**Class Diagram Decoder**

A bold entry in the first column means modified, a strikethrough means removed.

|  |  |  |
| --- | --- | --- |
| **Package Name** | **Package Description** | **Packages Used** |
| GUI | Contains GUI boundaries and a control class. | Database, Stock, Sales, Customers, Reports, Staff, Rates, Utils |
| Database | Contains the database abstraction layer and the helper interface (Helpers are implemented in the packages they are used in). |  |
| **Stock** | Contains the blank management system and blank types. | Database |
| Sales | Contains the sales, refunds and transactions. | Database, Rates, Staff |
| Customers | Contains customer accounts and discount plans. | Database, Utils |
| **Reports** | Contains report producers. | Stock, Sales~~, Rates~~ |
| Staff | Contains staff accounts. | Database, Utils |
| **Rates** | Contains the exchange ~~and tax~~ rates. | Database |
| Utils | Contains utility classes. |  |

New and removed classes are bolded.

|  |  |
| --- | --- |
| **Package Name** | **Objects (Entity, Control, Boundary, Interface)** |
| GUI | [Awaiting GUI] {Managed by Alfred Manville} |
| Database | DB\_Connector (Control), IDB\_Connector (Interface), SQLHelper (Control) |
| **Stock** | Blank (Entity), BlankCollection (Entity), BlankController (Control), IStockAccessor (Interface), BlankSQLHelper (Control), **BlankType (Entity), BlankTypeCollection (Entity), BlankTypeSQLHelper (Control)** |
| Sales | Sale (Entity), SaleCollection (Entity), TransactionCollection (Entity), Transaction (Entity), Refund (Entity), RefundCollection (Entity), SaleController (Control), ISalesAccessor (Interface), SaleSQLHelper (Control), RefundSQLHelper (Control), TransactionSQLHelper (Control) |
| Customers | Customer (Entity), CustomerCollection (Entity), DiscountPlan (Entity), DiscountPlanCollection (Entity), CustomerController (Control), ICustomerAccessor (Interface), CustomerSQLHelper (Control), DiscountPlanSQLHelper (Control) |
| Reports | IReportGenerator (Interface), TicketStockReport (Control), InterlineSalesReport (Control), LocalSalesReport (Control), ReporterControl (Control), IReportAccessor (Interface) |
| Staff | Account (Entity), AccountCollection (Entity), AccountController (Control), IStaffAccessor (Interface), AccountSQLHelper (Control) |
| **Rates** | ExchangeRate (Entity), ExchangeRateCollection (Entity), **~~TaxRate(Entity), TaxRateCollection (Entity)~~**, RateController (Control), IRateAccessor (Interface), ExchangeRateSQLHelper (Control), **~~TaxRateSQLHelper (Control)~~** |
| Utils | PersonalInformation (Entity) |

Note, the 2 below tables are missing GUI objects. {} – Multiplicity, [] – Relation Type, <-> - Relation Direction Visually, x – Navigable False Side, () – Relation Name

|  |  |
| --- | --- |
| **Object** | **Associations** |
| DB\_Connector |  |
| IDB\_Connector | [Realization] <- DB\_Connector |
| **SQLHelper** | [Generalization] <- BlankSQLHelper; [Generalization] <- SaleSQLHelper; [Generalization] <- RefundSQLHelper; [Generalization] <- TransactionSQLHelper; [Generalization] <- CustomerSQLHelper; [Generalization] <- DiscountPlanSQLHelper; [Generalization] <- AccountSQLHelper; [Generalization] <- ExchangeRateSQLHelper; **~~[Generalization] <- TaxRateSQLHelper;~~ [Generalization] <- BlankTypeSQLHelper** |
| **Blank** | [Composition] {1} x-> {0..1} BlankSQLHelper (Uses); [Dependency] -> IDB\_Connector; **[Aggregation] {1} x-> {1} BlankType (isBlankType)** |
| BlankCollection | [Aggregation] {1} x-> {0..\*} Blank (hasBlanks) |
| **BlankController** | [Dependency] -> ISalesAccessor; [Dependency] -> IDB\_Connector; [Composition] {1} x-> {1} BlankCollection (blanks); **[Composition] {1} x-> {1} BlankTypeCollection (blankTypes)** |
