

```
Customer.java *DigitalItemOrder.java MusicOrder.java Tester.java Solution.java X
```

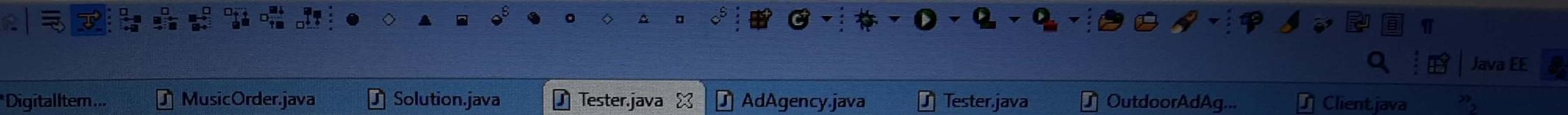
1 package jewel;
2
3 public class Solution {
4 double billAmount = 0.0d;
5 double amount = 0.0d;
6 public double calBillAmount(String[] availJewelArr,int[] availPriceArr,String[] purchasedJewelArr,int[] purchasedQuantityArr) {
7 for(int i=0; i< availJewelArr.length; i++) {
8 for(int j=0; j<purchasedJewelArr.length; j++) {
9 if(availJewelArr[i] == purchasedJewelArr[j]) {
10 amount = availPriceArr[i] * purchasedQuantityArr[j];
11 billAmount += amount;
12 }
13 else {
14 billAmount = -1;
15 }
16 if(billAmount > 20000) {
17 billAmount = billAmount - (billAmount*3/100);
18 }
19 }
20 }
21 return billAmount;
22 }
23 }
24

Console X Progress Coverage Debug

<terminated> Tester (13) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (10-Apr-2021, 2:38:49 am)
Total Price : 27.3
V1001

Writable Smart Insert 1:1:0 191M of 256M

Project Run Window Help



```
1 package jewel;
2
3 public class Tester {
4
5     public static void main(String[] args) {
6         String[] availJewelArr = {"Bentex", "Silver", "Gold", "Platinum"};
7         int[] availPriceArr = {20, 50, 2600, 3000};
8         String[] purchasedJewelArr = {"Silver", "Gold", "Platinum"};
9         int[] purchasedQuantityArr = {20, 7, 3};
10        Solution s= new Solution();
11        s.calBillAmount(availJewelArr , availPriceArr , purchasedJewelArr , purchasedQuantityArr);
12        System.out.println("Bill Amount: " + s.calBillAmount(availJewelArr,availPriceArr,purchasedJewelArr,purchasedQuantityArr));
13    }
14
15
16 }
17
```

Console Progress Coverage Debug

terminated> Tester (12) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (10-Apr-2021, 2:55:52 pm)

Booking Id: S101

Bill amount: 432250

Writable

Smart Insert

1:1:0

148M of 256M



Customer.java *DigitalItemOrder.java MusicOrder.java Tester.java

```
1 package digitalArts;
2
3 public class Customer {
4     private String custName;
5     private String emailId;
6     private int age;
7
8     public Customer(String custName, String emailId, int age) {
9         this.custName = custName;
10        this.emailId = emailId;
11        this.age = age;
12    }
13
14     public boolean validateCustomerDetails() {
15         String x= "[A-Z]{1}[a-z]{3,}";
16         String y= "[a-z1-9]{1,}(@){1}[a-z1-9]{1,}(.com|.edu)";
17         if(this.custName.matches(x) & this.age>18 & this.emailId.matches(y)) {
18             return true;
19         }
20         else
21             return false;
22     }
23
24     public String getCustName() {
25         return custName;
26     }
27
28     public void setCustName(String custName) {
29         this.custName = custName;
30     }
31
32     public String getEmailId() {
33         return emailId;
34     }
```

Console Progress Coverage Debug

<terminated> Tester (13) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (10-Apr-2021, 3:15:05 pm)

Total Price : 16.8

Writable

Smart Insert

17:75:469

109M of 256M



03:15 PM
10-04-2021 ENG

```
Customer.java *DigitalItemOrder.java MusicOrder.java Tester.java
8
24 public String getCustName() {
25     return custName;
26 }
27
28 public void setCustName(String custName) {
29     this.custName = custName;
30 }
31
32 public String getEmailId() {
33     return emailId;
34 }
35
36 public void setEmailId(String emailId) {
37     this.emailId = emailId;
38 }
39
40 public int getAge() {
41     return age;
42 }
43
44 public void setAge(int age) {
45     this.age = age;
46 }
47
48
49 @Override
50 public String toString() {
51     return "Customer [custName=" + custName + ", emailId=" + emailId + ", age=" + age + "]";
52 }
53
54
55 }
56
```

