CREATE SCHEMA IF NOT EXISTS `Capstone` DEFAULT CHARACTER SET utf8 ;

USE `Capstone` ;

-- -----------------------------------------------------

-- Table `Capstone`.`Customer`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `Capstone`.`Customer` (

`CustomerID` INT NOT NULL AUTO\_INCREMENT,

`CustomerName` VARCHAR(45) NOT NULL,

`CustomerAddress` VARCHAR(45) NOT NULL,

`CustomerCity` VARCHAR(45) NOT NULL,

`CustomerState` VARCHAR(2) NOT NULL,

`CustomerZip` INT(5) NOT NULL,

`CustomerContactName` VARCHAR(45) NOT NULL,

`CustomerPhone` VARCHAR(45) NOT NULL,

PRIMARY KEY (`CustomerID`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `Capstone`.`Inventory`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `Capstone`.`Inventory` (

`InventoryID` INT NOT NULL AUTO\_INCREMENT,

`InventoryTitle` VARCHAR(45) NOT NULL,

`InventoryPrice` DECIMAL(2) NOT NULL,

PRIMARY KEY (`InventoryID`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `Capstone`.`Employee`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `Capstone`.`Employee` (

`EmployeeID` INT NOT NULL AUTO\_INCREMENT,

`EmployeeName` VARCHAR(45) NOT NULL,

`EmployeePassword` VARCHAR(45) NOT NULL,

`EmployeePhone` VARCHAR(45) NOT NULL,

`EmployeeStatus` VARCHAR(45) NOT NULL,

`EmployeeAddress` VARCHAR(45) NOT NULL,

`EmployeeCity` VARCHAR(45) NOT NULL,

`EmployeeState` VARCHAR(2) NOT NULL,

`EmployeeZip` VARCHAR(5) NOT NULL,

PRIMARY KEY (`EmployeeID`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `Capstone`.`Task`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `Capstone`.`Task` (

`TaskID` INT NOT NULL AUTO\_INCREMENT,

`CustomerID` INT NOT NULL,

`EmployeeID` INT NOT NULL,

`TaskStatus` VARCHAR(45) NOT NULL,

`TaskDescription` VARCHAR(200) NOT NULL,

`TaskResolution` VARCHAR(200) NULL,

PRIMARY KEY (`TaskID`),

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

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

CONSTRAINT `CustomerID`

FOREIGN KEY (`CustomerID`)

REFERENCES `Capstone`.`Customer` (`CustomerID`)

ON DELETE NO ACTION

ON UPDATE CASCADE,

CONSTRAINT `EmployeeID`

FOREIGN KEY (`EmployeeID`)

REFERENCES `Capstone`.`Employee` (`EmployeeID`)

ON DELETE NO ACTION

ON UPDATE CASCADE)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `Capstone`.`TaskLineItem`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `Capstone`.`TaskLineItem` (

`TaskLineItemID` INT NOT NULL AUTO\_INCREMENT,

`InventoryID` INT NOT NULL,

`TaskID` INT NOT NULL,

`TaskLineItemQuant` INT NOT NULL,

PRIMARY KEY (`TaskLineItemID`),

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

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

CONSTRAINT `InventoryID`

FOREIGN KEY (`InventoryID`)

REFERENCES `Capstone`.`Inventory` (`InventoryID`)

ON DELETE NO ACTION

ON UPDATE CASCADE,

CONSTRAINT `TaskID`

FOREIGN KEY (`TaskID`)

REFERENCES `Capstone`.`Task` (`TaskID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `Capstone`.`TaskTimeStamp`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `Capstone`.`TaskTimeStamp` (

`TaskTimeStampID` INT NOT NULL AUTO\_INCREMENT,

`TaskID` INT NOT NULL,

`TaskTimeStampStart` DATETIME NULL,

`TaskTimeStampEnd` DATETIME NULL,

`TaskStatus` VARCHAR(45) NOT NULL,

`TaskTimeStampMiles` INT NOT NULL,

`TaskTimeStampCreated` VARCHAR(45) NOT NULL,

PRIMARY KEY (`TaskTimeStampID`),

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

CONSTRAINT `TaskID`

FOREIGN KEY (`TaskID`)