| IStockAccessor | [Realization] <- BlankController |
| BlankSQLHelper | [Association] {1} x-> {1} Blank (Reads from) |
| **BlankType** | **[Composition] {1} x-> {0..1} BlankTypeSQLHelper (Uses); [Dependency] -> IDB\_Connector** |
| **BlankTypeCollection** | **[Aggregation] {1} x-> {0..\*} BlankType (hasBlankTypes)** |
| **BlankTypeSQLHelper** | **[Association] {1} x-> {1} BlankType (Reads from)** |
| Sale | [Composition] {1} x-> {0..1} SaleSQLHelper (Uses); [Composition] {1} x-> {0..1} TransactionCollection (transactions);[Dependency] -> IDB\_Connector |
| SaleCollection | [Aggregation] {1} x-> {0..\*} Sale (hasSales) |
| TransactionCollection | [Aggregation] {1} x-> {0..\*} Transaction (hasTransactions) |
| **Transaction** | [Composition] {1} x-> {0..1} TransactionSQLHelper (Uses); [Composition] {1} x-> {0..1} Refund (refund); [Dependency] -> IDB\_Connector; **[Dependency] -> IRateAccessor; [Aggregation] {1} x-> {1} RateController (Uses)** |
| Refund | [Composition] {1} x-> {0..1} RefundSQLHelper (Uses); [Dependency] -> IDB\_Connector |
| RefundCollection | [Aggregation] {1} x-> {0..\*} Refund (hasRefunds) |
| **SaleController** | **~~[Dependency] -> IRateAccessor;~~** [Dependency] -> IDB\_Connector; [Composition] {1} x-> {1} SaleCollection (sales); [Composition] {1} x-> {1} RefundCollection (refunds); **[Dependency] -> IStaffAccessor; [Aggregation] {1} x-> {1} AccountController (Uses)** |
| ISalesAccessor | [Realization] <- SaleController |
| SaleSQLHelper | [Association] {1} x-> {1} Sale (Reads from) |
| RefundSQLHelper | [Association] {1} x-> {1} Refund (Reads from) |
| TransactionSQLHelper | [Association] {1} x-> {1} Transaction (Reads from) |
| Customer | [Composition] {1} x-> {0..1} CustomerSQLHelper (Uses); [Aggregation] {1} x-> {0..1} DiscountPlan (hasPlan); [Dependency] -> IDB\_Connector; **[Aggregation] {1} x-> {0..1} (customerInformation)** |
| CustomerCollection | [Aggregation] {1} x-> {0..\*} Customer (hasCustomers) |
| DiscountPlan | [Composition] {1} x-> {0..1} DiscountPlanSQLHelper (Uses); [Dependency] -> IDB\_Connector |
| DiscountPlanCollection | [Aggregation] {1} x-> {0..\*} DiscountPlan (hasPlans) |
| CustomerController | [Dependency] -> IDB\_Connector; [Composition] {1} x-> {1} CustomerCollection (customers); [Composition] {1} x-> {1} DiscountPlanCollection (plans) |
| ICustomerAccessor | [Realization] <- CustomerController |
| CustomerSQLHelper | [Association] {1} x-> {1} Customer (Reads from) |
| DiscountPlanSQLHelper | [Association] {1} x-> {1} DiscountPlan (Reads from) |
| IReportGenerator | [Realization] <- TicketStockReport; [Realization] <- InterlineSalesReport; [Realization] <- LocalSalesReport |
| **TicketStockReport** | [Dependency] -> IStockAccessor; **[Dependency] -> ISalesAccessor** |
| **InterlineSalesReport** | [Dependency] -> ISalesAccessor; **~~[Dependency] -> IRateAccessor;~~** **[Dependency] -> IStockAccessor** |
| **LocalSalesReport** | [Dependency] -> ISalesAccessor; [**~~Dependency] -> IRateAccessor;~~ [Dependency] -> IStockAccessor** |
| **ReporterControl** | [Composition] {1} x-> {1} TicketStockReport (stockReporter); [Composition] {1} x-> {1} InterlineSalesReport (interlineSalesReporter); [Composition] {1} x-> {1} LocalSalesReport (localSalesReporter); **[Dependency] -> IStockAccessor; [Dependency] -> ISalesAccessor** |
| IReportAccessor | [Realization] <- ReporterControl |
| Account | [Composition] {1} x-> {0..1} AccountSQLHelper (Uses); [Dependency] -> IDB\_Connector; **[Aggregation] {1} x-> {0..1} (accountInformation)** |
| AccountCollection | [Aggregation] {1} x-> {0..\*} Account (hasAccounts) |
