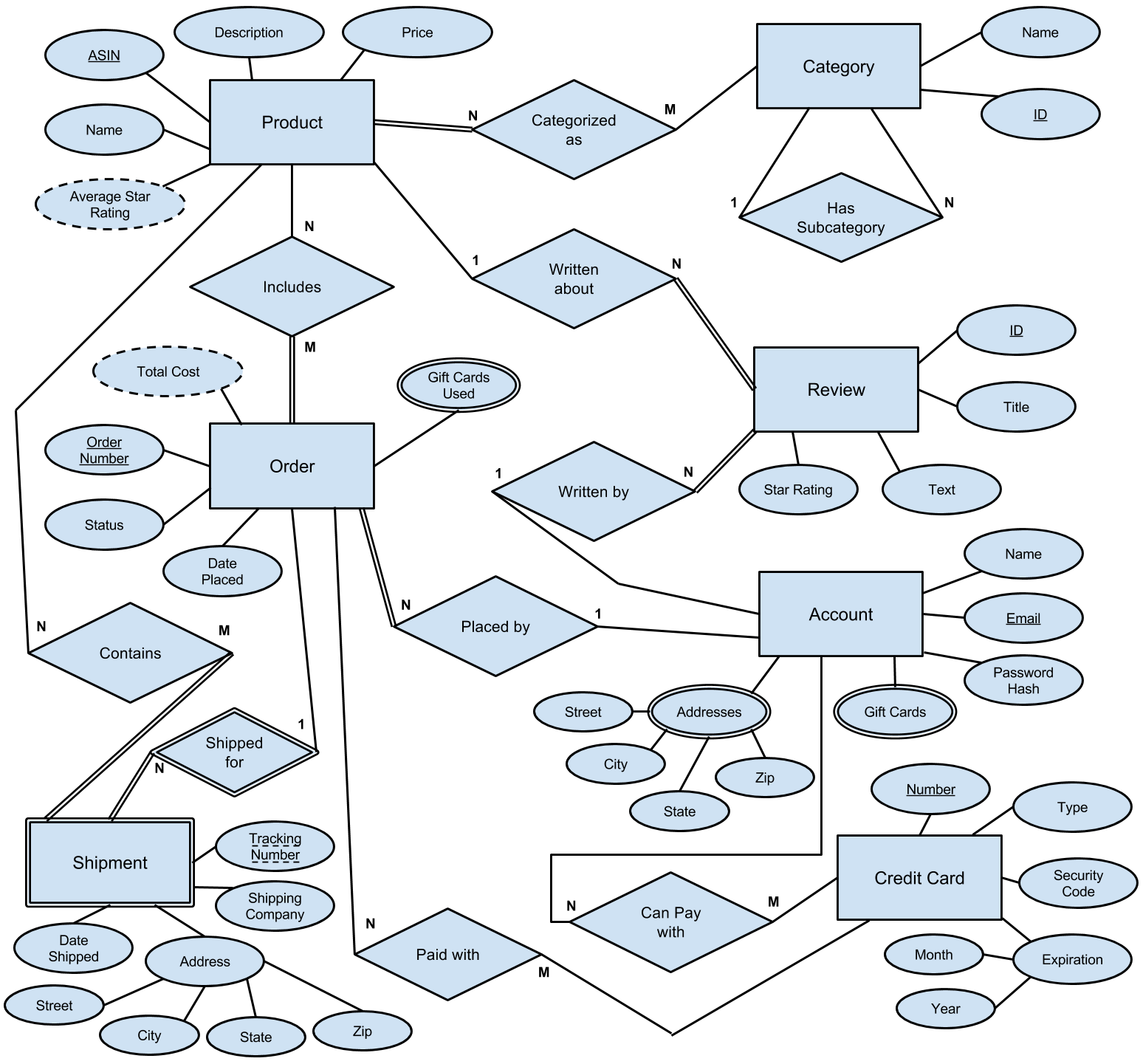
**1. ER Diagram**



**2. Oracle Code for Creating the Database**

/\* Drop the tables if they already exist. \*/

DROP TABLE product CASCADE CONSTRAINTS;

DROP TABLE category CASCADE CONSTRAINTS;

DROP TABLE review CASCADE CONSTRAINTS;

DROP TABLE account CASCADE CONSTRAINTS;

DROP TABLE orders CASCADE CONSTRAINTS;

DROP TABLE shipment CASCADE CONSTRAINTS;

DROP TABLE credit\_card CASCADE CONSTRAINTS;

DROP TABLE product\_categorized\_as CASCADE CONSTRAINTS;

DROP TABLE order\_includes CASCADE CONSTRAINTS;

DROP TABLE shipment\_contains CASCADE CONSTRAINTS;

DROP TABLE account\_credit\_cards CASCADE CONSTRAINTS;

DROP TABLE order\_paid\_with CASCADE CONSTRAINTS;

DROP TABLE order\_gift\_cards CASCADE CONSTRAINTS;

DROP TABLE account\_gift\_cards CASCADE CONSTRAINTS;

DROP TABLE account\_addresses CASCADE CONSTRAINTS;

/\* Create the tables. \*/

CREATE TABLE product

(

ASIN CHAR (10) CONSTRAINT product\_asin\_pk PRIMARY KEY,

Name VARCHAR2 (128) CONSTRAINT product\_name\_nn NOT NULL,

Description VARCHAR2 (2048),

Price NUMBER (9, 2) CONSTRAINT product\_price\_nn NOT NULL

);

CREATE TABLE category

(

Category\_ID NUMBER (10) CONSTRAINT category\_category\_id\_pk PRIMARY KEY,

Name VARCHAR2 (128) CONSTRAINT category\_name\_nn NOT NULL,

Subcategory\_Of NUMBER (10)

