Lab 10

1.

Alarm API

Suppose we have the following interface for an alarm system.

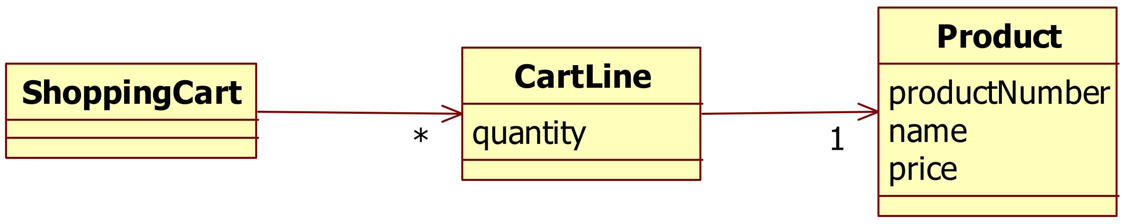
```
public interface Alarm {
	void turnSmokeOn();
	void turnSmokeOff();
	void turnMotionOn();
	void turnMotionOff();
	void turnTemperatureOn();
	void turnTemperatureOff();
	// if the motionsensor senses motion, wait 'delay' seconds
	// before the motion alarm goes off
	void setMotion(int delay);
	// if the temperature reaches this value the temperature alarm
	// goes off
	void setTemperature(int temperature);
	// start with logging temperature values every [interval]
	//minutes
	void startTempLogging();
	// stop logging temparature values
	void stopTempLogging();
	// change logging interval to [interval] minutes
	void setTempLogInterval(int interval);
	// get the history of temperatures for the last number of days
	List<TempLogRecord> getTemparatureHistory(int days);
	// start with logging alarm events
	void startLogging();
	// stop logging alarm events
	void stopLogging();
	// change logging interval to [interval] minutes
	void setLogInterval(int interval);
	// get the history of alarm events for the last number of days
	List<AlarmLogRecord> getAlarmHistory(int days);

}
```

Is this interface a good interface, and if not, how would you change this interface.

2.

Suppose we have a shopping component with the following domain classes:



We design the following API for this component:

```
public interface ShoppingCart {

  void addToCart(int cartId, CartLine cartLine, Product product);
  void updateCart(int cartId, CartLine cartLine);
  void removeFromCart(int cartId, CartLine cartLine);
  int getQuantityOnStock(Product product);
  Order createOrderFromCart(int cartId);
  Shoppingcart getShoppingcart(int cartId);
  saveCart(ShoppingCart shoppingCart);
}
```

Is this interface a good interface, and if not, how would you change this interface.

3.

In the project **SpeakerRegistration** project of assignment 10, we create a new Speaker as follows:

Speaker speaker1 = new Speaker("Frank", "Brown", "fbrown@acme.com", 3, true, "www.brownblog.com", browser, "Acme inc.", 800);

Refactor the code so that we know what these parameters mean, and the Speaker should also be immutable.