| AccountController | [Dependency] -> IDB\_Connector; [Composition] {1} x-> {1} AccountCollection (accounts) |
| IStaffAccessor | [Realization] <- AccountController |
| AccountSQLHelper | [Association] {1} x-> {1} Account (Reads from) |
| ExchangeRate | [Composition] {1} x-> {0..1} ExchangeRateSQLHelper (Uses); [Dependency] -> IDB\_Connector |
| ExchangeRateCollection | [Aggregation] {1} x-> {0..\*} ExchangeRate (hasExchangeRates) |
| **~~TaxRate~~** | **~~[Composition] {1} x-> {0..1} TaxRateSQLHelper (Uses); [Dependency] -> IDB\_Connector~~** |
| **~~TaxRateCollection~~** | **~~[Aggregation] {1} x-> {0..\*} TaxRate (hasTaxRates)~~** |
| **RateController** | [Dependency] -> IDB\_Connector; [Composition] {1} x-> {1} ExchangeRateCollection (exchanger); ~~[Composition] {1} x-> {1} TaxRateCollection (taxer)~~ |
| IRateAccessor | [Realization] <- RateController |
| ExchangeRateSQLHelper | [Association] {1} x-> {1} ExchangeRate (Reads from) |
| **~~TaxRateSQLHelper~~** | **~~[Association] {1} x-> {1} TaxRate (Reads from)~~** |
| **PersonalInformation** |  |

Note the below table has Classifier Operations underlined and highlighted blue, Abstract Operations ***italicized and bolded*** and attributes with getters and setters are defined with @ before their visibility declaration. *In progress, missing pieces are the reporting package [Await…] means in progress, please don’t use the new stuff defined in this row yet.*

|  |  |  |
| --- | --- | --- |
| **Object** | **Attributes** | **Operations** |
| **DB\_Connector** | -conn : Connection | +connect(databasePath : string, username : string, password : string) : Connection  +execute(sql : string) : int  +executeAndReturn(sql : string) : ResultSet  +backup() : byte[]  +restore(data : byte[]) : void  +close() : int  +DB\_Connector() |
| **IDB\_Connector** |  | ***+connect(databasePath : string, username : string, password : string) : Connection***  ***+execute(sql : string) : int***  ***+executeAndReturn(sql : string) : ResultSet***  ***+backup() : byte[]***  ***+restore(data : byte[]) : void***  ***+close() : int*** |
| **SQLHelper** |  | ***+store(o : object) : string***  ***+load(o: object , r : ResultSet) : boolean***  ***#assureGeneration() : boolean*** |
| **Blank** | @ -ID : long  @ -returned : boolean  @ -blacklisted : boolean  @ -voided : Boolean  -assignedID : long  @ -description : string | #Blank(connector : IDB\_Connector, id : long)  #Blank(connector : IDB\_Connector, returned : boolean, blacklisted : boolean, void : boolean, description : string)  +assign(staffID : long) : void  +getAssignment() : long  +getBlankType() : BlankType  +getDescription() : string  +setDescription(description : string) : void |
| BlankCollection | @ -index : int = 0 | +BlankCollection()  +addItem(item : Blank) : int  +removeItem(id : int) : void  +getNextItem() : Blank  +getSize() : int |
| **BlankController** | -connector : IDB\_Connector | +BlankController(connector : IDB\_Connector)  +createBlank(id : long, assignedID : long) : void  +returnBlank(id : long) : void  +blacklistBlank(id : long) : void  +voidBlank(id : long) : void  +reAssignBlank(id : long) : void  +getBlankType(id : long) : int  +getBlanks(assignedID : long) : long[]  +isBlankVoided(id : long) : boolean  +isBlankReturned(id : long) : boolean  +isBlankBlacklisted(id : long) : boolean  +createBlankType(typeID : int, description : string) : void  +setBlankTypeDescription(typeID : int, description : string) : void  +getBlankTypeDescription() : string  +deleteBlankType(typeID : int) : void  +listBlankTypes() : int[] |
