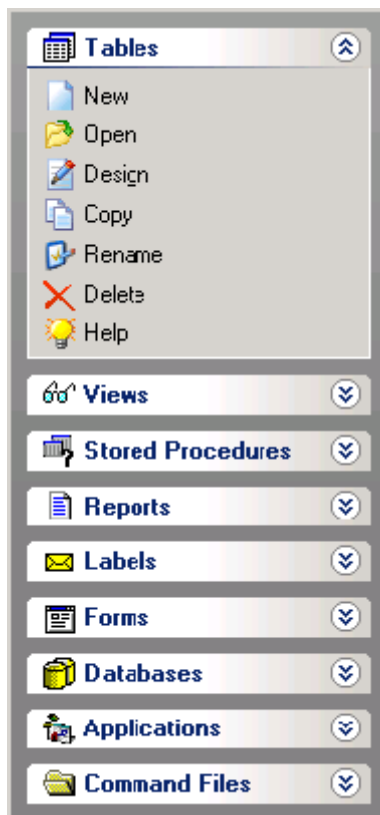
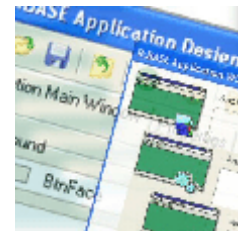
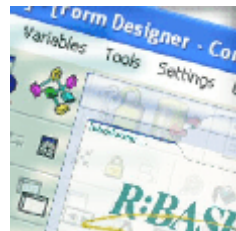
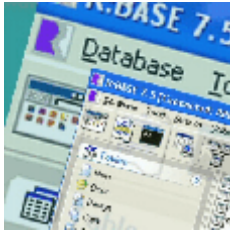


From The Edge:

Using Enhanced **UNLOAD STRUCTURE** Command in R:BASE

(R:BASE 7.6 and Turbo V-8 for Windows)



This is the database you have always wanted!

A. Razzak Memon

March 16, 2007

From the Edge: Using Enhanced **UNLOAD STRUCTURE** Command in R:BASE
Section: Enhanced Commands
Chapter: Running R:BASE Your Way!
Platform: R:BASE 7.6 and Turbo V-8 for Windows
Builds: R:BASE 7.6 (7.6.1.30316 or higher ...)
R:BASE C/S:1 7.6 for Windows (Build: 7.6.1.30316 or higher)
R:BASE V-8 Turbo (8.0.15.30316 or higher ...)

Traditionally, when you use the UNLOAD STRUCTURE or UNLOAD ALL commands, R:BASE engine generates a bunch of R:BASE specific code, including CHARacter Settings, DISCONNECT statement, STATICDB, ROWLOCK, FASTLOCK settings, CREATE SCHEMA, CREATE TABLE and so on ... Then, often we have to edit the text file, clean up the extra stuff and only leave the SQL commands that are necessary to build the table or view.

Practically, if you are using the UNLOAD STRUCTURE FOR ... command, it's because you want to transfer the TABLE or VIEW to a different database that already exists and that usually has a different name. All you really want is to transfer the table or view structure.

Did you know that using the enhanced UNLOAD command in the latest versions of R:BASE (see supported versions above), now you can UNLOAD the STRUCTURE that would generate only the SQL commands necessary to rebuild the table or view in question (with all rights, comments, rules, etc), but none of the extra stuff (CREATE SCHEMA or R:BASE settings)? That way, you could simply RUN the resulting file without having to edit it.

Traditional Command: UNLOAD STRUCTURE FOR tablename

Includes:

- Character Settings
- DISCONNECT Statement
- Define Settings for STATICDB, ROWLOCKS and FASTLOCK
- CREATE Database ... statement
- CREATE TABLE ... statement
- ALTER TABLE statements to ADD PRIMARY KEY, etc ...
- CREATE INDEX statements to create INDEXes
- COMMENT ON statements for each column
- COMMENT ON statement for table

UNLOAD ALL FOR tablename

Includes:

