We decided to operate with Calendar instead of Date as Date is almost deprecated, also Calendar is quite useful and versatile to use. Also in this report we will comment on the changes made with respect the previous lab, as most of the code is reused and remains the same.

FullOnlineStore:
We implemented a list of AuctionItem in order to have easy access to all the auction items, as well as a sales list where we can save and arrange all the sales of the store.
Also now the store can only have one instance as all the attributes are static and the constructor private.
Now we also have a date, that is an instance of Calendar Class. to control the deadline of the AuctionItems and the date of sale. And when we add a new instance on the sales list or itemsStock list it is sorted by date on sales case and by price on itemsStock case.

Item:
Due to implementing the Taxable interface we had to implement the getPriceOnlyTax(), getPricePlusTax() and sumTotalTax(Taxable t). This returns the price of each item with its tax in a different way each, the name is self-explanatory.
Also with the implementation of Comparable interface we had to implement the compareTo() function, to compare two Item instances, using the price of the Item as the attribute to sort with.

AuctionItem:
Now the AuctionItem has a Deadline that is a Calendar instance, for this reason there are some minor changes in frozen() and constructor.

Package:
Due to implementing the Taxable interface we had to implement the getPriceOnlyTax(), getPricePlusTax() and sumTotalTax(Taxable t). This returns the price of each item with its tax in a different way each, the name is self-explanatory.

Buyer:
Now the Buyer when buying, with buy(Item i, double number) pays the price of the full product, with taxes included.

Admin:
The Admin now operates using the Calendar class, producing a minor change on manageAuction(AuctionItem i, Calendar d).

Taxable:
New interface that introduces the new tax, IVA, and defines the functions required by the Classes that implements it.

Sale:
New Class that implements Comparable and has the date of sale, the item sold and the buyer.With the implementation of Comparable interface we had to implement the compareTo() function, to compare two Sale instances, using the date of the sale as the attribute to sort with. The other functions are to get the date, buyer and item of the transaction.

We had some problems trying to figure out the price the user receives, but we figured that the Buyer should receive the price with all the taxes while the Seller receives the price without taxes, as those are not his profits. We also had some problems with updating the program to work with Calendar instances instead of the previous Strings. In general we could make it worked out well and we could make it work fine.