| **IStockAccessor** |  | ***+createBlank(id : long, assignedID : long) : void***  ***+returnBlank(id : long) : void***  ***+blacklistBlank(id : long) : void***  ***+voidBlank(id : long) : void***  ***+reAssignBlank(id : long) : void***  ***+getBlankType(id : long) : int***  ***+getBlanks(assignedID : long) : long[]***  ***+isBlankVoided(id : long) : boolean***  ***+isBlankReturned(id : long) : boolean***  ***+isBlankBlacklisted(id : long) : boolean***  ***+createBlankType(typeID : int, description : string) : void***  ***+setBlankTypeDescription(typeID : int, description : string) : void***  ***+getBlankTypeDescription() : string***  ***+deleteBlankType(typeID : int) : void***  ***+listBlankTypes() : int[]*** |
| **BlankType** | @ -ID : int  @ -description : string | #BlankType(connector : IDB\_Connector, id : int)  #BlankType(connector : IDB\_Connector, id : int, description : string) |
| **BlankTypeCollection** | @ -index : int = 0 | +BlankTypeCollection()  +addItem(item : BlankType) : int  +removeItem(id : int) : void  +getNextItem() : BlankType  +getSize() : int |
| **BlankSQLHelper** |  | +store(o : object) : string  +load(o: object , r : ResultSet) : boolean  #assureGeneration() : boolean  #BlankSQLHelper() |
| **BlankTypeSQLHelper** |  | +store(o : object) : string  +load(o: object , r : ResultSet) : boolean  #assureGeneration() : boolean  #BlankTypeSQLHelper() |
| **Sale** | @ -ID : long  ~~@ -blank : long~~  @ -customer : long  @ -type : string  @ -commissionRate : double  @ -saleDate : Date  @ -dueDate : Date  @ -cost : double  @ -tax : double  @ -secondaryTax : double | +Sale(connector : IDB\_Connector, id : long)  ~~+Sale(connector : IDB\_Connector, blank : long, customer : long, type : string, commissionRate : double, dueDate : Date, saleDate : Date)~~  +Sale(connector : IDB\_Connector, id : long, customer : long, type : string, commissionRate : double, dueDate : Date, saleDate : Date, cost : double, tax : double, secondaryTax : double)  +isPaymentOverdue() : boolean  +getTotalCost() : double  #getTransactions() : TransactionCollection  +getTransaction(id : long) : Transaction |
| SaleCollection | @ -index : int = 0 | +SaleCollection()  +addItem(item : Sale) : int  +removeItem(id : int) : void  +getNextItem() : Sale  +getSize() : int |
| TransactionCollection | @ -index : int = 0 | +TransactionCollection()  +addItem(item : Transaction) : int  +removeItem(id : int) : void  +getNextItem() : Transaction  +getSize() : int |
| **Transaction** | @ -ID : long  @ -saleID : long  @ -date : Date  ~~@ -taxRate : double~~  @ -localCurrencyName : string  @ -localCurrencyCost : double  ~~-usdCost : double~~  @ -usdCost : double  @ -paymentType : string  @ -cardNumber : long  @ -chequeNumber : long | +Transaction(connector : IDB\_Connector, id : long)  ~~+Transaction(connector : IDB\_Connector, saleID : long, date : Date, taxRate : double, localCurrencyName : string, localCurrencyCost : string, paymentType : string, cardNumber : long)~~  +Transaction(connector : IDB\_Connector, saleID : long, date : Date, localCurrencyName : string, localCurrencyCost : string, paymentType : string, cardNumber : long, chequeNumber : long)  ~~+convertToUSD(conversionFactor : double) : void~~  ~~+getUSDCost() : double~~  +assignRefund(refundIn : Refund) : boolean  +getRefund() : Refund |
| Refund | @ -ID : long  @ -transactionID : long  @ -refundDate : Date  @ -localCurrencyRefund : double | +Refund(connector : IDB\_Connector, id : long)  +Refund(connector : IDB\_Connector, transactionID : long, refundDate : Date, localCurrencyAmount : double) |
| RefundCollection | @ -index : int = 0 | +RefundCollection()  +addItem(item : Refund) : int  +removeItem(id : int) : void  +getNextItem() : Refund  +getSize() : int |