- Character Settings
- DISCONNECT Statement
- Define Settings for STATICDB, ROWLOCKS and FASTLOCK
- CREATE Database ... statement
- CREATE TABLE ... statement
- Define Settings for CASE, AUTOSKIP, REVERSE, BELL, NULL, DATE YEAR, DATE CENTURY, DATE SEQUENCE, TIME SEQUENCE, TOLERANCE, and ZERO
- LOAD statement to load data
- Define Settings for DATE FORMAT, TIME FORMAT, DATE SEQUENCE, TIME SEQUENCE
- ALTER TABLE statements to ADD PRIMARY KEY, etc ...
- CREATE INDEX statements to create INDEXes
- COMMENT ON statements for each column
- COMMENT ON statement for table

Enhanced Command:

UNLOAD STRUCTURE FOR TABLE tablename

To unload Table Definition only

UNLOAD STRUCTURE FOR tablename NOSCHEMA

To unload Table Definition with Constraints, Indexes and Comments

Examples:

Example 01: [To unload table structure \(Traditional\)](#)
See Appendix A for sample output

```
CONNECT RRBYW10
OUTPUT Customer.STR
UNLOAD STRUCTURE FOR Customer
OUTPUT SCREEN
```

Example 02: [To unload table definition for one table](#)
See Appendix B for sample output

```
CONNECT RRBYW10
OUTPUT Customer.STR
UNLOAD STRUCTURE FOR TABLE Customer
OUTPUT SCREEN
```

Example 03: [To unload table definition with Constraints, Indexes and Comments](#)
See Appendix C for sample output

```
CONNECT RRBYW10
OUTPUT Customer.STR
UNLOAD STRUCTURE FOR Customer NOSCHEMA
OUTPUT SCREEN
```

Example 04: [To unload table definition, Constraints, Indexes, Comments and data](#)
See Appendix D for sample output

```
CONNECT RRBYW10
OUTPUT Customer.ALL
UNLOAD ALL FOR Customer
OUTPUT SCREEN
```

Example 05: [To unload table definitions for all tables](#)
See Appendix E for sample output

```
CONNECT RRBYW10
OUTPUT AllTables.STR
UNLOAD STRUCTURE FOR TABLEDEF
OUTPUT SCREEN
```

Example 06: [To unload table definitions, Constraints, Indexes and Comments for all tables](#)

See Appendix F for sample output

```
CONNECT RRBYW10
OUTPUT AllTables.STR
UNLOAD STRUCTURE FOR TABLES
OUTPUT SCREEN
```

Example 07: [To unload multiple table definitions, Constraints, Indexes and Comments using wild card](#)

See Appendix G for sample output

```
CONNECT RRBYW10
OUTPUT MulipleTables.STR
UNLOAD STRUCTURE FOR TABLES CO%
OUTPUT SCREEN
```

This will unload structure for all tables starting with "Co", such as:
CompUsed, Contact and ContactCallNotes

Example 08: [To unload definition of a given View](#)

See Appendix H for sample output

```
CONNECT RRBYW10
OUTPUT CustomerView.VIE
UNLOAD STRUCTURE FOR VIEW CustomerView
OUTPUT SCREEN
```

```
CONNECT RRBYW10
OUTPUT CustomerView.VIE
UNLOAD STRUCTURE FOR CustomerView NOSHEMA
OUTPUT SCREEN
```

Example 09: [To unload definition of a given View \(Traditional\)](#)

See Appendix I for sample output

```
CONNECT RRBYW10
OUTPUT CustomerView.VIE
UNLOAD STRUCTURE FOR CustomerView
OUTPUT SCREEN
```

Example 10: [To unload definition of all views](#)

