Person Table:

CREATE TABLE `ecommercedb`.`Person` (

`SSN` INT NOT NULL ,

`FirstName` VARCHAR(45) NOT NULL ,

`LastName` VARCHAR(45) NOT NULL ,

`MiddleName` VARCHAR(45) NULL ,

`Age` TINYINT NOT NULL ,

`Gender` VARCHAR(10) NOT NULL ,

`MobileNumber` INT NOT NULL ,

`ResidentialNumber` INT NULL ,

`EmailID` VARCHAR(45) NOT NULL ,

PRIMARY KEY (`SSN`) );

Employee Table:  
CREATE TABLE `ecommercedb`.`Employee` (

`EmployeeID` VARCHAR(20) NOT NULL ,

`SSN` INT NOT NULL ,

`AccountNumber` VARCHAR(30) NOT NULL ,

PRIMARY KEY (`EmployeeID`) );  
  
Customer Table:  
CREATE TABLE `ecommercedb`.`Customer` (

`CustomerID` VARCHAR(20) NOT NULL ,

`SSN` INT NOT NULL ,

`CustomerTypeID` TINYINT NOT NULL ,

PRIMARY KEY (`CustomerID`) );  
  
CustomerType Table:  
CREATE TABLE `ecommercedb`.`CustomerType` (

`CustomerTypeID` TINYINT NOT NULL ,

`TypeDescription` VARCHAR(45) NOT NULL ,

PRIMARY KEY (`CustomerTypeID`) );  
  
relation between person and employee  
ALTER TABLE `ecommercedb`.`employee`

ADD CONSTRAINT `SSN`

FOREIGN KEY (`SSN` )

REFERENCES `ecommercedb`.`person` (`SSN` )

ON DELETE RESTRICT

ON UPDATE RESTRICT

, ADD INDEX `SSN\_idx` (`SSN` ASC) ;  
  
UserAccount table:

CREATE TABLE `ecommercedb`.`UserAccount` (

`UserAccountNumber` VARCHAR(20) NOT NULL ,

`Username` VARCHAR(45) NOT NULL ,

`Password` VARCHAR(45) NOT NULL ,

`CreatedDate` DATETIME NOT NULL ,

`Visits` INT NULL ,

`SecurityQuestion` VARCHAR(45) NOT NULL ,

`SecurityAnswer` VARCHAR(45) NOT NULL ,

`PersonSSN` INT NOT NULL ,

PRIMARY KEY (`UserAccountNumber`) ,

INDEX `PersonSSN\_idx` (`PersonSSN` ASC) ,

CONSTRAINT `PersonSSN`

FOREIGN KEY (`PersonSSN` )

REFERENCES `ecommercedb`.`person` (`SSN` )

ON DELETE NO ACTION

ON UPDATE NO ACTION)

DEFAULT CHARACTER SET = latin1;  
  
  
UserSession table-

CREATE TABLE `ecommercedb`.`UserSession` (

`UserSessionID` VARCHAR(20) NOT NULL ,

`IPAddress` VARCHAR(45) NOT NULL ,

`Clicks` INT NULL ,

`SessionStartTimestamp` TIMESTAMP NOT NULL ,

`SessionEndTimestamp` TIMESTAMP NOT NULL ,

`UserAccountID` VARCHAR(20) NOT NULL ,

PRIMARY KEY (`UserSessionID`) ,

INDEX `UserAccountID\_idx` (`UserAccountID` ASC) ,

CONSTRAINT `UserAccountID`

FOREIGN KEY (`UserAccountID` )

REFERENCES `ecommercedb`.`useraccount` (`UserAccountNumber` )

ON DELETE NO ACTION

ON UPDATE NO ACTION)

DEFAULT CHARACTER SET = latin1;  
  
address and person:

CREATE TABLE `ecommercedb`.`PersonAddress` (

`PersonAddressSSN` INT NOT NULL ,

`PersonAddressID` INT NOT NULL ,

INDEX `PersonAddressSSN\_idx` (`PersonAddressSSN` ASC) ,

INDEX `PersonAddressID\_idx` (`PersonAddressID` ASC) ,

CONSTRAINT `PersonAddressSSN`

FOREIGN KEY (`PersonAddressSSN` )

REFERENCES `ecommercedb`.`person` (`SSN` )

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `PersonAddressID`

FOREIGN KEY (`PersonAddressID` )

REFERENCES `ecommercedb`.`address` (`AddressID` )

ON DELETE NO ACTION

ON UPDATE NO ACTION)