| **SaleController** | -connector : IDB\_Connector  -staffer : IStaffAccessor | +SaleController(connector : IDB\_Connector, staffer : IStaffAccessor)  +sell(blank : long, customer : long, type : string, commissionRate : double, dueDate : Date, saleDate : Date, cost : double, tax : double, secondaryTax : double) : long  +transact(saleID : long, date : Date, amount : double, currency : string, paymentType : string, cardNumber : long, chequeNumber : long, usdRate : long) : long  +fullyPaid(saleID : long) : boolean  +late(saleID : long, date : Date) : boolean  +refund(saleID : long) : long[]  +getSales(staff : long) : long[]  +getSalesByCustomer(customer : long) : long[]  +getSale(saleID : long) : Sale  +getRefund(refundID : long) : Refund |
| **ISalesAccessor** |  | ***+sell(blank : long, customer : long, type : string, commissionRate : double, dueDate : Date, saleDate : Date, cost : double, tax : double, secondaryTax : double) : long***  ***+transact(saleID : long, date : Date, amount : double, currency : string, paymentType : string, cardNumber : long, chequeNumber : long, usdRate : long) : long***  ***+fullyPaid(saleID : long) : boolean***  ***+late(saleID : long, date : Date) : boolean***  ***+refund(saleID : long) : long[]***  ***+getSales(staff : long) : long[]***  ***+getSalesByCustomer(customer : long) : long[]***  ***+getSale(saleID : long) : Sale***  ***+getRefund(refundID : long) : Refund*** |
| **SaleSQLHelper** |  | +store(o : object) : string  +load(o: object , r : ResultSet) : boolean  #assureGeneration() : boolean  #SaleSQLHelper() |
| **RefundSQLHelper** |  | +store(o : object) : string  +load(o: object , r : ResultSet) : boolean  #assureGeneration() : boolean  #RefundSQLHelper() |
| **TransactionSQLHelper** |  | +store(o : object) : string  +load(o: object , r : ResultSet) : boolean  #assureGeneration() : boolean  #TransactionSQLHelper() |
| Customer |  | +Customer(connector : IDB\_Connector, id : long)  +Customer(connector : IDB\_Connector, planID : long, firstName : string, lastName : string, phoneNumber : string, emailAddress : string, dateOfBirth : Date, postcode : string, houseNumber : string, streetName : string) |
| CustomerCollection | @ -index : int = 0 | +CustomerCollection()  +addItem(item : Customer) : int  +removeItem(id : int) : void  +getNextItem() : Customer  +getSize() : int |
| DiscountPlan | @ -ID : long  @ -type : int  @ -percentage : double | +DiscountPlan(connector : IDB\_Connector, id : long)  +DiscountPlan(connector : IDB\_Connector, type : int, percentage : double)  +getDiscounted(amount : double) : double |
| DiscountPlanCollection | @ -index : int = 0 | +DiscountPlanCollection()  +addItem(item : DiscountPlan) : int  +removeItem(id : int) : void  +getNextItem() : DiscountPlan  +getSize() : int |
| **CustomerController** | -connector : IDB\_Connector | +CustomerController(connector : IDB\_Connector)  +createAccount(info : PersonalInformation, planID : long) : long  +getPersonalInformation(customer : long) : PersonalInformation  +setPersonalInformation(customer : long, info : PersonalInformation) : void  +getAccountPlan(customer : long) : long  +setAccountPlan(customer : long, plan : long) : void  +deleteAccount(customer : long) : void  +listAccounts() : long[]  +createPlan(type : int, percentage : double) : long  +getPlanType(plan : long) : int  +getPlanPercentage(plan : long) : double  +usePlan(plan : long, amount : double) : double  +removePlan(plan : long) : void  +listPlans() : long[] |
