Clothing Store System UML Class Diagram Address -street : String <<set, get>> **OrderStatusNotifier** -city: String <<set, get>> <<interface>> -state : String <<set, get>> -order: IOrder Notifier -zipcode : String <<set, get>> + update(): void + OrderStatusNotifier(IOrder : order): +Address(street:String, city:String, + update(): void state:String, zipcode:String) Order <<interface>> **IAddress** -orderNumber: String <<set, get>> -purchasedClothing: List<Clothing> +setStreet(street : String) : void OrderReceipt <<set, get>> +setCity(city : String) : void -orderTotal : float <<set, get>> +setState(state : String) : void -totalPrice: float -orderStatus : String <<set, get>> +setZipcode(zipcode : String) : void -customer: ICustomer +getStreet() : String +getCity() : String +visitOrder(Order orderE): void +getState(): String +visitCollection(Collection +Order(orderNumber: String, +getZipcode() : String collection): void puchasedClothing: List<Clothing>, +visitShirtShirt ShirtE): void orderTotal: float, orderStatus: String +visitPants(Pants pantsE): void Customer -firstName : String <<set, get>> <<interface>> -lastName : String <<set, get>> **IOrderReceipt** <<interface>> -email: String <<set, get>> **IOrder** -shippingAddress: IAddress <<set, get>> +visitOrder(Order orderE): void +getOrderNumber(): String -orderHistory: Collection<IOrder> <<set,get>> +visitCollection(Collection collection): void +getPurchasedClothing(): List<Clothing> +visitShirtShirt ShirtE): void +getOrderTotal(): float +Customer(firstName:String, lastName:String, +visitPants(Pants pantsE): void +getOrderStatus(): String email:String, shippingAddress:IAddress, +setOrderTotal(float total): void orderHistory:Collection<IOrder> +accept(IOrderReceipt visitor): void +addOrder(order: IOrder) +addClothhing(Clothing c): void +placeOrder(): void +shipOrder(): void +deliverOrder(): void <<interface>> **ICustomer** + addOrder(IOrder order): void + getFirstName(): String + getLastName(): String + getEmail(): String + getShippingAddress(): IAddress + getOrderHistory(): List<IOrder> Clothing <<interface>> Department #department : Department +findShirt(): void +Clothing(department : Department) +findPants(): void +find(): void +accept (IOrderReceipt visitor): void **Pants** -productNumber : String <<set, get>> -productNumber : String <<set, get>> -price : float <<set, get>> -price : float <<set, get>> +findShirt(): void -size : String <<set, get>> -size : int <<set, get>> +findPants(): void -color : String <<set, get>> -color : String <<set, get>> -pantLength : String <<set, get>> -sleeveLength : String <<set, get>> +Shirt(productNumber:String, price:float, +Pants(productNumber:String, price:float, Womens size:String, color:String, sleeveLength:String, size:int, color:String, pantLength:String, department:Department) department:Department)

+ find(): void

+accept(IOrderReceipt visitor): void

+findShirt(): void

+findPants(): void

+find() : void

+accept(IOrderReceipt visitor): void