REFERENCES `Capstone`.`Task` (`TaskID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `Capstone`.`InventoryOrder`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `Capstone`.`InventoryOrder` (

`OrderID` INT NOT NULL AUTO\_INCREMENT,

`TaskID` INT NOT NULL,

`OrderDate` DATETIME NOT NULL,

`OrderReceived` DATETIME NULL,

`OrderStatus` VARCHAR(45) NOT NULL,

PRIMARY KEY (`OrderID`),

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

CONSTRAINT `TaskID`

FOREIGN KEY (`TaskID`)

REFERENCES `Capstone`.`Task` (`TaskID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `Capstone`.`OrderLineItem`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `Capstone`.`OrderLineItem` (

`OrderLineItemID` INT NOT NULL AUTO\_INCREMENT,

`OrderID` INT NOT NULL,

`InventoryID` INT NOT NULL,

`OrderLineItemQuantity` INT NOT NULL,

PRIMARY KEY (`OrderLineItemID`),

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

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

CONSTRAINT `InventoryID`

FOREIGN KEY (`InventoryID`)

REFERENCES `Capstone`.`Inventory` (`InventoryID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `OrderID`

FOREIGN KEY (`OrderID`)

REFERENCES `Capstone`.`InventoryOrder` (`OrderID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `Capstone`.`TaskTranfered`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `Capstone`.`TaskTranfered` (

`TaskTranferedID` INT NOT NULL AUTO\_INCREMENT,

`TaskID` INT NOT NULL,

`TaskTranferedFrom` INT NOT NULL,

`TaskTranferedTo` INT NOT NULL,

`TaskTranferedBy` INT NOT NULL,

`TaskTranferedTimestamp` DATETIME NOT NULL,

PRIMARY KEY (`TaskTranferedID`),

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

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

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

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

CONSTRAINT `TaskID`

FOREIGN KEY (`TaskID`)

REFERENCES `Capstone`.`Task` (`TaskID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `TaskTranferedFrom`

FOREIGN KEY (`TaskTranferedFrom`)

REFERENCES `Capstone`.`Employee` (`EmployeeID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `TaskTranferedTo`

FOREIGN KEY (`TaskTranferedTo`)

REFERENCES `Capstone`.`Employee` (`EmployeeID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `TaskTranferedBy`

FOREIGN KEY (`TaskTranferedBy`)

REFERENCES `Capstone`.`Employee` (`EmployeeID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `Capstone`.`InventoryTransfer`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `Capstone`.`InventoryTransfer` (

`InventoryTransferID` INT NOT NULL AUTO\_INCREMENT,

`InventoryID` INT NOT NULL,

`InventoryTransferFrom` INT NOT NULL,

`InventoryTransferTo` INT NOT NULL,

`InventoryTransferBy` INT NOT NULL,

`InventoryTransferDate` DATETIME NOT NULL,

`InventoryTransferQuantity` INT NOT NULL,

PRIMARY KEY (`InventoryTransferID`),

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

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

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

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

CONSTRAINT `InventoryID`

FOREIGN KEY (`InventoryID`)

REFERENCES `Capstone`.`Inventory` (`InventoryID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `InventoryTransferFrom`

FOREIGN KEY (`InventoryTransferFrom`)

REFERENCES `Capstone`.`Employee` (`EmployeeID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `InventoryTransferTo`

FOREIGN KEY (`InventoryTransferTo`)

REFERENCES `Capstone`.`Employee` (`EmployeeID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `InventoryTransferBy`

FOREIGN KEY (`InventoryTransferBy`)

REFERENCES `Capstone`.`Employee` (`EmployeeID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `Capstone`.`InStock`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `Capstone`.`InStock` (

`InStockID` INT NOT NULL AUTO\_INCREMENT,

`InStockInventoryID` INT NOT NULL,

`InStockEmployeeID` INT NOT NULL,

`InStockQuantity` INT NOT NULL,

PRIMARY KEY (`InStockID`),

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

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

CONSTRAINT `InStockInventoryID`

FOREIGN KEY (`InStockInventoryID`)

REFERENCES `Capstone`.`Inventory` (`InventoryID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `InStockEmployeeID`

FOREIGN KEY (`InStockEmployeeID`)

REFERENCES `Capstone`.`Employee` (`EmployeeID`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;