);

CREATE TABLE review

(

Review\_ID CHAR (14) CONSTRAINT review\_review\_id\_pk PRIMARY KEY,

Title VARCHAR2 (128),

Text VARCHAR2 (2048),

Star\_Rating NUMBER (1) CONSTRAINT review\_star\_rating\_nn NOT NULL,

Written\_By VARCHAR2 (64) CONSTRAINT review\_written\_by\_nn NOT NULL,

Written\_About CHAR (10) CONSTRAINT review\_written\_about\_nn NOT NULL

);

CREATE TABLE account

(

Email VARCHAR2 (64) CONSTRAINT account\_email\_pk PRIMARY KEY,

Name VARCHAR2 (128) CONSTRAINT account\_name\_nn NOT NULL,

Password\_Hash CHAR (40) CONSTRAINT account\_password\_hash\_nn NOT NULL

);

CREATE TABLE orders

(

Order\_Number NUMBER (17) CONSTRAINT orders\_order\_number\_pk PRIMARY KEY,

Date\_Placed DATE CONSTRAINT orders\_date\_placed\_nn NOT NULL,

Status VARCHAR2 (64) CONSTRAINT orders\_status\_nn NOT NULL,

Placed\_By VARCHAR2 (64) CONSTRAINT orders\_placed\_by\_nn NOT NULL

);

CREATE TABLE shipment

(

Order\_Number NUMBER (17),

Tracking\_Number VARCHAR2 (32),

Company VARCHAR2 (32) CONSTRAINT shipment\_company\_nn NOT NULL,

Date\_Shipped DATE,

Street VARCHAR2 (32) CONSTRAINT shipment\_street\_nn NOT NULL,

City VARCHAR2 (32) CONSTRAINT shipment\_city\_nn NOT NULL,

State CHAR (2) CONSTRAINT shipment\_state\_nn NOT NULL,

Zip NUMBER (5) CONSTRAINT shipment\_zip\_nn NOT NULL,

CONSTRAINT shipment\_ordno\_trackno\_pk PRIMARY KEY (Order\_Number, Tracking\_Number)

);

CREATE TABLE credit\_card

(

Card\_Number NUMBER (16) CONSTRAINT credit\_card\_card\_number\_pk PRIMARY KEY,

Type VARCHAR2 (16) CONSTRAINT credit\_card\_type\_nn NOT NULL,

Security\_Code NUMBER (4) CONSTRAINT credit\_card\_security\_code\_nn NOT NULL,

Exp\_Month NUMBER (2) CONSTRAINT credit\_card\_exp\_month\_nn NOT NULL,

Exp\_Year NUMBER (2) CONSTRAINT credit\_card\_exp\_year\_nn NOT NULL

);

CREATE TABLE product\_categorized\_as

(

Product\_ASIN CHAR (10),

Category\_ID NUMBER (10),

CONSTRAINT product\_cat\_as\_asin\_catid\_pk PRIMARY KEY (Product\_ASIN, Category\_ID)

);

CREATE TABLE order\_includes

(

Order\_Number NUMBER (17),

Product\_ASIN CHAR (10),

CONSTRAINT order\_includes\_ordno\_asin\_pk PRIMARY KEY (Order\_Number, Product\_ASIN)

);

CREATE TABLE shipment\_contains

(

Order\_Number NUMBER (17),

Tracking\_Number VARCHAR2 (32),

Product\_ASIN CHAR (10),

CONSTRAINT shipment\_cont\_ono\_tno\_asin\_pk PRIMARY KEY (Order\_Number, Tracking\_Number, Product\_ASIN)

);

CREATE TABLE account\_credit\_cards

(

Account\_Email VARCHAR2 (64),

Credit\_Card\_Number NUMBER (16),

CONSTRAINT account\_ccs\_email\_cardno\_pk PRIMARY KEY (Account\_Email, Credit\_Card\_Number)

);

CREATE TABLE order\_paid\_with

(

Order\_Number NUMBER (17),

Credit\_Card\_Number NUMBER (16),

CONSTRAINT order\_paid\_w\_ordno\_cardno\_pk PRIMARY KEY (Order\_Number, Credit\_Card\_Number)

);

CREATE TABLE order\_gift\_cards

(

Order\_Number NUMBER (17),

Gift\_Card\_Number CHAR (15),

CONSTRAINT order\_gcs\_ordno\_gcno\_pk PRIMARY KEY (Order\_Number, Gift\_Card\_Number)

);

CREATE TABLE account\_gift\_cards

(

Account\_Email VARCHAR2 (64),

Gift\_Card\_Number CHAR (15),

CONSTRAINT account\_gcs\_email\_gcno\_pk PRIMARY KEY (Account\_Email, Gift\_Card\_Number)

);

CREATE TABLE account\_addresses

(

Account\_Email VARCHAR2 (64),

Street VARCHAR2 (32),

City VARCHAR2 (32),

State CHAR (2),

Zip NUMBER (5),

CONSTRAINT account\_addrs\_em\_s\_c\_s\_z\_pk PRIMARY KEY (Account\_Email, Street, City, State, Zip)

);

/\* Add the foreign keys. \*/

ALTER TABLE category

ADD CONSTRAINT category\_subcategory\_of\_fk

FOREIGN KEY (Subcategory\_Of) REFERENCES category (Category\_ID);

ALTER TABLE review

ADD CONSTRAINT review\_written\_by\_fk

FOREIGN KEY (Written\_By) REFERENCES account (Email);

ALTER TABLE review

ADD CONSTRAINT review\_written\_about\_fk

FOREIGN KEY (Written\_About) REFERENCES product (ASIN);

