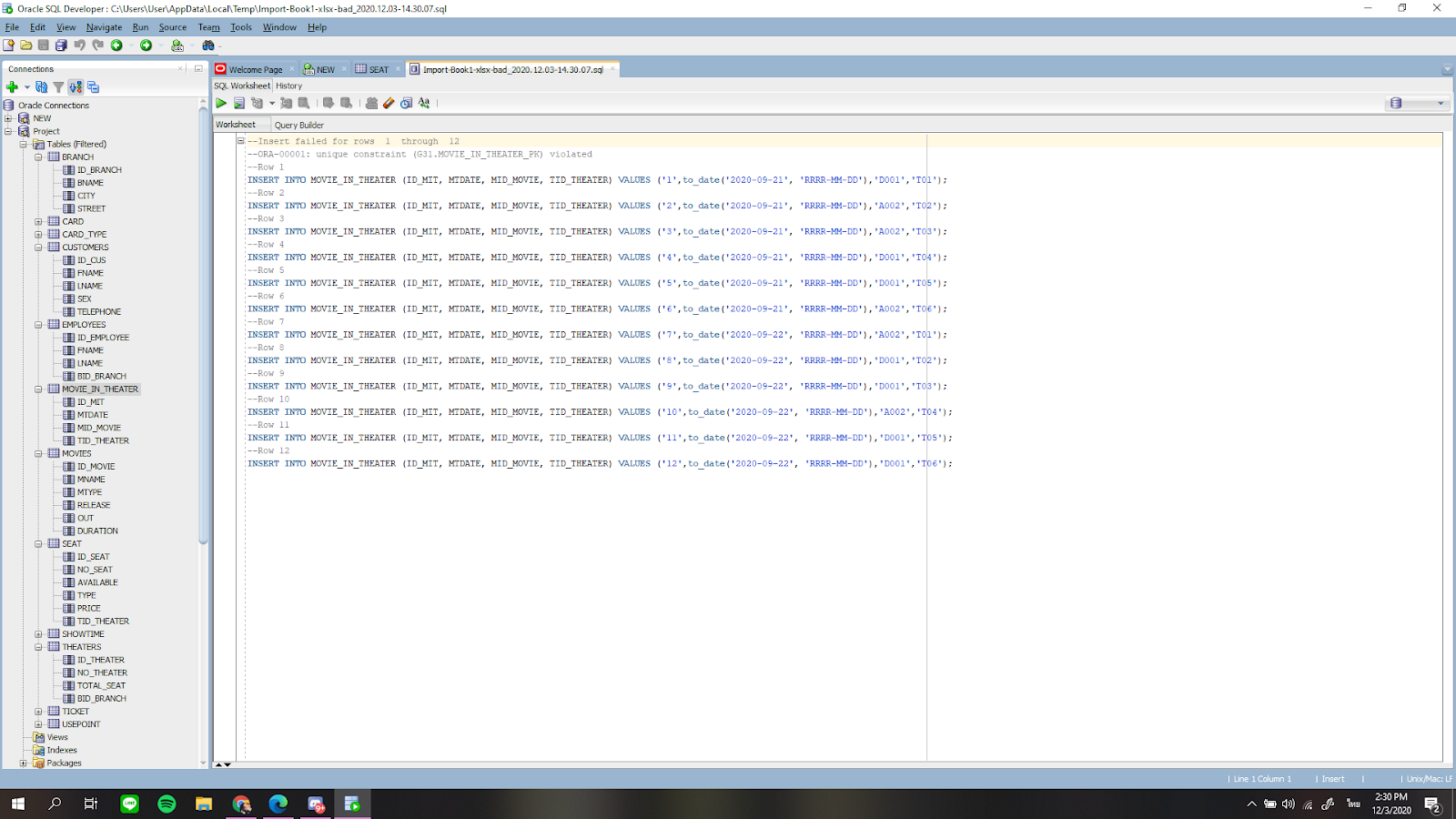
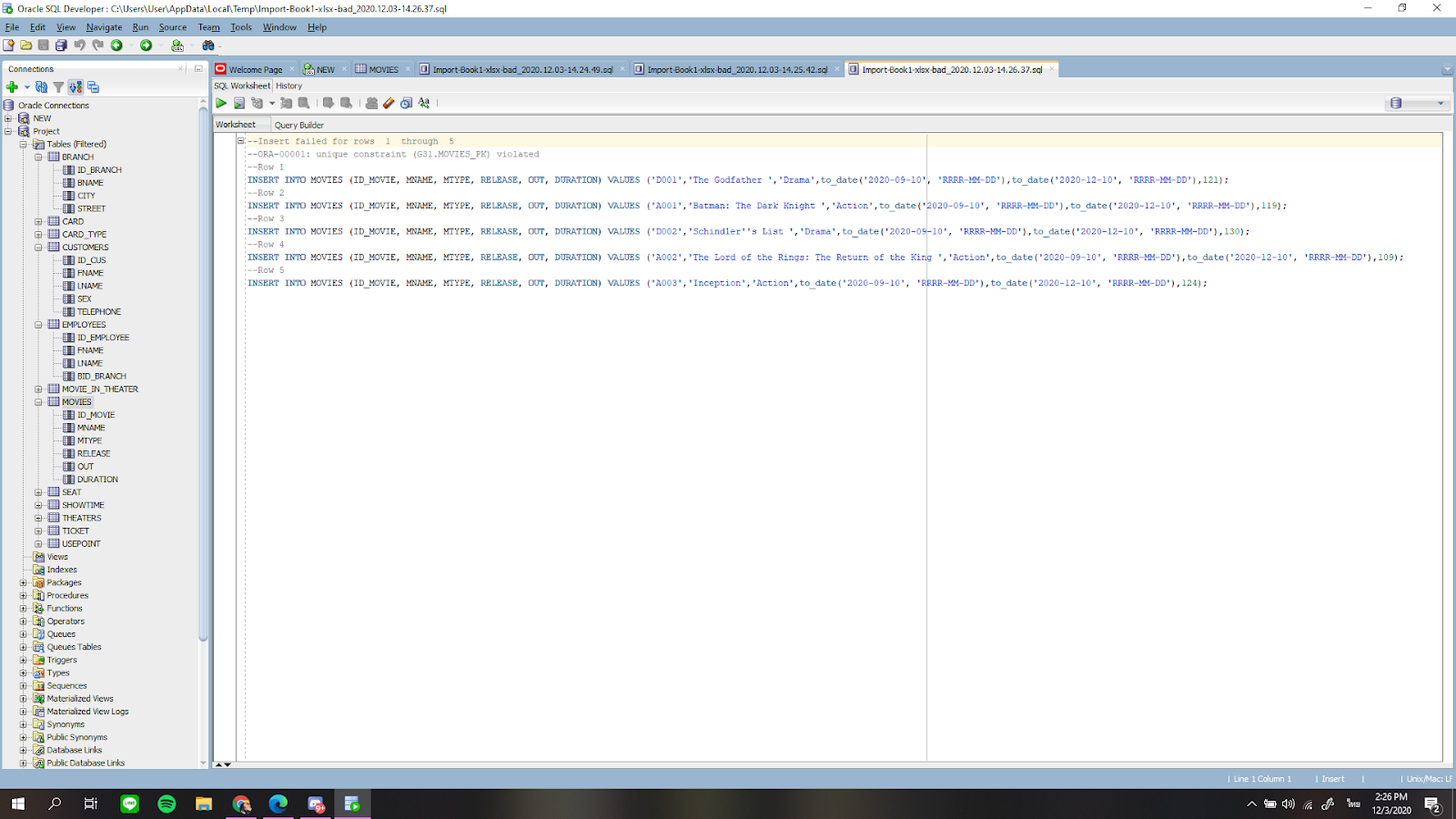
