

```
Contract ProcessSaleService::enterItem(barcode : Integer, quantity : Integer) : Boolean {
```

```
/*
```

```
 * Generated by RM2DOc - Definition
```

```
 * item is the object i in the instance set of class Item. i represents an object of class Item, and i meets:
```

```
 *   The attribute Barcode of the object i is equal to barcode
```

```
*/
```

```
definition:
```

```
    item:Item = Item.allInstance()->any(i:Item | i.Barcode = barcode)
```

```
/*
```

```
 * Generated by RM2DOc - Precondition
```

```
 * CurrentSale exists
```

```
 * The attribute IsComplete of the object CurrentSale is equal to false
```

```
 * item exists
```

```
 * The attribute StockNumber of the object item is greater than 0
```

```
*/
```

```
precondition:
```

```
    CurrentSale.oclIsUndefined() = false and
```

```
    CurrentSale.IsComplete = false and
```

```
    item.oclIsUndefined() = false and
```

```
    item.StockNumber > 0
```

```
/*
```

```
 * Generated by RM2DOc - Postcondition
```

```
 * sli represented the object of class SalesLineItem
```

```
 * The object sli was created
```

```
 * The object CurrentSaleLine became sli
```

```
 * The object sli was linked to the object CurrentSale by BelongedSale
```

```
 * The object sale was linked to the object saleslineitem by ContainedSalesLine
```

```
 * The attribute Quantity of the object sli became quantity
```

```
 * The object sli was linked to the object item by BelongedItem
```

```
 * The attribute StockNumber of the object item became the previous value of the attribute StockNumber of the object item minus quantity
```

```
 * The attribute Subamount of the object sli became the attribute Price of the object item times quantity
```

```
 * The object sli was put into the instance set of class SalesLineItem
```

```
 * The return value was true
```

```
*/
```

```
postcondition:
```

```
    let sli:SalesLineItem = SalesLineItem.newInstance() in
```

```
    sli.oclIsNew() and
```

```
    self.CurrentSaleLine = sli and
```

```
    sli.BelongedSale = CurrentSale and
```

```
    CurrentSale.ContainedSalesLine->includes(sli) and
```

```
    sli.Quantity = quantity and
```

```
    sli.BelongedItem = item and
```

```
    item.StockNumber = item.StockNumber@pre - quantity and
```

```
    sli.Subamount = item.Price * quantity and
```

```
    SalesLineItem.allInstance()->includes(sli) and
```

```
    result = true
```

```
}
```