ALTER TABLE orders

ADD CONSTRAINT orders\_placed\_by\_fk

FOREIGN KEY (Placed\_By) REFERENCES account (Email);

ALTER TABLE shipment

ADD CONSTRAINT shipment\_order\_number\_fk

FOREIGN KEY (Order\_Number) REFERENCES orders (Order\_Number);

ALTER TABLE product\_categorized\_as

ADD CONSTRAINT product\_cat\_as\_asin\_fk

FOREIGN KEY (Product\_ASIN) REFERENCES product (ASIN);

ALTER TABLE product\_categorized\_as

ADD CONSTRAINT product\_cat\_as\_catid\_fk

FOREIGN KEY (Category\_ID) REFERENCES category (Category\_ID);

ALTER TABLE order\_includes

ADD CONSTRAINT order\_includes\_ordno\_fk

FOREIGN KEY (Order\_Number) REFERENCES orders (Order\_Number);

ALTER TABLE order\_includes

ADD CONSTRAINT order\_includes\_asin\_fk

FOREIGN KEY (Product\_ASIN) REFERENCES product (ASIN);

ALTER TABLE shipment\_contains

ADD CONSTRAINT shipment\_cont\_ordno\_trkno\_fk

FOREIGN KEY (Order\_Number, Tracking\_Number) REFERENCES shipment (Order\_Number, Tracking\_Number);

ALTER TABLE shipment\_contains

ADD CONSTRAINT shipment\_cont\_asin\_fk

FOREIGN KEY (Product\_ASIN) REFERENCES product (ASIN);

ALTER TABLE account\_credit\_cards

ADD CONSTRAINT account\_ccs\_email\_fk

FOREIGN KEY (Account\_Email) REFERENCES account (Email);

ALTER TABLE account\_credit\_cards

ADD CONSTRAINT account\_ccs\_cardno\_fk

FOREIGN KEY (Credit\_Card\_Number) REFERENCES credit\_card (Card\_Number);

ALTER TABLE order\_paid\_with

ADD CONSTRAINT order\_paid\_with\_ordno\_fk

FOREIGN KEY (Order\_Number) REFERENCES orders (Order\_Number);

ALTER TABLE order\_paid\_with

ADD CONSTRAINT order\_paid\_with\_cardno\_fk

FOREIGN KEY (Credit\_Card\_Number) REFERENCES credit\_card (Card\_Number);

ALTER TABLE order\_gift\_cards

ADD CONSTRAINT order\_gcs\_ordno\_fk

FOREIGN KEY (Order\_Number) REFERENCES orders (Order\_Number);

ALTER TABLE account\_gift\_cards

ADD CONSTRAINT account\_gcs\_email\_fk

FOREIGN KEY (Account\_Email) REFERENCES account (Email);

ALTER TABLE account\_addresses

ADD CONSTRAINT account\_addrs\_email\_fk

FOREIGN KEY (Account\_Email) REFERENCES account (Email);

**3. Oracle Code for Loading the Data**

We emailed this code and its output because it was too long to include in this report. Most of the data was retrieved from the Amazon website using a Python script, but some of it was randomly generated, including credit cards, addresses, gift cards, and passwords.

527 tuples are created in the database:

* 22 tuples in the product table
* 57 tuples in the category table
* 102 tuples in the review table
* 105 tuples in the account table
* 13 tuples in the orders table
* 12 tuples in the shipment table
* 18 tuples in the credit\_card table
* 84 tuples in the product\_categorized\_as table
* 22 tuples in the order\_includes table
* 21 tuples in the shipment\_contains table
* 18 tuples in the account\_credit\_cards table
* 11 tuples in the order\_paid\_with table
* 12 tuples in the order\_gift\_cards table
* 13 tuples in the account\_gift\_cards table
* 17 tuples in the account\_addresses table

**4. Relational Algebra Queries**

**1. Determine the average star rating of Fundamentals of Database Systems (7th Edition) based on its reviews. (2 tables)**

Matching\_Product ← πASIN(σName='Fundamentals of Database Systems (7th Edition)'(Product))

Product\_Reviews ← πStar\_Rating(Matching\_Product ⋈ASIN=Written\_About Review)

Result ← ƷAVERAGE Star\_Rating(Product\_Reviews)

**2. Determine the total cost of order 10582090927795818. (2 tables)**

Products\_In\_Order ← πProduct\_ASIN(σOrder\_Number=10582090927795818(Order\_Includes))

Result ← ƷSUM Price(Products\_In\_Order ⋈Product\_ASIN=ASIN Product)

**3. List all of Nicholas Pruss's addresses that are in the city of Conway. (2 tables)**

Nicholas\_Email ← πEmail(σName='Nicholas Pruss'(Account))

Result ← πStreet, City, State, Zip(σCity='Conway'(Nicholas\_Email ⋈Email=Account\_Email Account\_Addresses))

**4. List all of the gift cards redeemed by Shane Johnson. (2 tables)**

Shane\_Email ← πEmail(σName='Shane Johnson'(Account))

Result ← πGift\_Card\_Number(Shane\_Email ⋈Email=Account\_Email Account\_Gift\_Cards)

**5. Find the names and email addresses of everyone who has an American Express credit card saved to their account. (3 tables)**

AmEx\_Cards ← πCard\_Number(σType='American Express'(Credit\_Card))

AmEx\_Emails ← πAccount\_Email(AmEx\_Cards ⋈Card\_Number=Credit\_Card\_Number Account\_Credit\_Cards)

Result ← πName, Email(AmEx\_Emails ⋈Account\_Email=Email Account)