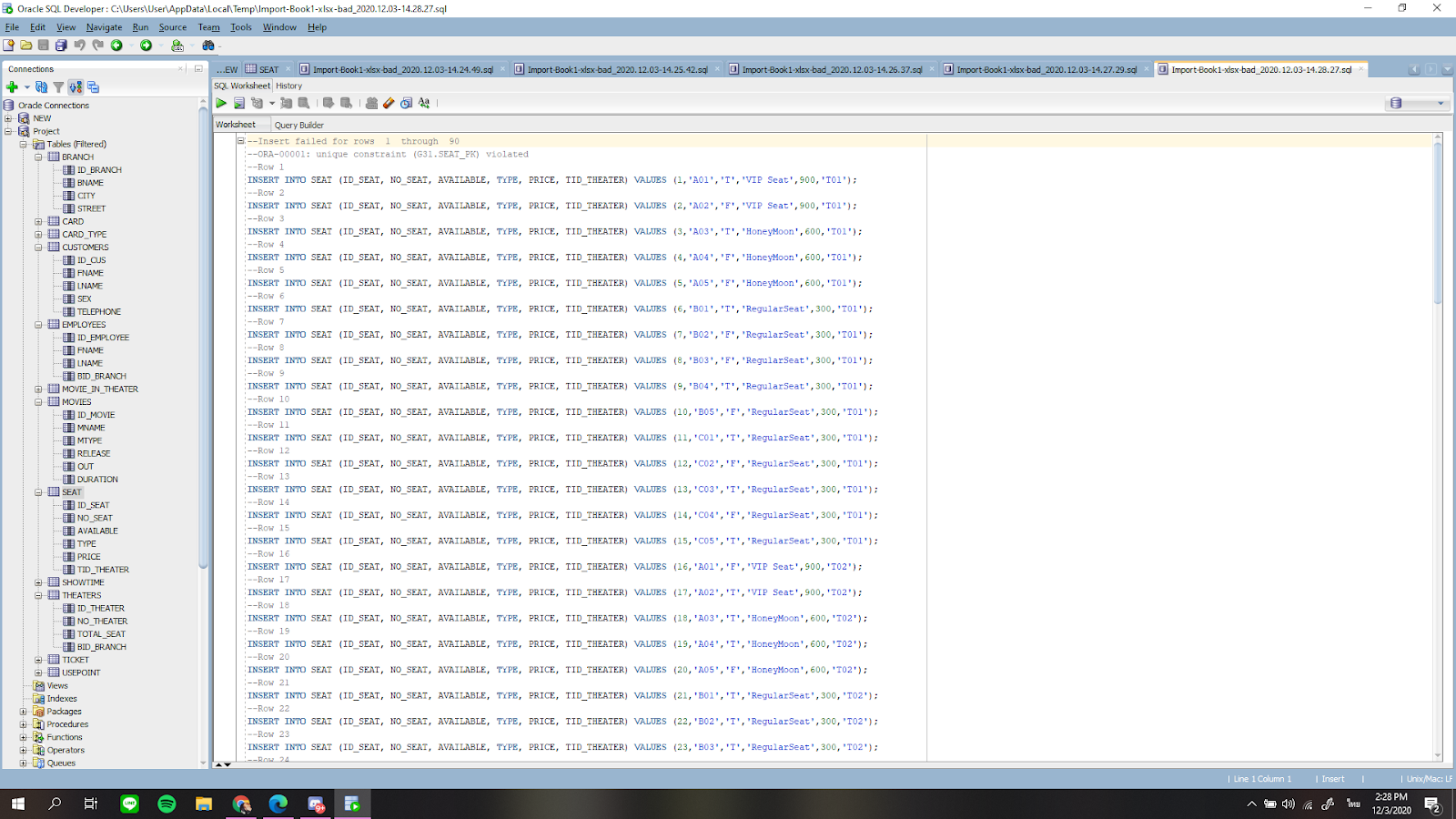
