```java
/* ************************************************************************ *\
*                        Programmierung 1 HS 2018 - Serie 3-1                *
*                Jonas Gehrlein (15-127-541) & Jan Dietrich (10-100-436)     *
*                                  File: Order.java                          *
\* ************************************************************************ */

public class Order
{
    private String customerName, customerAddress;
    static int order_id = 0;
    private int counter = 0;
    private String order_string;
    public int i = 0;
    private int total_price;
    public int cost;

    Book book1 = new Book();
    Book book2 = new Book();
    Book book3 = new Book();
    Book book4 = new Book();
    Book book5 = new Book();


    // Constructor which initializes the Order objects. It increases the order_id with each initialization.
    public Order()
    {
        order_id++;
    }
    // Set-Methods
    public String setCustomerName(String input_customer_name) {
    customerName = input_customer_name;

    return customerName;
    }


    public String setCustomerAddress(String input_customer_address) {
    customerAddress = input_customer_address;

    return customerAddress;
    }

    /* addBook Method. For Each iteration a new book variable will be filled with a book from the method.
    It is not very elegant but it works. In addition it fills the cost variables with the price of the book. */

    public void addBook(Book new_book) {
        if (i == 0) {
            book1 = new_book;
            cost = book1.getPrice();
        }
        if (i == 1) {
            book2 = new_book;
            cost = book2.getPrice() + cost;
        }
        if (i == 2) {
            book3 = new_book;
            cost = book3.getPrice() + cost;
        }
        if (i == 3) {
            book4 = new_book;
            cost = book4.getPrice() + cost;
        }
        if (i == 4) {
            book5 = new_book;
            cost = book5.getPrice() + cost;
        }
        i++;
        return;
    }
    // Calculates the total cost of the order which depends on the amount of books
```

```
        added to the order.
71      public int getTotal_price() {
72         return cost;
73      }
74
75
76      // Some getMethods.
77      public String getCustomerName() { return customerName; }
78      public String getCustomerAddress() { return customerAddress; }
79      public int getOrder_id(){return order_id;}
80
81      // Not very clean, I know. It works for Test.java but would not work for orders
        of 2, 3, 4 books. Solution would be more arguments of "if".
82      // If I had more time, I would add some formatting to the prices. Now it works
        manually.
83      public String toString ()
84      {
85          String order_string;
86          if (i>1 ) {
87              order_string = "Order id: " + order_id + ", " + "Customer: " +
                customerName + ", " + customerAddress + "\n" +
88                      book1.toString() + ", " + book1.getPrice() + " CHF" + "\n" +
89                      book2.toString() + ", " + book2.getPrice() + " CHF" + "\n" +
90                      book3.toString() + ", " + book3.getPrice() + " CHF" + "\n" +
91                      book4.toString() + ", " + book4.getPrice() + " CHF" + "\n" +
92                      book5.toString() + ", " + book5.getPrice() + " CHF" + "\n" +
93                      "Total price: " + getTotal_price() + " CHF" + "\n";
94          }
95          else
96              order_string = "Order id: " + order_id + ", " + "Customer: " +
                customerName + ", " + customerAddress + "\n" +
97                      book1.toString() + ", " + book1.getPrice() + " CHF" + "\n" +
                        "Total price: " + getTotal_price() + " CHF" + "\n";
98
99
100         return order_string;
101     }
102
103     public int getI()
104     {
105         return i;
106     }
107
108 }
```