**6. Find the tracking number, shipping company, and date shipped for all of Edwin Saez’s shipments. (3 tables)**

Edwin\_Email ← πEmail(σName='Edwin Saez'(Account))

Edwin\_Orders ← πOrder\_Number(Edwin\_Email ⋈Email=Placed\_By Orders)

Result ← πTracking\_Number, Company, Date\_Shipped(Edwin\_Orders ⋈Order\_Number=Order\_Number Shipment)

**7. Find the price of the most expensive product in the Books category. (3 tables)**

Books\_Category ← πCategory\_ID(σName='Books'(Category))

Books\_Products ← πProduct\_ASIN(Books\_Category ⋈Category\_ID=Category\_ID Product\_Categorized\_As)

Result ← ƷMAX Price(πPrice(Books\_Products ⋈Product\_ASIN=ASIN Product))

**8. Determine the total amount of money that Shane Johnson has spent at Amazon. (4 tables)**

User\_Email ← πEmail(σName='Shane Johnson'(Account))

User\_Orders ← πOrder\_Number(User\_Email ⋈Email=Placed\_By Orders)

User\_Products ← πProduct\_ASIN(User\_Orders ⋈Order\_Number=Order\_Number Order\_Includes)

Result ← ƷSUM Price(User\_Products ⋈Product\_ASIN=ASIN Product)

**9. Find the price of the most expensive product that each person has purchased using a gift card. (4 tables)**

Gift\_Card\_Orders ← πOrder\_Number, Placed\_By(Order\_Gift\_Cards ⋈Order\_Number=Order\_Number Orders)

Gift\_Card\_Products ← πProduct\_ASIN, Placed\_By(Gift\_Card\_Orders ⋈Order\_Number=Order\_Number Order\_Includes)

Result ← Placed\_ByƷMAX Price(πPrice, Placed\_By(Gift\_Card\_Products ⋈Product\_ASIN=ASIN Product))

**10. Find the names and email addresses of everyone who has purchased a product in the Video Games category. (5 tables)**

Games\_Category ← πCategory\_ID(σName='Video Games'(Category))

Games\_Products ← πProduct\_ASIN(Games\_Category ⋈Category\_ID=Category\_ID Product\_Categorized\_As)

Games\_Orders ← πOrder\_Number(Games\_Products ⋈Product\_ASIN=Product\_ASIN Order\_Includes)

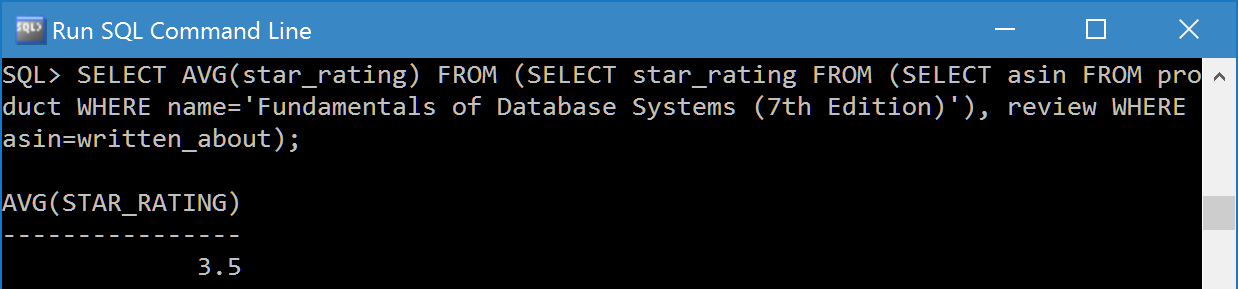
Games\_Accounts ← πPlaced\_By(Games\_Orders ⋈Order\_Number=Order\_Number Orders)

Result ← πName, Email(Games\_Accounts ⋈Placed\_By=Email Account)

**5. Oracle Queries**

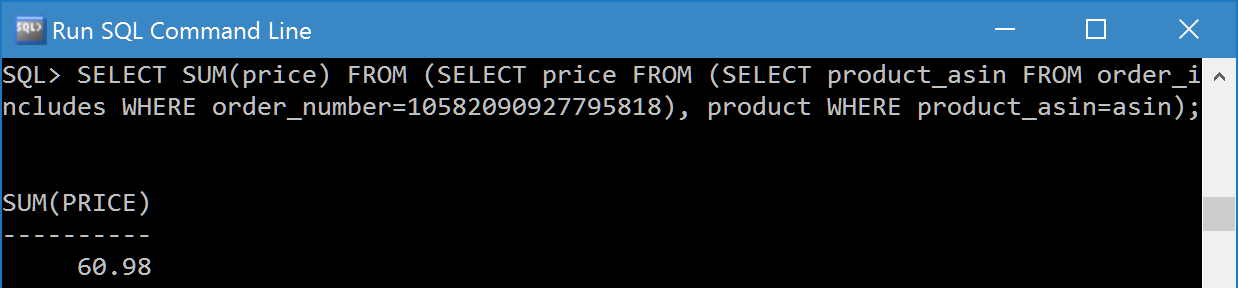
**1. Determine the average star rating of Fundamentals of Database Systems (7th Edition) based on its reviews. (2 tables)**

SELECT AVG(star\_rating) FROM (SELECT star\_rating FROM (SELECT asin FROM product WHERE name='Fundamentals of Database Systems (7th Edition)'), review WHERE asin=written\_about);



