Chris Beall

Keenan Benton

Carson Loudner

**CIS355 VTC PT 3**

MySQL

Database Name: carsonKeenanChrisPracticeDB

SQL commands:

CREATE TABLE Vendor (

VendorID VARCHAR(11) NOT NULL,

VendorName VARCHAR(20) NOT NULL,

VendorAddress VARCHAR(25),

PhoneNum VARCHAR(14),

RepresentativeName VARCHAR(20),

CONSTRAINT VendorPK PRIMARY KEY (VendorID)

);

/\*Employee( EmployeeID, EmployeeName, Type, SSN, StartDate, JobTitle, InitialPayRate, CurrentPayRate, EmployeeAddress, PhoneNumber, EmailAddress, EmergencyContactPhone, EmergencyContactName, Hours, SuperVisorID)\*/

CREATE TABLE Employee (

EmployeeID VARCHAR(11) NOT NULL,

EmployeeName VARCHAR (20),

EmployeeType VARCHAR (20),

SSN VARCHAR (11),

StartDate DATE DEFAULT CURRENT\_TIMESTAMP,

JobTitle VARCHAR (20),

InitialPayRate DECIMAL (9,2),

CurrentPayRate DECIMAL (9,2),

EmployeeAddress VARCHAR (40),

PhoneNumber VARCHAR(14),

EmailAddress VARCHAR (40),

EmergencyContactPhone VARCHAR (14),

EmergencyContactName VARCHAR (20),

Hours VARCHAR (20)

CHECK(Hours IN ('Part-Time', 'Full-Time')),

SupervisorID VARCHAR (11),

CONSTRAINT EmployeePK PRIMARY KEY (EmployeeID),

/\*CONSTRAINT EmployeeFK FOREIGN KEY (SupervisorID) REFERENCES Employee(SupervisorID)\*/

);

/\*PurchaseOrders(PurchaseOrderID, VendorID, OrderQuantity, QuantityReceived, DateOrdered, DateReceived, TotalAmount)\*/

CREATE TABLE PurchaseOrder (

PurchaseOrderID VARCHAR(11) NOT NULL,

VendorID VARCHAR(11) NOT NULL,

OrderQuantity VARCHAR (10),

QuantityReceived VARCHAR (10),

DateOrdered DATE DEFAULT CURRENT\_TIMESTAMP,

DateReceived DATE DEFAULT CURRENT\_TIMESTAMP,

TotalAmount DECIMAL (12),

CONSTRAINT PurchaseOrderPK PRIMARY KEY (PurchaseOrderID),

CONSTRAINT PurchaseOrderFK FOREIGN KEY (VendorID)

REFERENCES Vendor(VendorID));

/\*Packaging(BottleID, BottleCapacity, BottleShape, BottleColor, BottleUnitCost)\*/

CREATE TABLE Packaging (

BottleID VARCHAR(11) NOT NULL,

BottleCapacity VARCHAR (20),

BottleShape VARCHAR (20),

BottleColor VARCHAR (20),

BottleUnitCost DECIMAL (5,2),

CONSTRAINT PackagingPK PRIMARY KEY (BottleID));

/\*Wine (WineID, Category, Alcohol%, GrapeProportion, EmployeeID, BottleID)\*/

CREATE TABLE Wine (

WineID VARCHAR(11) NOT NULL,

Category VARCHAR (20),

AlcoholPercent DECIMAL (3,2),

GrapeProportion VARCHAR (20),

EmployeeID VARCHAR(11),

BottleID VARCHAR(11),

CONSTRAINT WinePK PRIMARY KEY (WineID),

CONSTRAINT WineFK1 FOREIGN KEY (EmployeeID) REFERENCES Employee (EmployeeID),

CONSTRAINT WineFK2 FOREIGN KEY (BottleID) REFERENCES Packaging (BottleID));

/\*Vineyard(VineyardID, Size, Location, EmployeeID, GrapeVariety, YineyardName, VineyardAddress )\*/