Console Progress Coverage Debug

<terminated> Tester (13) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (10-Apr-2021, 3:15:05 pm)
Total Price : 16.8

Writable

Smart Insert

17: 75: 469

115M of 256M

Scanned with CamScanner

Customer.java *DigitalItemOrder.java MusicOrder.java Tester.java

```
1 package digitalArts;
2
3 abstract public class DigitalItemOrder {
4     private Customer customer;
5     private String itemName;
6     private double orderPrice;
7     private String orderId;
8     private static int counter;
9     static {
10         DigitalItemOrder.counter=1001;
11     }
12     public DigitalItemOrder(Customer customer, String itemName) {
13         this.customer = customer;
14         this.itemName = itemName;
15     }
16
17     public void generateOrderId() {
18         this.orderId = itemName.charAt(0) + "" + counter++;
19     }
20
21     abstract public void calculateOrderPrice();
22     public Customer getCustomer() {
23         return customer;
24     }
25
26     public void setCustomer(Customer customer) {
27         this.customer = customer;
28     }
29     public String getItemName() {
30         return itemName;
31     }
32     public void setItemName(String itemName) {
33         this.itemName = itemName;
34     }
35 }
```

Console Progress Coverage Debug

<terminated> Tester (13) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (10-Apr-2021, 3:15:05 pm)

Total Price : 16.8

Writable Smart Insert 55 : 1 : 1276 118M of 256M

Scanned with CamScanner

Customer.java

*DigitalItemOrder.java

MusicOrder.java

Tester.java

```
23         return customer;
24     }
25
26     public void setCustomer(Customer customer) {
27         this.customer = customer;
28     }
29     public String getItemName() {
30         return itemName;
31     }
32     public void setItemName(String itemName) {
33         this.itemName = itemName;
34     }
35     public double getOrderPrice() {
36         return orderPrice;
37     }
38     public void setOrderPrice(double orderPrice) {
39         this.orderPrice = orderPrice;
40     }
41     public String getOrderId() {
42         return orderId;
43     }
44     public void setOrderId(String orderId) {
45         this.orderId = orderId;
46     }
47
48     @Override
49     public String toString() {
50         return "DigitalItemOrder [customer=" + customer + ", itemName=" + itemName + ", orderPrice=" + orderPrice
51             + ", orderId=" + orderId + "]";
52     }
53
54 }
55 }
```

Console Progress Coverage Debug

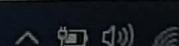
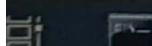
<terminated> Tester (13) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (10-Apr-2021, 3:15:05 pm)
Total Price : 16.8

Writable

Smart Insert

55:1:1276

121M of 256M



Customer.java *DigitalItemOrder.java MusicOrder.java Tester.java

```
2
3 public class MusicOrder extends DigitalItemOrder {
4     private String musicFormat;
5     private int bitRate;
6     private int durationInSec;
7     private static String[] musicFormatAvailable = {"mp3", "wav", "flac"};
8     private static int[] bitRateAvailable = {192, 128, 256};
9     private static double[] bitRateCost = {15.0, 10.0, 20.0};
10
11
12     public MusicOrder(Customer customer, String itemName, String musicFormat, int bitRate, int durationInSec) {
13         super(customer, itemName);
14         this.musicFormat = musicFormat;
15         this.bitRate = bitRate;
16         this.durationInSec = durationInSec;
17     }
18     public void calculateOrderPrice() {
19         double baseCost , orderPrice;
20         if(getCustomer().validateCustomerDetails()) {
21             for(int y=0; y < musicFormatAvailable.length ; y++) {
22                 if(musicFormatAvailable[y].equals(musicFormat)) {
23                     validateBitRate();
24                     if( validateBitRate() != -1 && this.durationInSec >0) {
25                         baseCost = bitRateCost[validateBitRate()];
26                         baseCost += ((this.durationInSec/60)*3);
27                         orderPrice = baseCost + ((baseCost*5)/100);
28                         generateOrderId();
29                         System.out.println("Total Price : " + orderPrice);
30                         break;
31                     }
32                 } else {
33                     orderPrice = -1.0;
34                     System.out.println("Total Price : " + orderPrice);
35                     generateOrderId("NA");
36                 }
37             }
38         }
39     }
40 }
```