**2. Determine the total cost of order 10582090927795818. (2 tables)**

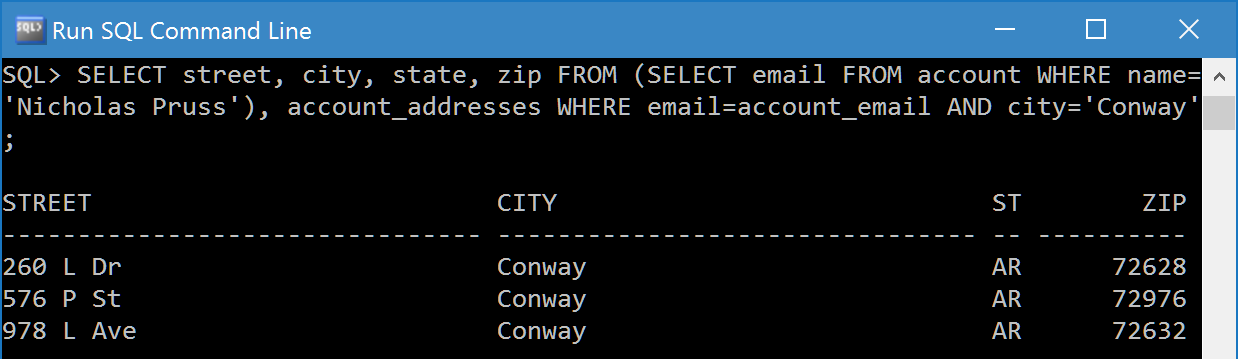
SELECT SUM(price) FROM (SELECT price FROM (SELECT product\_asin FROM order\_includes WHERE order\_number=10582090927795818), product WHERE product\_asin=asin);



**3. List all of Nicholas Pruss's addresses that are in the city of Conway. (2 tables)**

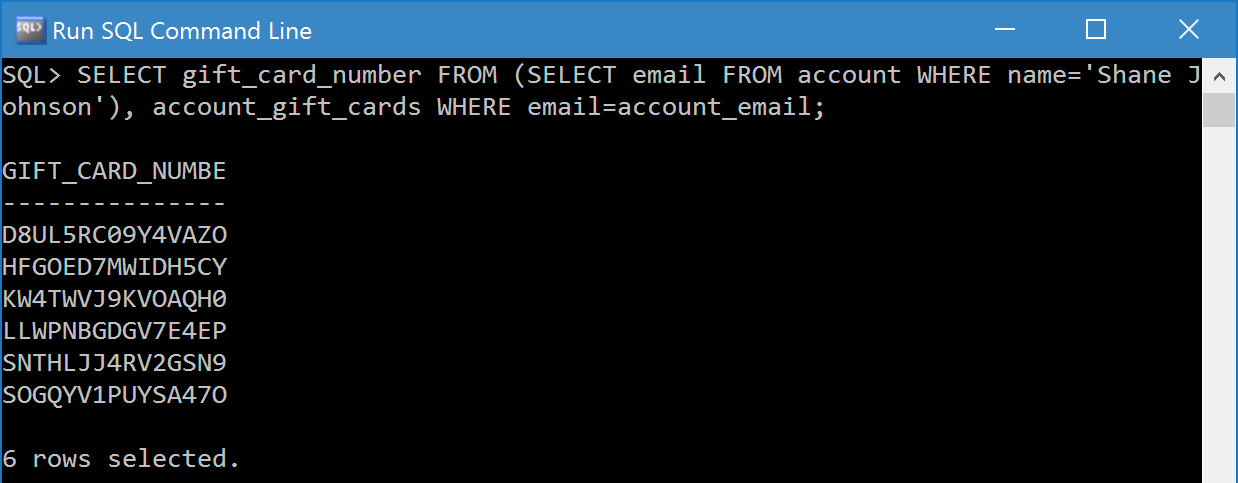
SELECT street, city, state, zip FROM (SELECT email FROM account WHERE name='Nicholas Pruss'), account\_addresses WHERE email=account\_email AND city='Conway';

(Results screenshot is on the next page.)



**4. List all of the gift cards redeemed by Shane Johnson. (2 tables)**

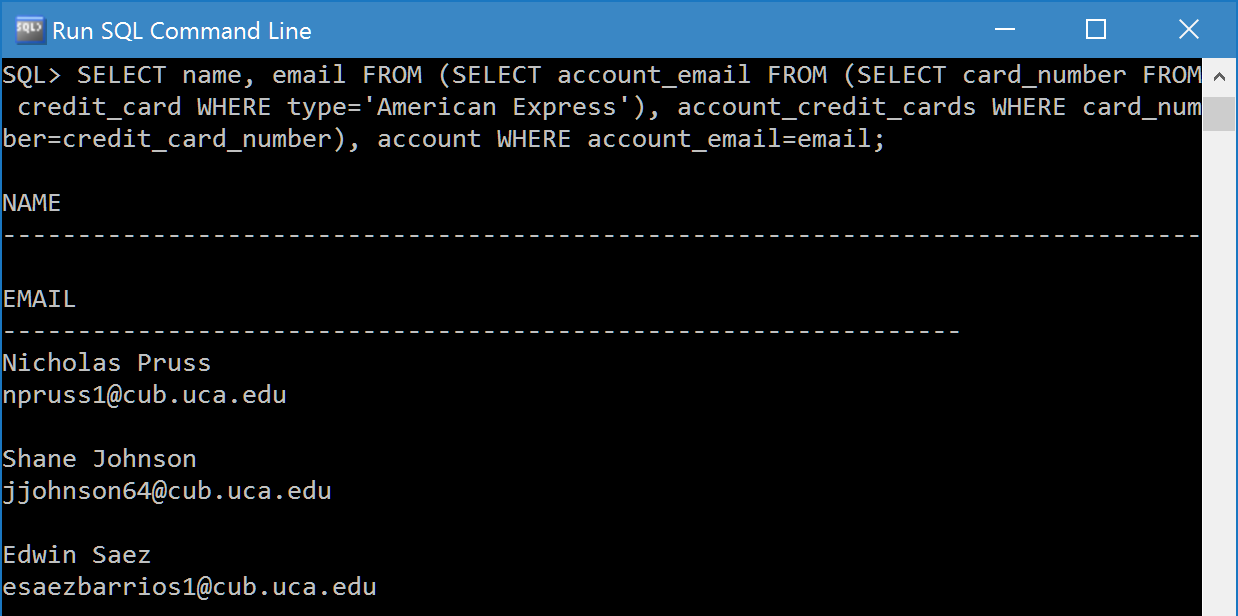
SELECT gift\_card\_number FROM (SELECT email FROM account WHERE name='Shane Johnson'), account\_gift\_cards WHERE email=account\_email;