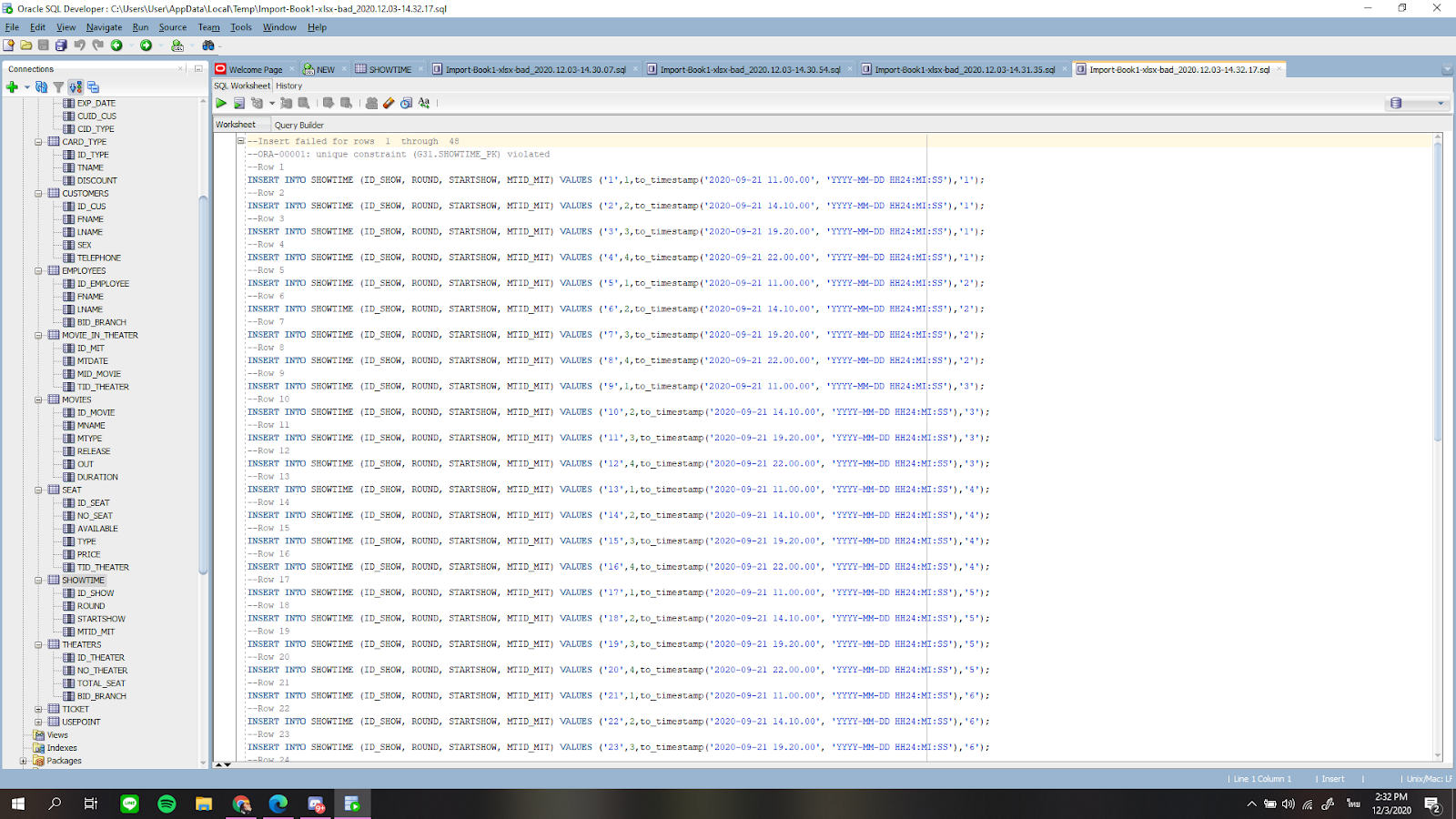
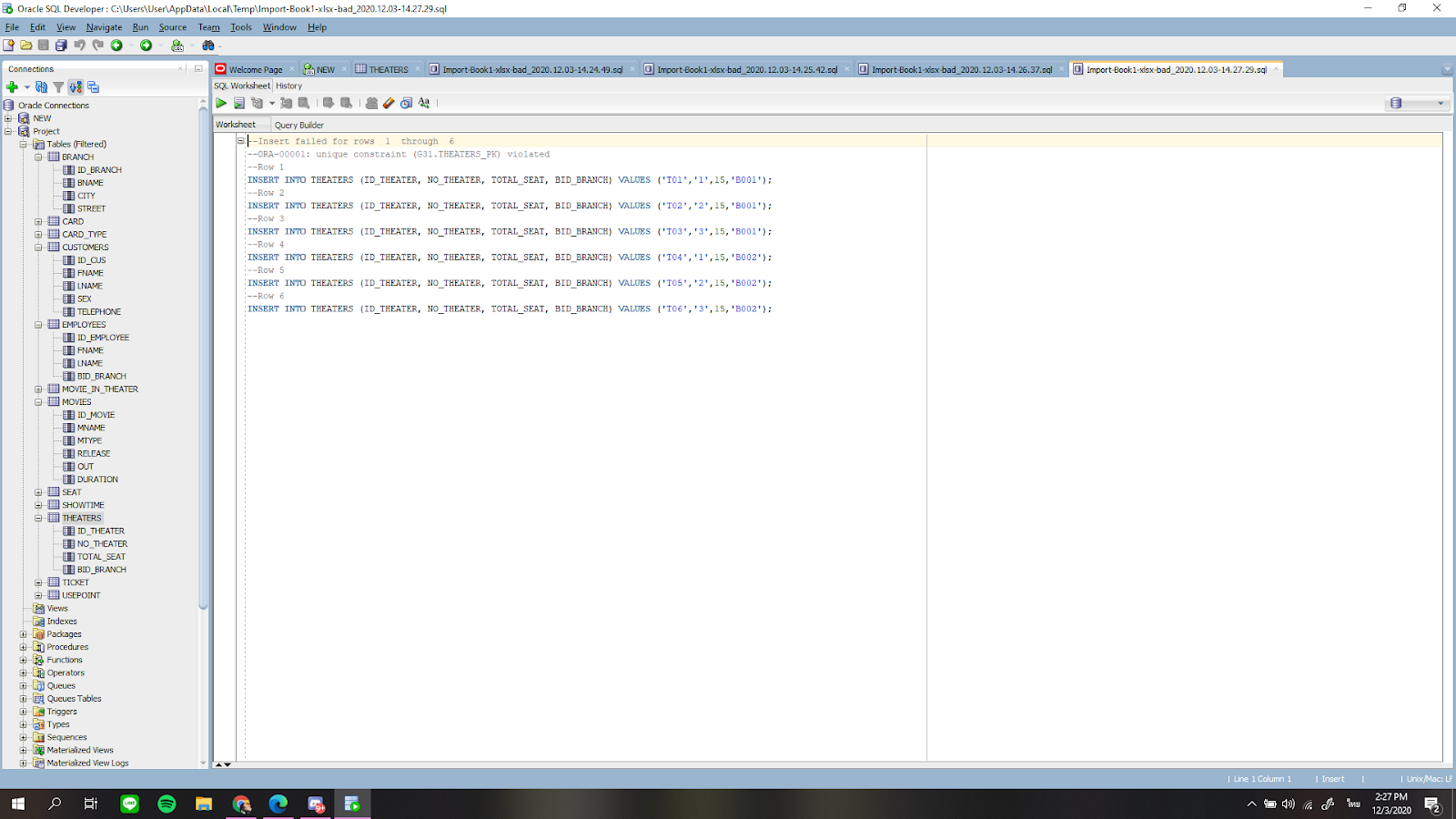