Console Progress Coverage Debug

<terminated> Tester (13) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (10-Apr-2021, 3:15:05 pm)

Total Price : 16.8

Writable

Smart Insert

12:112:468

123M of 256M

^ < > < > ENG

```
Customer.java *DigitalItemOrder.java MusicOrder.java Tester.java
31     }
32     else {
33         orderPrice = -1.0;
34         System.out.println("Total Price : " + orderPrice);
35         setOrderId("NA");
36         break;
37     }
38 }
39 }
40 }
41 else {
42     orderPrice = -1.0;
43     System.out.println("Total Price : " + orderPrice);
44     setOrderId("NA");
45 }
46 }
47
48
49
50+    public Integer validateBitRate() {
51        int bitRateIndex = -1;
52        for(int i=0; i<bitRateAvailable.length ; i++) {
53            if(this.bitRate==(bitRateAvailable[i])) {
54                bitRateIndex= i;
55                break;
56            }
57            else {
58                bitRateIndex = -1;
59            }
60        }
61        return bitRateIndex;
62    }
63 }
```

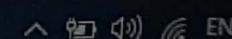
Console Progress Coverage Debug
<terminated> Tester (13) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (10-Apr-2021, 3:15:05 pm)
Total Price : 16.8

Writable

Smart Insert

12 : 112 : 468

127M of 256M



```
8
SE-1.8]     40      }
SE-1.8]     41      else {
SE-1.8]     42          orderPrice = -1.0;
SE-1.8]     43          System.out.println("Total Price : " + orderPrice);
SE-1.8]     44          setOrderId("NA");
SE-1.8]     45      }
SE-1.8]     46  }
SE-1.8]     47
SE-1.8]     48
SE-1.8]     49
SE-1.8] 50  public Integer validateBitRate() {
SE-1.8] 51      int bitRateIndex = -1;
SE-1.8] 52      for(int i=0; i<bitRateAvailable.length ; i++) {
SE-1.8] 53          if(this.bitRate==(bitRateAvailable[i])) {
SE-1.8] 54              bitRateIndex= i;
SE-1.8] 55              break;
SE-1.8] 56          }
SE-1.8] 57          else {
SE-1.8] 58              bitRateIndex = -1;
SE-1.8] 59          }
SE-1.8] 60      }
SE-1.8] 61      return bitRateIndex;
SE-1.8] 62  }
SE-1.8] 63
SE-1.8] 64  @Override
SE-1.8] 65  public String toString() {
SE-1.8] 66      return "MusicOrder [musicFormat=" + musicFormat + ", bitRate=" + bitRate + ", durationInSec=" + durationInSec
SE-1.8] 67      + "]";
SE-1.8] 68  }
SE-1.8] 69
SE-1.8] 70  }
SE-1.8] 71
SE-1.8] 72
```

Console X Progress Coverage Debug



<terminated> Tester (13) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (10-Apr-2021, 3:15:05 pm)

Total Price : 16.8

Writable

Smart Insert

12 : 112 : 468

128M of 256M

The screenshot shows a Java development environment with the following details:

- Code Editor:** The main window displays the `Tester.java` file. The code creates a `Customer` object with name "Nitin", email "nj@123.edu", and age 30. It then creates a `MusicOrder` object with type "Video", format "mp3", bit rate 128, and price 120. The `calculateOrderPrice()` method is called, and the `orderId` is printed.
- Console:** The bottom-left pane shows the output of the application's execution. It reads "Total Price : 16.8".
- Status Bar:** The bottom right shows the memory usage as "130M of 256M".

```
1 package digitalArts;
2
3 public class Tester {
4
5     public static void main(String[] args) {
6         Customer obj1 = new Customer("Nitin", "nj@123.edu", 30);
7         MusicOrder obj2 = new MusicOrder(obj1, "Video", "mp3", 128, 120);
8         obj2.calculateOrderPrice();
9         System.out.println(obj2.getOrderId());
10
11    }
12
13 }
14
15
```