| **ICustomerAccessor** |  | ***+createAccount(info : PersonalInformation, planID : long) : long***  ***+getPersonalInformation(customer : long) : PersonalInformation***  ***+setPersonalInformation(customer : long, info : PersonalInformation) : void***  ***+getAccountPlan(customer : long) : long***  ***+setAccountPlan(customer : long, plan : long) : void***  ***+deleteAccount(customer : long) : void***  ***+listAccounts() : long[]***  ***+createPlan(type : int, percentage : double) : long***  ***+getPlanType(plan : long) : int***  ***+getPlanPercentage(plan : long) : double***  ***+usePlan(plan : long, amount : double) : double***  ***+removePlan(plan : long) : void***  ***+listPlans() : long[]*** |
| **CustomerSQLHelper** |  | +store(o : object) : string  +load(o: object , r : ResultSet) : boolean  #assureGeneration() : boolean  #CustomerSQLHelper() |
| **DiscountPlanSQLHelper** |  | +store(o : object) : string  +load(o: object , r : ResultSet) : boolean  #assureGeneration() : boolean #DiscountPlanSQLHelper() |
| **IReportGenerator** |  | ***+generate() : Image***  ***+generateFor(subject : long) : Image***  ***+generateTables() : string[][][]***  ***+generateTablesFor(subject : long) : string[][][]***  ***+setPeriod(month : int, year : int) : void*** |
| **TicketStockReport** | -stock : IStockAccessor  -sales : ISalesAccessor  -month : int  -year : int | #TicketStockReport(stock : IStockAccessor, sales : ISalesAccessor)  +receivedBlanks(staffID : long) : long[]  +assignedBlanks(staffID : long) : long[]  +usedBlanks(staffID : long) : long[]  +remainingStock(staffID : long) : long[]  +generate() : Image  +generateFor(subject : long) : Image  +generateTables() : string[][][]  +generateTablesFor(subject : long) : string[][][]  +setPeriod(month : int, year : int) : void |
| **InterlineSalesReport** | -stock : IStockAccessor  -sales : ISalesAccessor  @ -currency : string | #InterlineSalesReport(stock : IStockAccessor, sales : ISalesAccessor)  +soldBlanks(subject : long) : long[]  +soldBlanksUSD(subject : long) : long[]  +soldBlanksUSDSum(subject : long) : long  +soldBlanksConversionFactor(subject : long) : double[]  +soldBlanksAsCurrency(subject : long) : long[]  +soldBlanksAsCurrencySum(subject : long) : long  +soldBlanksTax(subject : long) : long[]  +soldBlanksTaxSum(subject : long) : long  +soldBlanksSecondaryTax(subject : long) : long[]  +soldBlanksSecondaryTaxSum(subject : long) : long  +soldBlanksTotalCosts(subject : long) : long[]  +soldBlanksTotalCostsSum(subject : long) : long  +soldBlanksCash(subject : long) : long[]  +soldBlanksCashSum(subject : long) : long  +soldBlanksCardNumber(subject : long) : long[]  +soldBlanksCardUSD(subject : long) : long[]  +soldBlanksCardUSDSum(subject : long) : long  +soldBlanksCardAsCurrency(subject : long) : long[]  +soldBlanksCardAsCurrencySum(subject : long) : long  +soldBlanksCommissions(subject : long, plan : long) : long[][]  +soldBlanksCommissionSums(subject : long, plan : long) : long[][]  +soldBlanksTotalTax(subject : long) : long[]  +soldBlanksTotalTaxSum(subject : long) : long  +totalPaid(subject : long) : long  +totalCommission(subject : long) : long  +totalDebit(subject : long) : long  +totalNetRemittence(subject : long) : long  +generate() : Image  +generateFor(subject : long) : Image  +generateTables() : string[][][]  +generateTablesFor(subject : long) : string[][][]  +setPeriod(month : int, year : int) : void |