CREATE TABLE Vineyard (

VineyardID VARCHAR(11) NOT NULL,

Size VARCHAR(20),

Location VARCHAR (50),

EmployeeID VARCHAR(11),

GrapeVariety VARCHAR (20),

VineyardName VARCHAR (20),

VineyardAddress VARCHAR (30),

CONSTRAINT VineyardPK PRIMARY KEY (VineyardID),

CONSTRAINT VineyardFK FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID));

/\*Grape(GrapeID, ConversionRatio, StorageType, AgingRequirement, GrapeType)\*/

CREATE TABLE Grape (

GrapeID VARCHAR(11) NOT NULL,

ConversionRatio VARCHAR(10),

StorageType VARCHAR (20),

AgingRequirement VARCHAR (15),

GrapeType VARCHAR (20),

CONSTRAINT GrapePK PRIMARY KEY (GrapeID));

/\*BatchProductionHarvest(BatchID, GrapeID, Weight, Ripeness%, VineyardID)\*/

CREATE TABLE BatchProductionHarvest (

BatchID VARCHAR(11) NOT NULL,

GrapeID VARCHAR(11),

BatchWeight DECIMAL (10,2),

RipenessPercent DECIMAL (3, 2),

VineyardID VARCHAR(11),

CONSTRAINT BatchProductionHarvestPK PRIMARY KEY (BatchID),

CONSTRAINT BatchProductionHarvestFK1 FOREIGN KEY (GrapeID) REFERENCES Grape(GrapeID),

CONSTRAINT BatchProductionHarvestFK2 FOREIGN KEY (VineyardID) REFERENCES Vineyard(VineyardID));

/\*Product(SKU, QuantityOnHand, ProductType, Price, ProductName, Subcategory, VintageYear, WineID)\*/

CREATE TABLE Product (

SKU VARCHAR(11) NOT NULL,

QuantityOnHand VARCHAR(10),

ProductType VARCHAR (20),

Price DECIMAL (5,2),

ProductName VARCHAR (20),

Subcategory VARCHAR (20),

VintageYear VARCHAR(4),

WineID VARCHAR(11),

CONSTRAINT ProductPK PRIMARY KEY (SKU),

CONSTRAINT ProductFK FOREIGN KEY (WineID) REFERENCES Wine (WineID));

/\*Supplier(SupplierID, Name, PhoneNumber)\*/

CREATE TABLE Supplier (

SupplierID VARCHAR(11),

SupplierName VARCHAR(20),

PhoneNumber VARCHAR(14),

CONSTRAINT SupplierPK PRIMARY KEY (SupplierID)

);

/\*Food(SKU, ExpirationDate, FoodType, SupplierID)\*/

CREATE TABLE Food (

SKU VARCHAR (11) NOT NULL,

ExpirationDate DATE DEFAULT CURRENT\_TIMESTAMP,

FoodType VARCHAR (20),

SupplierID VARCHAR (11),

CONSTRAINT FoodPK PRIMARY KEY (SKU),

CONSTRAINT FoodFK FOREIGN KEY (SupplierID) REFERENCES Supplier(SupplierID));

/\*(WineProduct(SKU, Vintage, GrapeType, Size, Quantity)\*/

CREATE TABLE WineProduct (

SKU VARCHAR (11) NOT NULL,

Vintage INTEGER,

GrapeType VARCHAR (20),

Size VARCHAR (15),

Quantity VARCHAR (8),

CONSTRAINT WineProductPK PRIMARY KEY (SKU));

/\*WineMaker(EmployeeID, CertificationLevel, CertificationDate)\*/

CREATE TABLE Winemaker (

EmployeeID VARCHAR (11) NOT NULL,

CertificationLevel VARCHAR (20),

CertificationDate DATE DEFAULT CURRENT\_TIMESTAMP,

CONSTRAINT WinemakerPK PRIMARY KEY (EmployeeID));

/\*EmployeeRecertification(EmployeeID, RecertificationOutcome)\*/

CREATE TABLE EmployeeRecertification (

EmployeeID VARCHAR(11) NOT NULL,

RecertificationOutcome VARCHAR (20)

CHECK(RecertificationOutcome IN ('Recertified', 'Not Recertified')),

CONSTRAINT EmployeeRecertificationPK PRIMARY KEY (EmployeeID, RecertificationOutcome),