**5. Find the names and email addresses of everyone who has an American Express credit card saved to their account. (3 tables)**

SELECT name, email FROM (SELECT account\_email FROM (SELECT card\_number FROM credit\_card WHERE type='American Express'), account\_credit\_cards WHERE card\_number=credit\_card\_number), account WHERE account\_email=email;

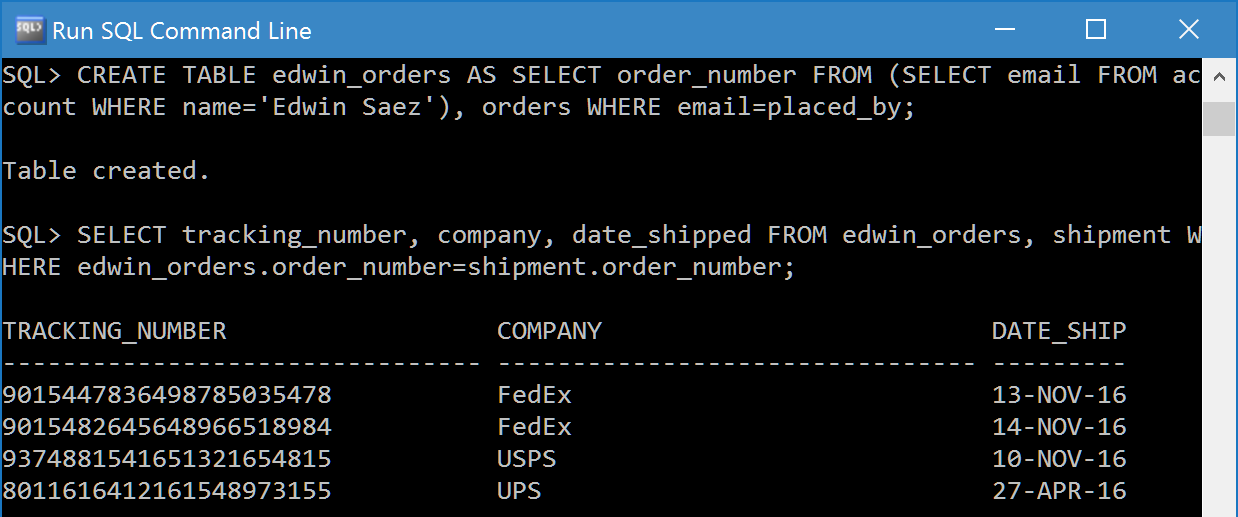
(Results screenshot is on the next page.)



**6. Find the tracking number, shipping company, and date shipped for all of Edwin Saez’s shipments. (3 tables)**

CREATE TABLE edwin\_orders AS SELECT order\_number FROM (SELECT email FROM account WHERE name='Edwin Saez'), orders WHERE email=placed\_by;

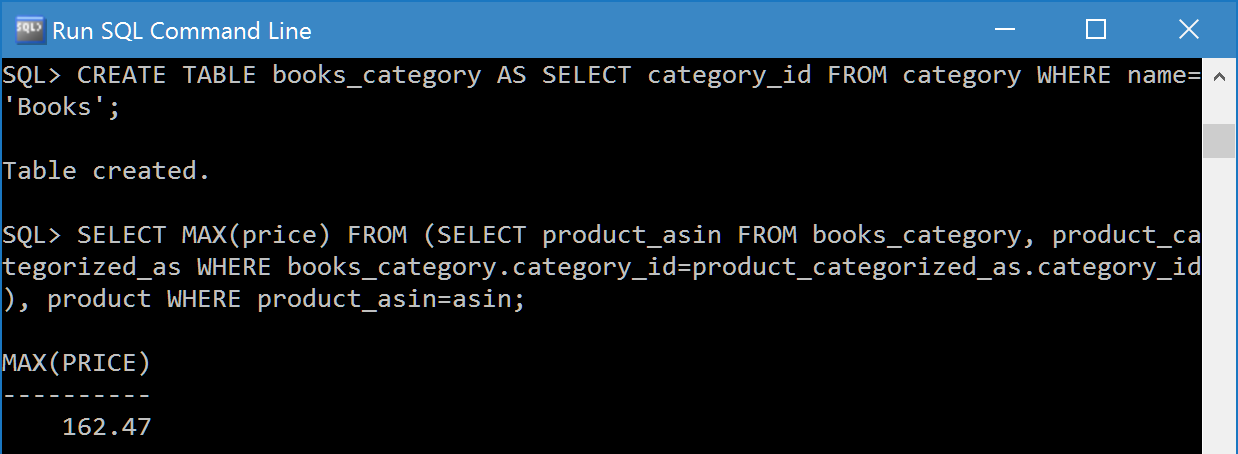
SELECT tracking\_number, company, date\_shipped FROM edwin\_orders, shipment WHERE edwin\_orders.order\_number=shipment.order\_number;