DEFAULT CHARACTER SET = latin1;  
  
CreditCard:

CREATE TABLE `ecommercedb`.`CreditCard` (

`AccountID` INT NOT NULL ,

`CreditCardNumber` INT NOT NULL ,

`HolderName` VARCHAR(45) NOT NULL ,

`ExpiryDate` DATE NOT NULL ,

`CustomerCardID` VARCHAR(20) NULL ,

PRIMARY KEY (`AccountID`) ,

INDEX `CustomerCardID\_idx` (`CustomerCardID` ASC) ,

CONSTRAINT `CustomerCardID`

FOREIGN KEY (`CustomerCardID` )

REFERENCES `ecommercedb`.`customer` (`CustomerID` )

ON DELETE RESTRICT

ON UPDATE RESTRICT)

DEFAULT CHARACTER SET = latin1;  
  
Payment type:

CREATE TABLE `ecommercedb`.`CreditCard` (

`AccountID` INT NOT NULL ,

`CreditCardNumber` INT NOT NULL ,

`HolderName` VARCHAR(45) NOT NULL ,

`ExpiryDate` DATE NOT NULL ,

`CustomerCardID` VARCHAR(20) NULL ,

PRIMARY KEY (`AccountID`) ,

INDEX `CustomerCardID\_idx` (`CustomerCardID` ASC) ,

CONSTRAINT `CustomerCardID`

FOREIGN KEY (`CustomerCardID` )

REFERENCES `ecommercedb`.`customer` (`CustomerID` )

ON DELETE RESTRICT

ON UPDATE RESTRICT)

DEFAULT CHARACTER SET = latin1;  
  
Payment:  
CREATE TABLE `ecommercedb`.`Payment` (

`PaymentID` INT NOT NULL ,

`PaymentAmount` DECIMAL(15,2) NOT NULL ,

`PaymentState` DECIMAL(15,2) NOT NULL ,

`PaymentTimestamp` TIMESTAMP NOT NULL ,

`PaymentTypeID` INT NOT NULL ,

PRIMARY KEY (`PaymentID`) ,

INDEX `PaymentTypeID\_idx` (`PaymentTypeID` ASC) ,

CONSTRAINT `PaymentTypeID`

FOREIGN KEY (`PaymentTypeID` )

REFERENCES `ecommercedb`.`paymenttype` (`PaymentTypeID` )

ON DELETE NO ACTION

ON UPDATE NO ACTION)

DEFAULT CHARACTER SET = latin1;  
  
Shopping Cart:  
CREATE TABLE `ecommercedb`.`ShoppingCart` (

`ShoppingCartID` INT NOT NULL ,

`Active` VARCHAR(20) NOT NULL ,

`ExpiryDate` DATETIME NOT NULL ,

`CartUserSessionID` VARCHAR(20) NOT NULL ,

PRIMARY KEY (`ShoppingCartID`) ,

INDEX `CartUserSessionID\_idx` (`CartUserSessionID` ASC) ,

CONSTRAINT `CartUserSessionID`

FOREIGN KEY (`CartUserSessionID` )

REFERENCES `ecommercedb`.`usersession` (`UserSessionID` )

ON DELETE RESTRICT

ON UPDATE RESTRICT)

DEFAULT CHARACTER SET = latin1;  
  
InventoryItemCategory:  
CREATE TABLE `ecommercedb`.`InventoryItemCategory` (

`CategoryID` INT NOT NULL ,

`CategoryDescription` VARCHAR(100) NOT NULL ,

PRIMARY KEY (`CategoryID`) )

DEFAULT CHARACTER SET = latin1;  
  
InventoryItem:

CREATE TABLE `ecommercedb`.`InventoryItem` (

`InventoryItemID` INT NOT NULL ,

`Title` VARCHAR(45) NOT NULL ,

`Price` DECIMAL(15,2) NOT NULL ,

`Details` VARCHAR(100) NOT NULL ,

`ManufacturedDate` DATE NOT NULL ,

PRIMARY KEY (`InventoryItemID`) )

DEFAULT CHARACTER SET = latin1;  
  