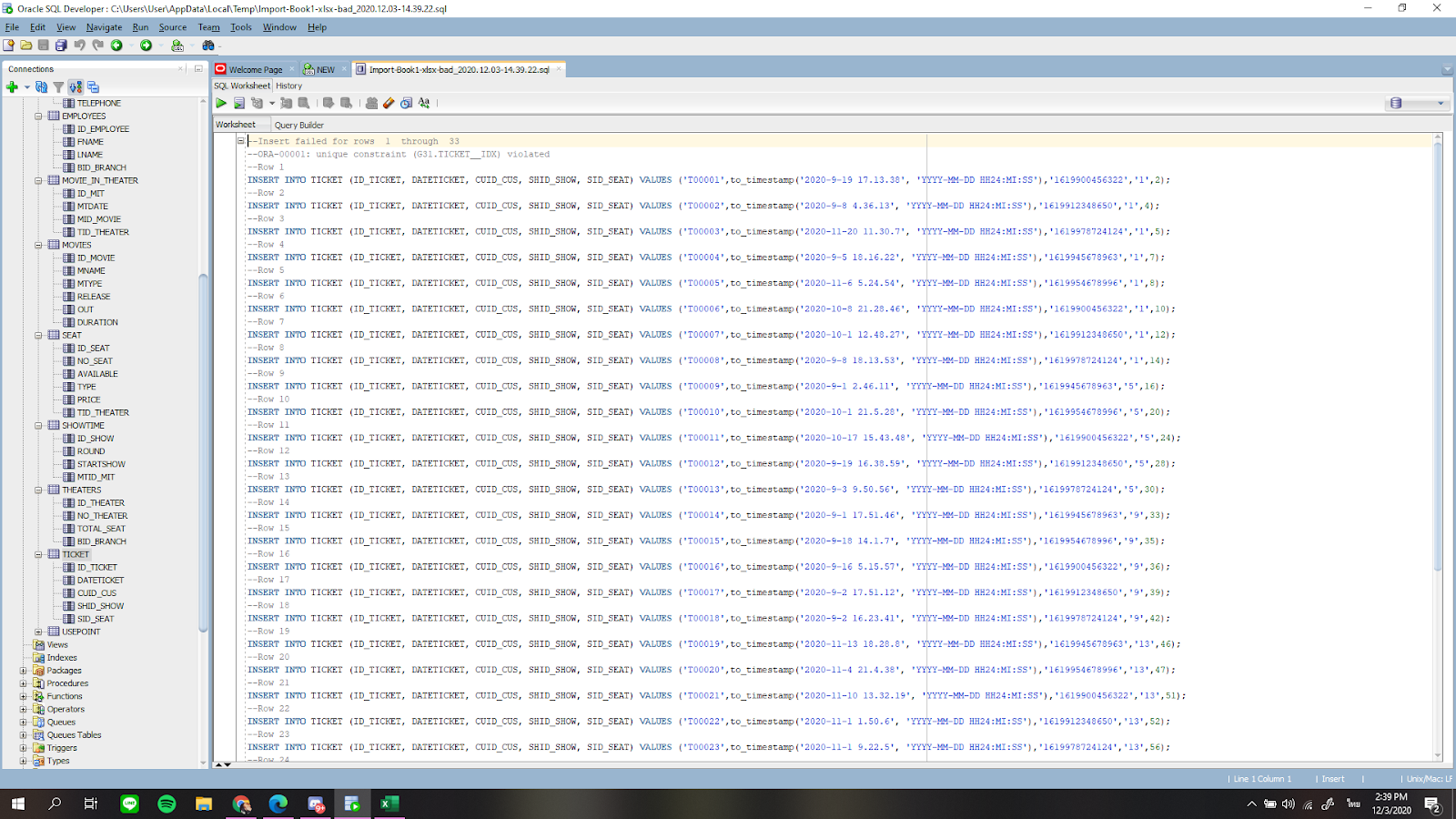
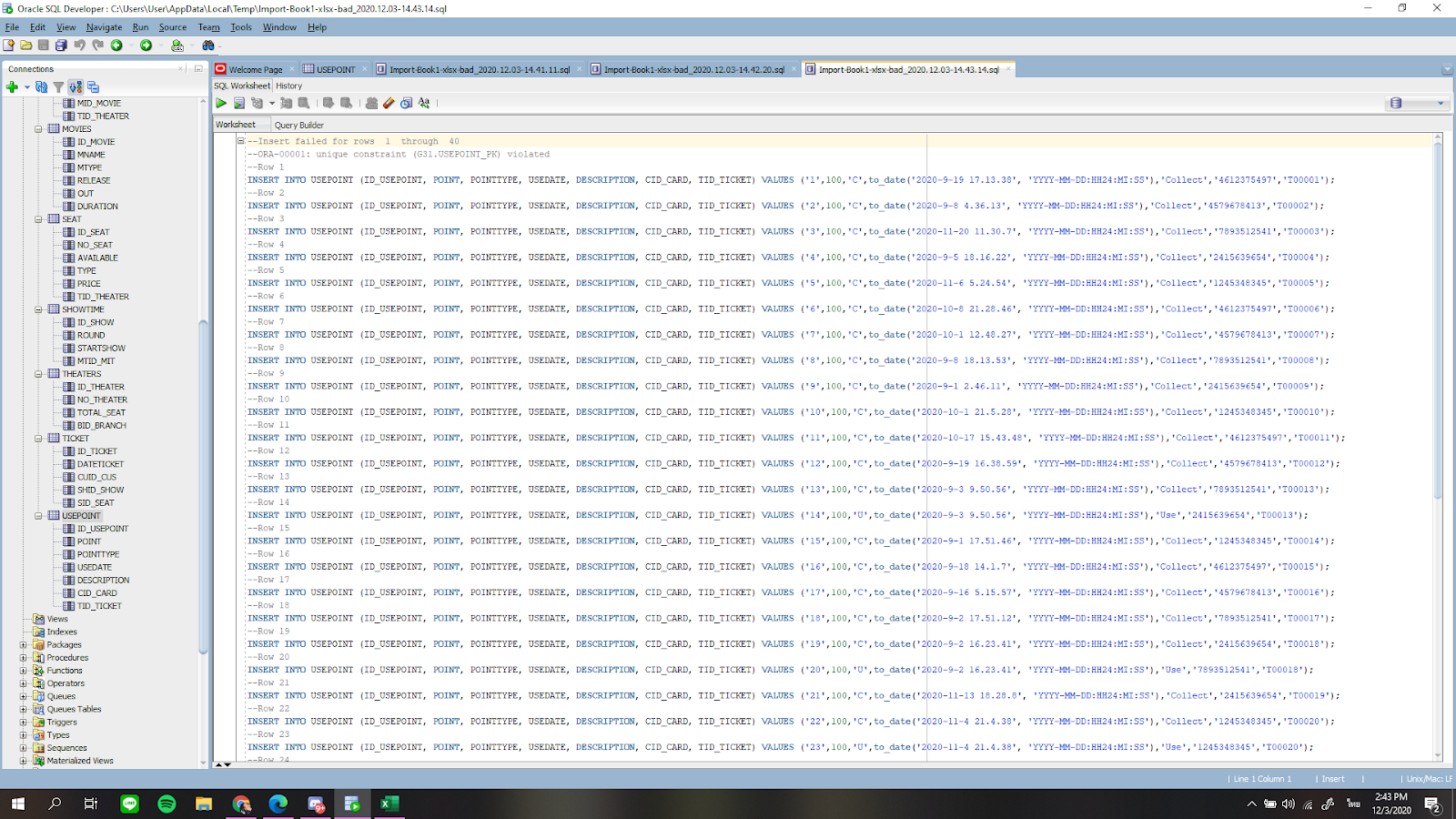


1. Table: card\_type



1. Table: customers
2. Table: employees



1. Table: movie\_in\_theater
2. Table: movies
3. Table: seat
4. Table: showtime
5. Table: theaters
6. Table: ticket
7. Table: usepoint