**7. Find the price of the most expensive product in the Books category. (3 tables)**

CREATE TABLE books\_category AS SELECT category\_id FROM category WHERE name='Books';

SELECT MAX(price) FROM (SELECT product\_asin FROM books\_category, product\_categorized\_as WHERE books\_category.category\_id=product\_categorized\_as.category\_id), product WHERE product\_asin=asin;

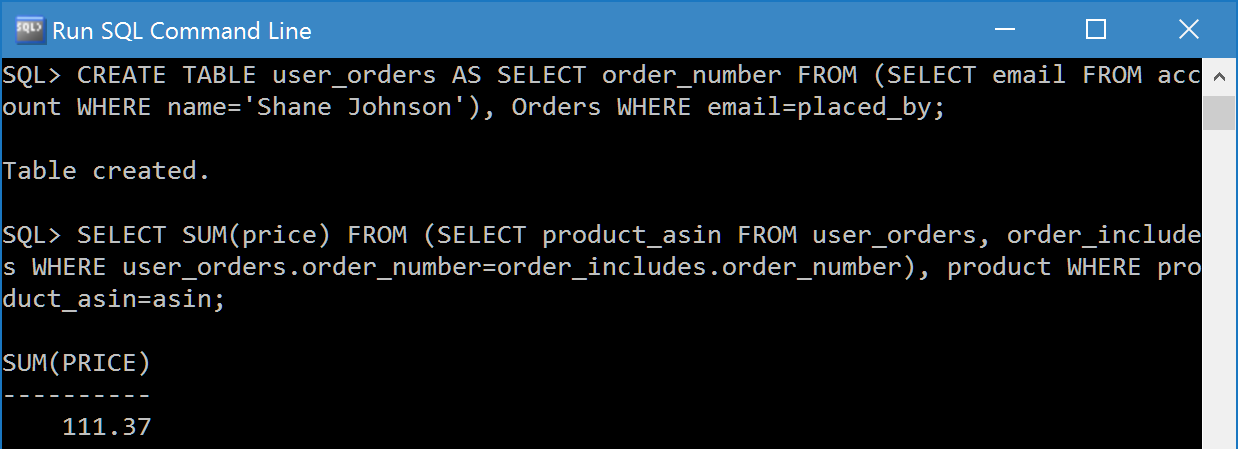


**8. Determine the total amount of money that Shane Johnson has spent at Amazon. (4 tables)**

CREATE TABLE user\_orders AS SELECT order\_number FROM (SELECT email FROM account WHERE name='Shane Johnson'), Orders WHERE email=placed\_by;

SELECT SUM(price) FROM (SELECT product\_asin FROM user\_orders, order\_includes WHERE user\_orders.order\_number=order\_includes.order\_number), product WHERE product\_asin=asin;

(Results screenshot is on the next page.)



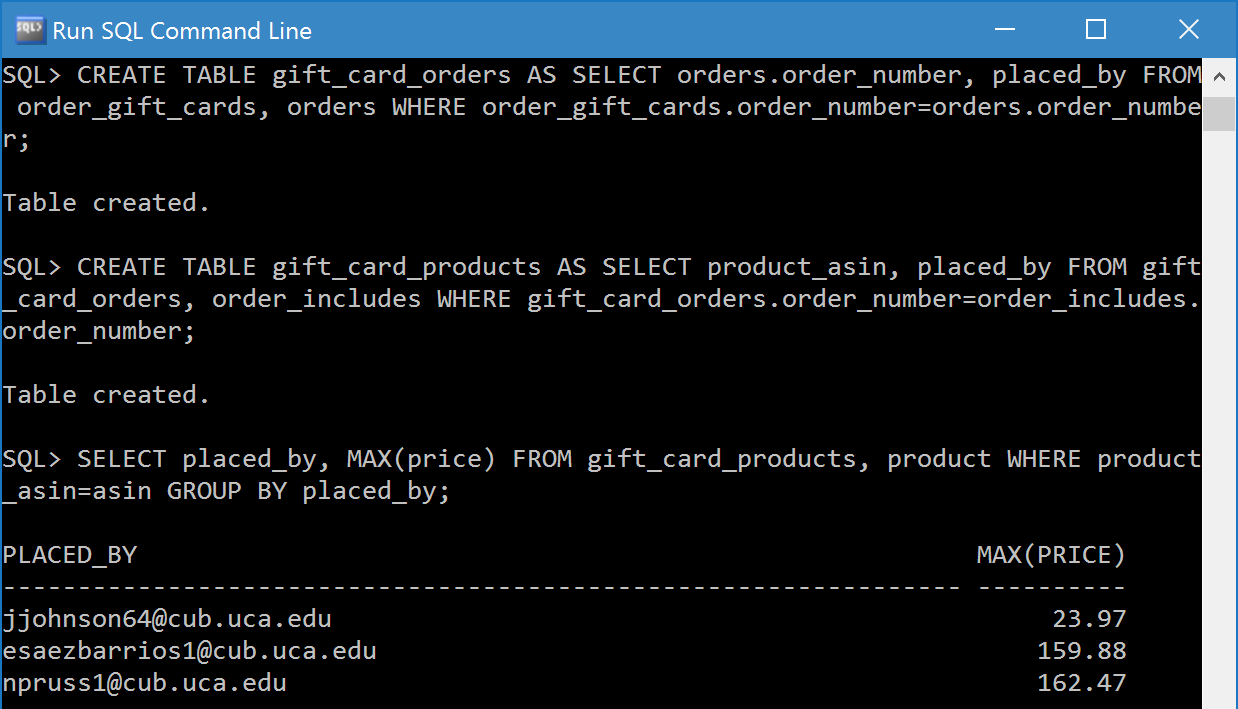
**9. Find the price of the most expensive product that each person has purchased using a gift card. (4 tables)**

CREATE TABLE gift\_card\_orders AS SELECT orders.order\_number, placed\_by FROM order\_gift\_cards, orders WHERE order\_gift\_cards.order\_number=orders.order\_number;

CREATE TABLE gift\_card\_products AS SELECT product\_asin, placed\_by FROM gift\_card\_orders, order\_includes WHERE gift\_card\_orders.order\_number=order\_includes.order\_number;

SELECT placed\_by, MAX(price) FROM gift\_card\_products, product WHERE product\_asin=asin GROUP BY placed\_by;

(Results screenshot is on the next page.)