Customer Reviews:  
CREATE TABLE `ecommercedb`.`CustomerReviews` (

`CustomerReviewID` INT NOT NULL ,

`ItemRate` VARCHAR(45) NOT NULL ,

`ReviewNotes` VARCHAR(45) NOT NULL ,

`ReviewInventoryID` INT NOT NULL ,

`ReviewCustomerId` VARCHAR(20) NOT NULL ,

PRIMARY KEY (`CustomerReviewID`) ,

INDEX `ReviewCustomerID\_idx` (`ReviewCustomerId` ASC) ,

INDEX `ReviewInventoryID\_idx` (`ReviewInventoryID` ASC) ,

CONSTRAINT `ReviewCustomerID`

FOREIGN KEY (`ReviewCustomerId` )

REFERENCES `ecommercedb`.`customer` (`CustomerID` )

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `ReviewInventoryID`

FOREIGN KEY (`ReviewInventoryID` )

REFERENCES `ecommercedb`.`inventoryitem` (`InventoryItemID` )

ON DELETE NO ACTION

ON UPDATE NO ACTION)

DEFAULT CHARACTER SET = latin1;  
  
Cart Details:  
CREATE TABLE `ecommercedb`.`CartDetails` (

`CartID` INT NOT NULL ,

`InventoryID` INT NOT NULL ,

`CartState` VARCHAR(45) NOT NULL ,

`Timestamp` TIMESTAMP NULL ,

INDEX `CartID\_idx` (`CartID` ASC) ,

INDEX `InventoryID\_idx` (`InventoryID` ASC) ,

CONSTRAINT `CartID`

FOREIGN KEY (`CartID` )

REFERENCES `ecommercedb`.`shoppingcart` (`ShoppingCartID` )

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `InventoryID`

FOREIGN KEY (`InventoryID` )

REFERENCES `ecommercedb`.`inventoryitem` (`InventoryItemID` )

ON DELETE RESTRICT

ON UPDATE RESTRICT)

DEFAULT CHARACTER SET = latin1;  
  
Address for warehouse and shipping:  
CREATE TABLE `ecommercedb`.`Address` (

`AddressID` INT NOT NULL ,

`ApartmentNumber` VARCHAR(45) NOT NULL ,

`StreetName` VARCHAR(45) NOT NULL ,

`City` VARCHAR(45) NOT NULL ,

`State` VARCHAR(45) NOT NULL ,

`Zipcode` VARCHAR(45) NOT NULL ,

`LandMark` VARCHAR(45) NULL ,

PRIMARY KEY (`AddressID`) )

DEFAULT CHARACTER SET = latin1;  
  
Warehouse:  
CREATE TABLE `ecommercedb`.`Warehouse` (

`WarehouseID` INT NOT NULL ,

`WarehouseName` VARCHAR(45) NOT NULL ,

`WarehouseRevenue` DECIMAL(15,2) NOT NULL ,

`WarehouseExpenditure` DECIMAL(15,2) NOT NULL ,

`WarehouseAddressID` INT NOT NULL ,

PRIMARY KEY (`WarehouseID`) )

DEFAULT CHARACTER SET = latin1;  
  
ShippingStatus:  
CREATE TABLE `ecommercedb`.`ShippingStatus` (

`ShippingStatusID` INT NOT NULL ,

`ShippingStatus` VARCHAR(45) NOT NULL ,

PRIMARY KEY (`ShippingStatusID`) )

DEFAULT CHARACTER SET = latin1;  
  
Shipping:  
CREATE TABLE `ecommercedb`.`Shipping` (

`ShippingID` INT NOT NULL ,

`ShippingMethod` VARCHAR(45) NOT NULL ,

`ShippingCharge` DECIMAL(15,2) NOT NULL ,

`ShippingDate` DATE NOT NULL ,

`ShippingStateID` INT NOT NULL ,

`ShippingAddressID` INT NOT NULL ,

PRIMARY KEY (`ShippingID`) ,

INDEX `ShippingStateID\_idx` (`ShippingStateID` ASC) ,

INDEX `ShippingAddressID\_idx` (`ShippingAddressID` ASC) ,

CONSTRAINT `ShippingStateID`

FOREIGN KEY (`ShippingStateID` )

REFERENCES `ecommercedb`.`shippingstatus` (`ShippingStatusID` )

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `ShippingAddressID`

FOREIGN KEY (`ShippingAddressID` )

REFERENCES `ecommercedb`.`address` (`AddressID` )

ON DELETE NO ACTION

ON UPDATE NO ACTION)

DEFAULT CHARACTER SET = latin1;  
  
