Name	
QUID	
Total (100)	

Q.1 [100 Points] Code for the embedded comments:

```
package quizzes.quiz2;
public class Product {
	private String name;
	private int id;
	private double price;
	private int quantity;
	private boolean available =true;
	public static int serialId = 100;
	public String getName() { return name;    }
	public void setName(String name) { this.name = name;  }
	public int getId() { return id;     }
	private void setId(int id) { this.id = id;      }
	public int getQuantity() { return quantity;  }
	public void setQuantity(int quantity) { this.quantity = quantity; }
	public boolean isAvailable() { return available; }
	private void setAvailable(boolean available) { this.available = available;    }
	public double getPrice() { return price;  }
	/*A first constructor that receives only the name and the price.
	  The value of id is automatically set in the constructor when an object is
	  created. The value of id should be 101 for the first object, 102 for the
	  second object..etc. */
	public Product(String name, double price) {









	}
	/* A second constructor that receives the name, price, and quantity.
	 * This constructor makes use of the first constructor. */
	public Product(String name, double price, int quantity) {









	}
	/*This method returns a formatted string having id, name, price, and quantity */
	@Override
	public String toString() {









	}
```

```
/*This method returns the total number of created objects so far. */
public static int getCount() {




}
/*This method sets the price after checking that it is greater than 0.
  It returns true if the new price is accepted otherwise, it returns false.   */
public boolean setPrice(double price) {









}
/*This method increases the quantity by a value n. You need to verify that n is
  a positive number and if so, increase quantity by n, change available to true,
  and return true. However, if n is not positive, do not do any changes and just
  return false. */
public boolean increaseQuantity(int n) {









}
```

```
    /*This method decreases the quantity by a value n. You should check that the
      remaining quantity is not negative before decreasing the quantity otherwise,
      do not do any changes and just return false if the remaining quantity would
      become negative. If the remaining quantity is greater than zero, return true.
      If the remaining quantity is 0, available should be changed to false and
      return true. */
    public boolean decreaseQuantity(int n) {









    }
    /*This method increases the price by p%. p is integer in (0,100). */
    public void increasePrice(int p) {




    }
    /*This method increases the price by 10%. The method makes use of the method
      increasePrice that takes a percentage. */
    public void increasePrice() {




    }
    /*This method returns true if the price of this object is less than the price of
      other otherwise, it returns false.  */
    public boolean isCheaper(Product other) {




    }
}
```

```
package quizzes.quiz2;
public class ProductTester {
	public static void main(String[] args) {
		/*Create an anonymous object of ProductTester */


	}
	public ProductTester() {
		/*Create the object p1 of the Product class using the second constructor:
		  name is Grapes, price is 20.0, and quantity is 50 */


		/*Call the method decQuantity to decrease the quantity of p1 by 5 */



		/*Call the method increasePrice to increase the price of p1 by 15% */



		/*Create the object p2 of the Product class using the second constructor:
		  name is Strawberry, price is 40.0, and quantity is 30 */



		/*Print the cheaper of p1 and p2 using toString */



	}
}
```