CONSTRAINT EmployeeRecertificationFK1 FOREIGN KEY (EmployeeID) REFERENCES Employee (EmployeeID),

);

/\*Intern(EmployeeID, InternshipRequirement, GraduationDate)\*/

CREATE TABLE Intern (

EmployeeID VARCHAR(11) NOT NULL,

InternshipRequirement VARCHAR (40),

GraduationDate DATE DEFAULT CURRENT\_TIMESTAMP,

CONSTRAINT InternPK PRIMARY KEY (EmployeeID));

/\*Driver(EmployeeID, CDLCertified, CDLExpirationDate)\*/

CREATE TABLE Driver (

EmployeeID VARCHAR(11) NOT NULL,

CDLCertified VARCHAR (20)

CHECK (CDLCertified IN ('Certified', 'Not Certified')),

CDLExpirationDate DATE DEFAULT CURRENT\_TIMESTAMP,

CONSTRAINT EmployeeIDPK PRIMARY KEY (EmployeeID));

/\*Room (RoomNum, RoomType, RoomSize)\*/

CREATE TABLE Room (

RoomNum INTEGER NOT NULL,

RoomType VARCHAR (20),

RoomSize VARCHAR (15),

CONSTRAINT RoomPK PRIMARY KEY (RoomNum));

/\*Event (EventID, EventType, RoomNum, Date, Time)\*/

CREATE TABLE VTCEvent (

EventID VARCHAR(11) NOT NULL,

EventType VARCHAR (50),

RoomNum INTEGER,

EventDate DATE DEFAULT CURRENT\_TIMESTAMP,

EventTime VARCHAR (10),

CONSTRAINT VTCEventPK PRIMARY KEY (EventID),

CONSTRAINT VTCEventFK FOREIGN KEY (RoomNum) REFERENCES Room (RoomNum));

/\*EmployeeEvent(EmployeeID, EventID)\*/

CREATE TABLE EmployeeEvent (

EmployeeID VARCHAR(11) NOT NULL,

EventID VARCHAR(11) NOT NULL,

CONSTRAINT EmployeeEventPK PRIMARY KEY (EmployeeID, EventID),

CONSTRAINT EmployeeEventFK1 FOREIGN KEY (EmployeeID) REFERENCES Employee (EmployeeID),

CONSTRAINT EmployeeEventFK2 FOREIGN KEY (EventID) REFERENCES VTCEvent (EventID));

/\*Customer(CustomerID, CustomerAddress, PhoneNumber)\*/

CREATE TABLE Customer (

CustomerID VARCHAR(11),

CustomerAddress VARCHAR(45),

PhoneNumber VARCHAR(12),

CONSTRAINT CustomerPK PRIMARY KEY (CustomerID),

);

/\*BookedEvent (CustomerID, EventID, NumberOfAttendance, DirectToTastingRoom)\*/

CREATE TABLE BookedEvent (

CustomerID VARCHAR(11) NOT NULL,

EventID VARCHAR(11) NOT NULL,

NumberOfAttendance VARCHAR (5),

DirectToTastingRoom VARCHAR (6),

CHECK (DirectToTastingRoom IN ('Yes','No')),

CONSTRAINT BookedEventPK PRIMARY KEY (CustomerID, EventID),

CONSTRAINT BookedEventFK1 FOREIGN KEY (CustomerID) REFERENCES Customer (CustomerID),

CONSTRAINT BookedEventFK2 FOREIGN KEY (EventID) REFERENCES VTCEvent (EventID));

/\*Order (OrderID, ShippingStatus, DateReceived, CustomerID, TypeOfPayment, DateOfPayment, TotalAmount)\*/

CREATE TABLE CustomerOrder (

OrderID VARCHAR(11),

ShippingStatus VARCHAR(11),

DateReceived DATE DEFAULT CURRENT\_TIMESTAMP,

CustomerID VARCHAR(11),

TypeOfPayment VARCHAR(12),

DateOfPayment DATE DEFAULT CURRENT\_TIMESTAMP,

TotalAmount DECIMAL(7),

CONSTRAINT CustomerOrderPK PRIMARY KEY (OrderID),

CONSTRAINT CustomerOrderFK FOREIGN KEY (CustomerID) REFERENCES Customer (CustomerID)