| **LocalSalesReport** | -stock : IStockAccessor  -sales : ISalesAccessor  @ -currency : string | #LocalSalesReport(stock : IStockAccessor, sales : ISalesAccessor)  +soldBlanks(subject : long) : long[]  +soldBlanksAsCurrency(subject : long) : long[]  +soldBlanksAsCurrencySum(subject : long) : long  +soldBlanksUSD(subject : long) : long[]  +soldBlanksUSDSum(subject : long) : long  +soldBlanksCash(subject : long) : long[]  +soldBlanksCashSum(subject : long) : long  +soldBlanksCardNumber(subject : long) : long[]  +soldBlanksCardUSD(subject : long) : long[]  +soldBlanksCardUSDSum(subject : long) : long  +soldBlanksCardAsCurrency(subject : long) : long[]  +soldBlanksCardAsCurrencySum(subject : long) : long  +soldBlanksChequeNumber(subject : long) : long[]  +soldBlanksCheque(subject : long) : long[]  +soldBlanksChequeSum(subject : long) : long[]  +soldBlanksInvoice(subject : long) : long[]  +soldBlanksInvoiceSum(subject : long) : long  +soldBlanksTax(subject : long) : long[]  +soldBlanksTaxSum(subject : long) : long  +soldBlanksCommissions(subject : long, plan : long) : long[][]  +soldBlanksCommissionSums(subject : long, plan : long) : long[][]  +totalPaid(subject : long) : long  +totalCommission(subject : long) : long  +totalDebit(subject : long) : long  +totalNetRemittence(subject : long) : long  +generate() : Image  +generateFor(subject : long) : Image  +generateTables() : string[][][]  +generateTablesFor(subject : long) : string[][][]  +setPeriod(month : int, year : int) : void |
| **ReporterControl** |  | +ReporterControl(stock : IStockAccessor, sales : ISalesAccessor)  +getTicketStock() : TicketStockReport  +getInterlineSales() : InterlineSalesReport  +getLocalSales() : LocalSalesReport |
| **IReportAccessor** |  | ***+getTicketStock() : TicketStockReport***  ***+getInterlineSales() : InterlineSalesReport***  ***+getLocalSales() : LocalSalesReport*** |
| **Account** | @ -role : string  @ -commission: double  -hashedPassword : string  -passwordSalt : string | +Account(connector : IDB\_Connector, id : long)  ~~+Account(connector : IDB\_Connector, id : long, firstName : string, lastName : string, phoneNumber : string, emailAddress : string, dateOfBirth : Date, postcode : string, houseNumber : string, streetName : string, role : string, commission : double, password : string)~~  +Account(connector : IDB\_Connector, id : long, firstName : string, lastName : string, phoneNumber : string, emailAddress : string, dateOfBirth : Date, postcode : string, houseNumber : string, streetName : string, role : string, commission : double, currency : string, password : string)  +attemptAuthentication(password : string) : Boolean  +setPassword(password : string)  +getHashedPassword() : string  +getPasswordSalt() : string  +setCurrency(currency : string) : void  +getCurrency() : string |
| AccountCollection | @ -index : int = 0 | +AccountCollection()  +addItem(item : Account) : int  +removeItem(id : int) : void  +getNextItem() : Account  +getSize() : int |
| **AccountController** | -authenticatedAccount : string = ""  -authenticatedRole : string = ""  -authenticatedRate : double = 0  -connector : IDB\_Connector | +AccountController(connector : IDB\_Connector)  ~~+createAccount(info : PersonalInformation, role : string, commission : double, password : string) : boolean~~  +createAccount(info : PersonalInformation, role : string, commission : double, currency : string, password : string) : void  +authenticateAccount(emailAddress : string, password : string) : boolean  +changePassword(emailAddress : string, password : string) : void  +getPersonalInformation(emailAddress : string) : PersonalInformation  +setPersonalInformation(emailAddress : string, info : PersonalInformation) : void  +listAccounts(role : string) : string[]  +deleteAccount(emailAddress : string) : void  +getCommissionRate(emailAddress : string) : double  +setCommissionRate(emailAddress : string, rate : double) : void |
