

Package `org.bookstore.order.service`

Class OrderService

`java.lang.Object`
`org.bookstore.order.service.OrderService`

`public class OrderService`
`extends Object`

The class OrderService defines a service to manage the orders of a bookstore.

Constructor Summary

Constructors

Constructor	Description
<code>OrderService()</code>	

Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method	Description
<code>void</code>	<code>cancelOrder(long id)</code>	Tries to cancel an order.
<code>Order</code>	<code>findOrder(long id)</code>	Finds an order by identifier.
<code>Order</code>	<code>placeOrder(long customerId, List <OrderItem> items)</code>	Places an order with the bookstore.
<code>List <OrderInfo></code>	<code>searchOrders(long customerId, int year)</code>	Searches for orders by customer and year.

Methods inherited from class `java.lang.Object`

`clone` , `equals` , `finalize` , `getClass` , `hashCode` , `notify` , `notifyAll` , `toString` , `wait` , `wait` , `wait`

Constructor Details

OrderService

```
public OrderService()
```

Method Details

placeOrder

```
public Order placeOrder(long customerId,
                        List <OrderItem> items)
    throws CustomerNotFoundException,
           PaymentFailedException
```

Places an order with the bookstore.

Parameters:

`customerId` - the identifier of the customer

`items` - the items to be ordered

Returns:

the data of the placed order

Throws:

`CustomerNotFoundException` - if no customer with the specified identifier exists

`PaymentFailedException` - if the credit card payment failed

findOrder

```
public Order findOrder(long id)
    throws OrderNotFoundException
```

Finds an order by identifier.

Parameters:

`id` - the identifier of the order

Returns:

the data of the found order

Throws:

`OrderNotFoundException` - if no order with the specified identifier exists

searchOrders

```
public List <OrderInfo> searchOrders(long customerId,
                                     int year)
    throws CustomerNotFoundException
```

Searches for orders by customer and year.

Parameters:

`customerId` - the identifier of the customer

`year` - the year of the orders

Returns:

the matching orders

Throws:

`CustomerNotFoundException` - if no customer with the specified identifier exists

cancelOrder

```
public void cancelOrder(long id)
    throws OrderNotFoundException,
           OrderAlreadyShippedException
```

Tries to cancel an order.

Parameters:

`id` - the identifier of the order

Throws:

`OrderNotFoundException` - if no order with the specified identifier exists

`OrderAlreadyShippedException` - if the order has already been shipped