**10. Find the names and email addresses of everyone who has purchased a product in the Video Games category. (5 tables)**

CREATE TABLE games\_category AS SELECT category\_id FROM category WHERE name='Video Games';

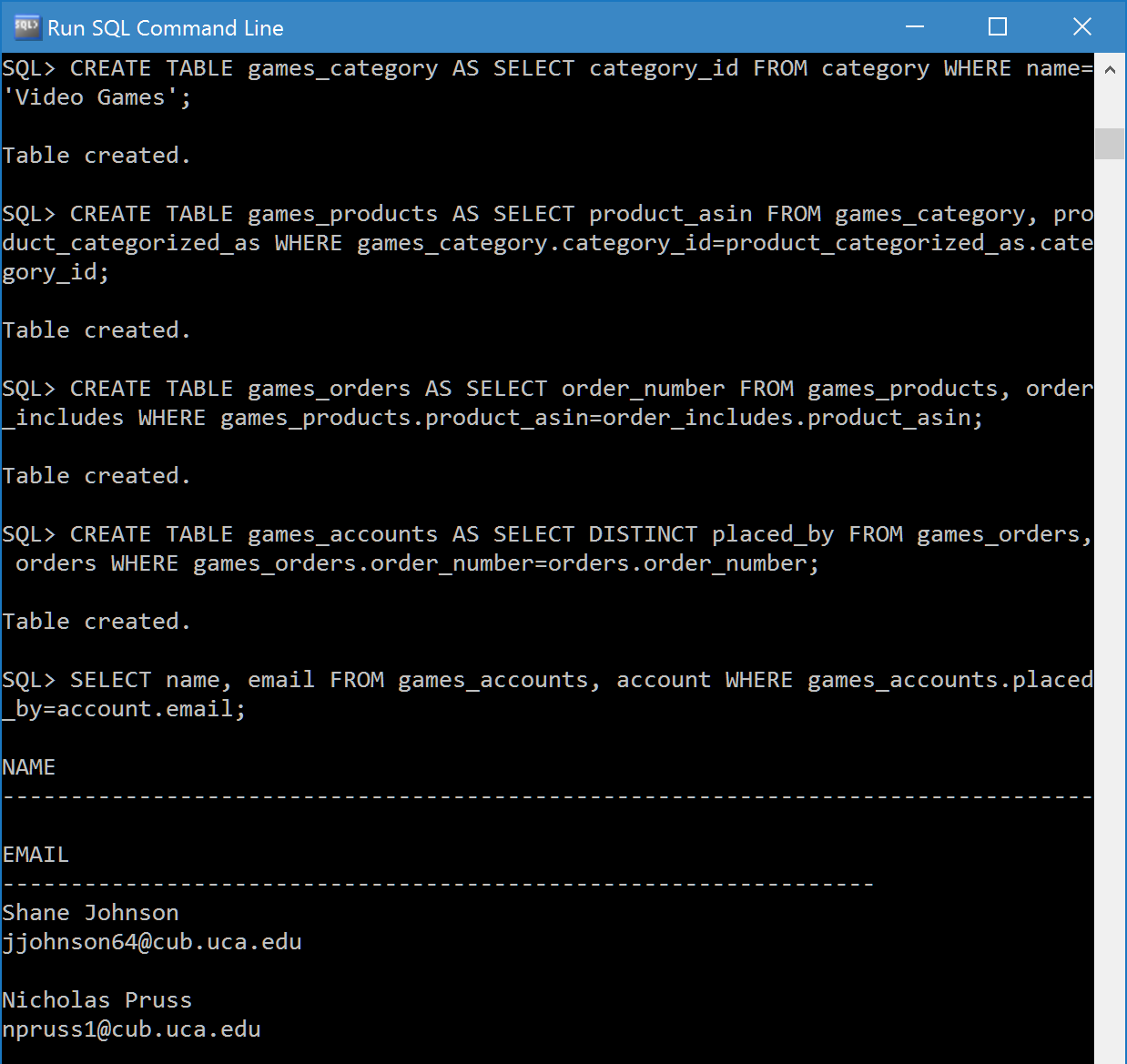
CREATE TABLE games\_products AS SELECT product\_asin FROM games\_category, product\_categorized\_as WHERE games\_category.category\_id=product\_categorized\_as.category\_id;

CREATE TABLE games\_orders AS SELECT order\_number FROM games\_products, order\_includes WHERE games\_products.product\_asin=order\_includes.product\_asin;

CREATE TABLE games\_accounts AS SELECT DISTINCT placed\_by FROM games\_orders, orders WHERE games\_orders.order\_number=orders.order\_number;

SELECT name, email FROM games\_accounts, account WHERE games\_accounts.placed\_by=account.email;

(Results screenshot is on the next page.)



**6. Presentation Slides**

The presentation slides begin on the next page.