);

/\*OrderLine (OrderID, SKU, OrderQuantity)\*/

CREATE TABLE OrderLine (

OrderID VARCHAR(11) NOT NULL,

SKU VARCHAR (11) NOT NULL,

OrderQuantity INTEGER,

CONSTRAINT OrderLinePK PRIMARY KEY (OrderID, SKU),

CONSTRAINT OrderLineFK1 FOREIGN KEY (OrderID) REFERENCES CustomerOrder (OrderID),

CONSTRAINT OrderLineFK2 FOREIGN KEY (SKU) REFERENCES Product (SKU));

/\*CustomerStatus(CustomerID, Status)\*/

CREATE TABLE CustomerStatus (

CustomerID VARCHAR(11),

CustomerStatus VARCHAR(12),

CONSTRAINT CustomerStatusPK PRIMARY KEY (CustomerID, CustomerStatus)

);

/\*CustomerBuyingPreferences(CustomerID, Preference)\*/

CREATE TABLE CustomerBuyingPreferences (

CustomerID VARCHAR (11) NOT NULL,

Preference VARCHAR (20),

CONSTRAINT CustomerBuyingPreferencePK PRIMARY KEY (CustomerID, Preference)

);

/\*IndividualCustomer(CustomerID, FirstName, LastName, ShippingAddress, DateOfBirth)\*/

CREATE TABLE IndividualCustomer (

CustomerID VARCHAR (11) NOT NULL,

FirstName VARCHAR (15),

LastName VARCHAR (25),

ShippingAddress VARCHAR (45),

DateOfBirth DATE DEFAULT CURRENT\_TIMESTAMP,

CONSTRAINT IndividualCustomerPK PRIMARY KEY (CustomerID));

/\*ClubMember(CustomerID, DateJoined, DateLeft, WinesOffered)\*/

CREATE TABLE ClubMember (

CustomerID VARCHAR(11) NOT NULL,

DateJoined DATE DEFAULT CURRENT\_TIMESTAMP,

DateLeft DATE DEFAULT CURRENT\_TIMESTAMP,

WinesOffered VARCHAR (20),

CONSTRAINT ClubMemberPK PRIMARY KEY (CustomerID));

/\*Organization(CustomerID, CompanyName, TaxID, RetailLicenseNumber, Discount)\*/

CREATE TABLE Organization (

CustomerID VARCHAR (11) NOT NULL,

CompanyName VARCHAR (30),

TaxID VARCHAR (11),

RetailLicenseNumber INTEGER,

Discount DECIMAL (3,2),

CONSTRAINT OrganizationPK PRIMARY KEY (CustomerID));

/\*Receives (CustomerID, SKU, WineType1, WineType2, Date)\*/

CREATE TABLE Receives (

CustomerID VARCHAR (11) NOT NULL,

SKU VARCHAR (11) NOT NULL,

WineType1 VARCHAR (20),

WineType2 VARCHAR (20),

ReceivesDate DATE DEFAULT CURRENT\_TIMESTAMP,

CONSTRAINT ReceivesPK PRIMARY KEY (CustomerID, SKU),

CONSTRAINT ReceivesFK1 FOREIGN KEY (CustomerID) REFERENCES Customer (CustomerID),

CONSTRAINT ReceivesFK2 FOREIGN KEY (SKU) REFERENCES Product (SKU));

/\*Reservation (ReservationID, EventType, StartDate, EndDate, CustomerID)\*/

CREATE TABLE Reservation (

ReservationID VARCHAR (11) NOT NULL,

EventType VARCHAR (30),

StartDate DATE DEFAULT CURRENT\_TIMESTAMP,

EndDate DATE DEFAULT CURRENT\_TIMESTAMP,

CustomerID VARCHAR (11),

CONSTRAINT ReservationPK PRIMARY KEY (ReservationID),

CONSTRAINT ReservationFK FOREIGN KEY (CustomerID) REFERENCES Customer (CustomerID));

**Database diagram:**