| IStaffAccessor |  | ***~~+createAccount(info : PersonalInformation, role : string, commission : double, password : string) : boolean~~***  ***+createAccount(info : PersonalInformation, role : string, commission : double, currency : string, password : string) : void***  ***+authenticateAccount(emailAddress : string, password : string) : boolean***  ***+changePassword(emailAddress : string, password : string) : void***  ***+getPersonalInformation(emailAddress : string) : PersonalInformation***  ***+setPersonalInformation(emailAddress : string, info : PersonalInformation) : void***  ***+listAccounts(role : string) : string[]***  ***+deleteAccount(emailAddress : string) : void***  ***+getCommissionRate(emailAddress : string) : double***  ***+setCommissionRate(emailAddress : string, rate : double) : void*** |
| **AccountSQLHelper** |  | +store(o : object) : string  +load(o: object , r : ResultSet) : boolean  #assureGeneration() : boolean #AccountSQLHelper() |
| ExchangeRate | @ -currencyName : string  @ -usdConversionRate : double | +ExchangeRate(connector : IDB\_Connector, name : string)  +ExchangeRate(connector : IDB\_Connector, name : string, usdConversionRate : double)  +convertToUSD(localAmount : double) : double  +convertFromUSD(usdAmount : double) : double |
| ExchangeRateCollection | @ -index : int = 0 | +ExchangeRateCollection()  +addItem(item : ExchangeRate) : int  +removeItem(id : int) : void  +getNextItem() : ExchangeRate  +getSize() : int |
| **~~TaxRate~~** | ~~@ -countryCode : string~~  ~~@ -taxRate : double~~ | ~~+TaxRate(connector : IDB\_Connector, countryCode : string)~~  ~~+TaxRate(connector : IDB\_Connector, countryCode : string, taxRate : double)~~  ~~+taxAmount(toTax : double) : double~~  ~~+unTaxAmount(toUnTax : double) : double~~ |
| **~~TaxRateCollection~~** | ~~@ -index : int = 0~~ | ~~+TaxRateCollection()~~  ~~+addItem(item : TaxRate) : int~~  ~~+removeItem(id : int) : void~~  ~~+getNextItem() : TaxRate~~  ~~+getSize() : int~~ |
| **RateController** | -connector : IDB\_Connector | +RateController(connector IDB\_Connector)  ~~+getTaxRate(countryCode : string) : double~~  ~~+setTaxRate(countryCode : string, rate : double) : void~~  ~~+removeTaxRate(countryCode : string) : boolean~~  ~~+getTaxableCountries() : string[]~~  +getConversionRate(currency : string) : double  +setConversionRate(currency : string, rate : double) : void  +removeConversionRate(currency : string) : void  +getConvertableCurrencies() : string[] |
| **IRateAccessor** |  | ***+setDBConnector(conn : IDB\_Connector) : void***  ***~~+getTaxRate(countryCode : string) : double~~***  ***~~+setTaxRate(countryCode : string, rate : double) : void~~***  ***~~+removeTaxRate(countryCode : string) : boolean~~***  ***~~+getTaxableCountries() : string[]~~***  ***+getConversionRate(currency : string) : double***  ***+setConversionRate(currency : string, rate : double) : void***  ***+removeConversionRate(currency : string) : void***  ***+getConvertableCurrencies() : string[]*** |
| **ExchangeRateSQLHelper** |  | +store(o : object) : string  +load(o: object , r : ResultSet) : boolean  #assureGeneration() : boolean #ExchangeRateSQLHelper() |
| **~~TaxRateSQLHelper~~** |  | ~~+store(o : object) : string~~  ~~+load(o: object , r : ResultSet) : boolean +TaxRateSQLHelper()~~ |
| PersonalInformation | @ -firstName : string  @ -lastName : string  @ -phoneNumber : string  @ -emailAddress : string  @ -dateOfBirth : Date  @ -postcode : string  @ -houseNumber : string  @ -streetName : string | +PersonalInformation(firstName : string, lastName : string, phoneNumber : string, emailAddress : string, dateOfBirth : Date, postcode : string, houseNumber : string, streetName : string)  +getSQLString() : string |