Order:  
CREATE TABLE `ecommercedb`.`Order` (

`OrderID` INT NOT NULL ,

`OrderDate` DATE NOT NULL ,

`OrderStatus` VARCHAR(45) NOT NULL ,

`OrderTotalAmount` DECIMAL(15,2) NOT NULL ,

`PaidAmount` DECIMAL(15,2) NOT NULL ,

`BalanceDue` DECIMAL(15,2) NOT NULL ,

`CustomerOrderID` VARCHAR(20) NOT NULL ,

PRIMARY KEY (`OrderID`) ,

INDEX `CustomerOrderID\_idx` (`CustomerOrderID` ASC) ,

CONSTRAINT `CustomerOrderID`

FOREIGN KEY (`CustomerOrderID` )

REFERENCES `ecommercedb`.`customer` (`CustomerID` )

ON DELETE RESTRICT

ON UPDATE RESTRICT)

DEFAULT CHARACTER SET = latin1;  
  
Invoice:  
CREATE TABLE `ecommercedb`.`Invoice` (

`InvoiceID` INT NOT NULL ,

`InvoiceDate` DATE NOT NULL ,

`DueDate` DATE NOT NULL ,

`AmountDue` DECIMAL(15,2) NOT NULL ,

`AmountReceived` DECIMAL(15,2) NOT NULL ,

`BalanceDue` DECIMAL(15,2) NOT NULL ,

`InvoiceCustomerID` VARCHAR(20) NOT NULL ,

PRIMARY KEY (`InvoiceID`) ,

INDEX `InvoiceCustomerID\_idx` (`InvoiceCustomerID` ASC) ,

CONSTRAINT `InvoiceCustomerID`

FOREIGN KEY (`InvoiceCustomerID` )

REFERENCES `ecommercedb`.`customer` (`CustomerID` )

ON DELETE RESTRICT

ON UPDATE RESTRICT)

DEFAULT CHARACTER SET = latin1;  
  
OrderItem:  
CREATE TABLE `ecommercedb`.`OrderItem` (

`OrderItemNumber` INT NOT NULL ,

`Quantity` INT NOT NULL ,

`OrderNumber` INT NOT NULL ,

`InventoryItemNumber` INT NOT NULL ,

PRIMARY KEY (`OrderItemNumber`) ,

INDEX `OrderNumber\_idx` (`OrderNumber` ASC) ,

INDEX `InventoryItemNumber\_idx` (`InventoryItemNumber` ASC) ,

CONSTRAINT `OrderNumber`

FOREIGN KEY (`OrderNumber` )

REFERENCES `ecommercedb`.`order` (`OrderID` )

ON DELETE RESTRICT

ON UPDATE RESTRICT,

CONSTRAINT `InventoryItemNumber`

FOREIGN KEY (`InventoryItemNumber` )

REFERENCES `ecommercedb`.`inventoryitem` (`InventoryItemID` )

ON DELETE RESTRICT

ON UPDATE RESTRICT)

DEFAULT CHARACTER SET = latin1;  
  
Dependent:  
CREATE TABLE `ecommercedb`.`Dependent` (

`EmployeeID` VARCHAR(20) NOT NULL ,

`DependentName` VARCHAR(45) NOT NULL ,

`DependentAge` TINYINT NOT NULL ,

`DependentRelation` VARCHAR(45) NOT NULL ,

`DependentGender` VARCHAR(45) NOT NULL ,

PRIMARY KEY (`EmployeeID`) ,

CONSTRAINT `EmployeeID`

FOREIGN KEY (`EmployeeID` )

REFERENCES `ecommercedb`.`employee` (`EmployeeID` )

ON DELETE RESTRICT

ON UPDATE RESTRICT)

DEFAULT CHARACTER SET = latin1;  
  
DepartmentType:  
CREATE TABLE `ecommercedb`.`DepartmentType` (

`DepartmentTypeID` INT NOT NULL ,

`DepartmentTypeName` VARCHAR(45) NOT NULL ,

`DepartmentTypeDetails` VARCHAR(45) NOT NULL ,

PRIMARY KEY (`DepartmentTypeID`) )

DEFAULT CHARACTER SET = latin1;  
  
Department:  
CREATE TABLE `ecommercedb`.`Department` (

`DepartmentID` INT NOT NULL ,

`DepartmentName` VARCHAR(45) NOT NULL ,

`DepartmentManager` VARCHAR(45) NOT NULL ,

`DepartmentTypeID` INT NOT NULL ,

`DepartmentEmployeeID` VARCHAR(20) NOT NULL ,

PRIMARY KEY (`DepartmentID`) ,

INDEX `DepartmentTypeID\_idx` (`DepartmentTypeID` ASC) ,

INDEX `DepartmentEmployeeID\_idx` (`DepartmentEmployeeID` ASC) ,

CONSTRAINT `DepartmentTypeID`

FOREIGN KEY (`DepartmentTypeID` )

REFERENCES `ecommercedb`.`departmenttype` (`DepartmentTypeID` )

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `DepartmentEmployeeID`

FOREIGN KEY (`DepartmentEmployeeID` )

REFERENCES `ecommercedb`.`employee` (`EmployeeID` )

ON DELETE NO ACTION

ON UPDATE NO ACTION)

DEFAULT CHARACTER SET = latin1;

Grant :

Admin:-

CREATE USER 'admin'@'localhost' IDENTIFIED BY 'admin';

GRANT ALL ON ecommercedb.\* TO 'admin'@'localhost';

GRANT SELECT ON ecommercedb.\* TO 'admin'@'localhost';

grant update on ecommercedb.employee to 'admin'@'localhost';

grant update on ecommercedb.person to 'admin'@'localhost';

grant update on ecommercedb.product to 'admin'@'localhost';

grant update on ecommercedb.productitemcategory to 'admin'@'localhost';

grant update on ecommercedb.department to 'admin'@'localhost';

grant update on ecommercedb.departmenttype to 'admin'@'localhost';

grant insert on ecommercedb.department to 'admin'@'localhost';

grant insert on ecommercedb.departmenttype to 'admin'@'localhost';

grant insert on ecommercedb.product to 'admin'@'localhost';

grant insert on ecommercedb.productitemcategory to 'admin'@'localhost';

GRANT USAGE ON \*.\* TO 'admin'@'localhost' WITH MAX\_QUERIES\_PER\_HOUR 90;

CREATE USER 'HR'@'localhost' IDENTIFIED BY 'HR';

GRANT ALL ON ecommercedb.employee TO 'HR'@'localhost';

GRANT SELECT ON ecommercedb.person TO 'HR'@'localhost';

GRANT SELECT ON ecommercedb.useraccount TO 'HR'@'localhost';

GRANT SELECT ON ecommercedb.department TO 'HR'@'localhost';

GRANT SELECT ON ecommercedb.dependent TO 'HR'@'localhost';

GRANT UPDATE ON ecommercedb.employee TO 'HR'@'localhost';

GRANT USAGE ON \*.\* TO 'HR'@'localhost' WITH MAX\_QUERIES\_PER\_HOUR 90;  
  
CREATE USER 'warehouse'@'localhost' IDENTIFIED BY 'warehouse';

GRANT ALL ON ecommercedb.shippingwarehouse TO 'warehouse'@'localhost';

GRANT SELECT ON ecommercedb.shippingstatus TO 'warehouse'@'localhost';

GRANT SELECT ON ecommercedb.customerorder TO 'warehouse'@'localhost';

GRANT SELECT ON ecommercedb.inventoryitem TO 'warehouse'@'localhost';

GRANT UPDATE ON ecommercedb.inventoryitem TO 'warehouse'@'localhost';

GRANT INSERT ON ecommercedb.inventoryitem TO 'warehouse'@'localhost';

GRANT USAGE ON \*.\* TO 'warehouse'@'localhost' WITH MAX\_QUERIES\_PER\_HOUR 90;

CREATE USER 'shipping'@'localhost' IDENTIFIED BY 'shipping';

GRANT ALL ON ecommercedb.shipping TO 'shipping'@'localhost';

GRANT SELECT ON ecommercedb.shippingstatus TO 'shipping'@'localhost';

GRANT SELECT ON ecommercedb.customerorder TO 'shipping'@'localhost';

GRANT SELECT ON ecommercedb.inventoryitem TO 'shipping'@'localhost';

GRANT UPDATE, insert on ecommercedb.shipping TO 'shipping'@'localhost';

GRANT INSERT ON ecommercedb.shippingstatus TO 'shipping'@'localhost';

GRANT USAGE ON \*.\* TO 'warehouse'@'localhost' WITH MAX\_QUERIES\_PER\_HOUR 90;

Trigger:

USE `ecommercedb`;

DELIMITER $$

CREATE TRIGGER `checkCustomerorder`

AFTER INSERT ON customerorder

FOR EACH ROW

-- Edit trigger body code below this line. Do not edit lines above this one

BEGIN

IF new.OrderTotalAmount> 400 THEN

UPDATE customer SET CustomerTypeID=2

WHERE customer.CustomerID= new.CustomerOrderID;

END IF;

END;