See Appendix J for sample output

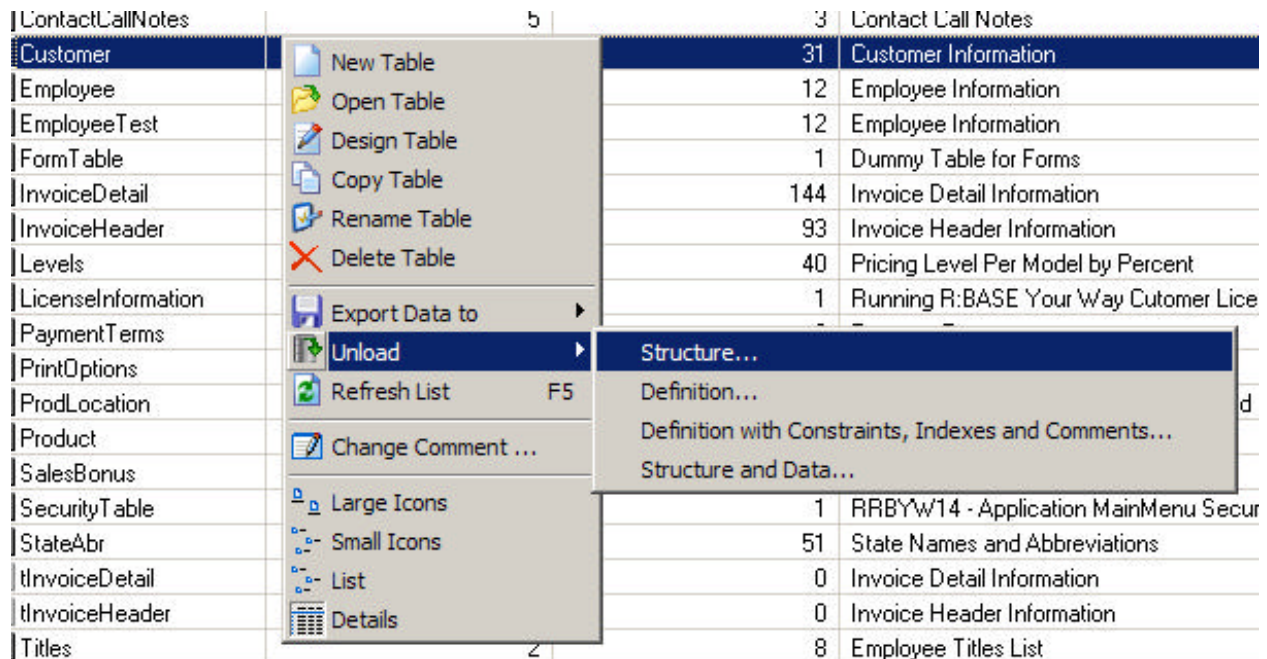
```
CONNECT RRBYW10
OUTPUT AllViews.VIE
UNLOAD STRUCTURE FOR VIEWS
OUTPUT SCREEN
```

Example 11: To unload definition of multiple views using wild card
 See Appendix K for sample output

```
CONNECT RRBYW10
OUTPUT MulipleViews.VIE
UNLOAD STRUCTURE FOR VIEWS New%
OUTPUT SCREEN
```

This will unload structure for all views starting with "New", such as:
 NewContact, NewCustomer, NewEmployee, NewInvoiceDetails, and NewInvoiceHeader

You may also use the Database Explorer to UNLOAD STRUCTURE using enhanced commands:



Appendix A.
(Traditional Output)

```
COMMAND:          UNLOAD STRUCTURE FOR Customer

SET QUOTES=NULL
SET QUOTES='
SET DELIMIT=NULL
SET DELIMIT=', '
SET LINEEND=NULL
SET LINEEND='p'
SET SEMI=NULL
SET SEMI=';'
SET PLUS=NULL
SET PLUS='+ '
SET SINGLE=NULL
SET SINGLE='_ '
SET MANY=NULL
SET MANY='% '
SET IDQUOTES=NULL
SET IDQUOTES='`'
SET CURRENCY '$' PREF 2 B
DISCONNECT
SET STATICDB OFF
SET ROWLOCKS ON
SET FASTLOCK OFF
CREATE SCHEMA  AUTHOR RRBYW14  NONE
CREATE TABLE `Customer` +
(`CustID` INTEGER NOT NULL +
('Customer ID Number Cannot be NULL. '), +
`Company` TEXT (40) , +
`CustAddress` TEXT (30) , +
`CustCity` TEXT (20) , +
`CustState` TEXT (2) , +
`CustZip` TEXT (10) , +
`CustPhone` TEXT (12) , +
`CustFax` TEXT (12) , +
`ModLevel` INTEGER , +
`LastUpdateDate` DATE , +
`LastUpdateTime` TIME , +
`LastOrderDate` DATE , +
`CustURL` TEXT (60) , +
`CustEmail` TEXT (60) , +
`CustStatus` TEXT (1) , +
`PaymentTerm` TEXT (20) , +
`CreditLimit` CURRENCY )
ALTER TABLE `Customer` ADD PRIMARY KEY +
(`CustID` ) +
('The customer id number must be unique for this table.',+
 'This customer id number is being referenced in another table, sorry.6',+
 'This value is being reference in another table.')
CREATE INDEX CustState ON `Customer` +
(`CustState` ASC SIZE 2 )
```

```

AUTONUM `CustID` IN `Customer` USING +
0. 0. NONUM
COMMENT ON `CustID` IN `Customer` IS +
'Customer Identification Number'
COMMENT ON `Company` IN `Customer` IS +
'Customer Company Name'
COMMENT ON `CustAddress` IN `Customer` IS +
'Customer Address'
COMMENT ON `CustCity` IN `Customer` IS +
'Customer City'
COMMENT ON `CustState` IN `Customer` IS +
'Customer State'
COMMENT ON `CustZip` IN `Customer` IS +
'Customer Zip Code'
COMMENT ON `CustPhone` IN `Customer` IS +
'Customer Phone Number'
COMMENT ON `CustFax` IN `Customer` IS +
'Customer Fax Number'
COMMENT ON `ModLevel` IN `Customer` IS +
'Customer Mod Level'
COMMENT ON `LastUpdateDate` IN `Customer` IS +
'Last Update Date'
COMMENT ON `LastUpdateTime` IN `Customer` IS +
'Last Update Time'
COMMENT ON `LastOrderDate` IN `Customer` IS +
'Last Order Data'
COMMENT ON `CustURL` IN `Customer` IS +
'Customer WWW URL Address'
COMMENT ON `CustEMail` IN `Customer` IS +
'Customer E-Mail Address'
COMMENT ON `CustStatus` IN `Customer` IS +
'Customer Status (A/I)'
COMMENT ON `PaymentTerm` IN `Customer` IS +
'Payment Terms'
COMMENT ON `CreditLimit` IN `Customer` IS +
'Credit Limit'
COMMENT ON TABLE `Customer` IS +
'Customer Information'

```

Appendix B.

(To Unload Table Definition Only)

COMMAND: UNLOAD STRUCTURE FOR TABLE Customer

```
CREATE TABLE `Customer` +
(`CustID` INTEGER NOT NULL +
('Customer ID Number Cannot be NULL.') , +
`Company` TEXT (40) , +
`CustAddress` TEXT (30) , +
`CustCity` TEXT (20) , +
`CustState` TEXT (2) , +
`CustZip` TEXT (10) , +
`CustPhone` TEXT (12) , +
`CustFax` TEXT (12) , +
`ModLevel` INTEGER , +
`LastUpdateDate` DATE , +
`LastUpdateTime` TIME , +
`LastOrderDate` DATE , +
`CustURL` TEXT (60) , +
`CustEmail` TEXT (60) , +
`CustStatus` TEXT (1) , +
`PaymentTerm` TEXT (20) , +
`CreditLimit` CURRENCY )
```

Appendix C.

(To Unload Table Definition with Constraints, Indexes and Comments)

COMMAND: UNLOAD STRUCTURE FOR Customer NOSHEMA

```
CREATE TABLE `Customer` +
(`CustID` INTEGER NOT NULL +
('Customer ID Number Cannot be NULL.')
```

, +
`Company` TEXT (40), +
`CustAddress` TEXT (30), +
`CustCity` TEXT (20), +
`CustState` TEXT (2), +
`CustZip` TEXT (10), +
`CustPhone` TEXT (12), +
`CustFax` TEXT (12), +
`ModLevel` INTEGER, +
`LastUpdateDate` DATE, +
`LastUpdateTime` TIME, +
`LastOrderDate` DATE, +
`CustURL` TEXT (60), +
`CustEmail` TEXT (60), +
`CustStatus` TEXT (1), +
`PaymentTerm` TEXT (20), +
`CreditLimit` CURRENCY)
ALTER TABLE `Customer` ADD PRIMARY KEY +
(`CustID`) +
('The customer id number must be unique for this table.',+
'This customer id number is being referenced in another table, sorry.6',+
'This value is being reference in another table.')
CREATE INDEX CustState ON `Customer` +
(`CustState` ASC SIZE 2)
AUTONUM `CustID` IN `Customer` USING +
0. 0. NONUM
COMMENT ON `CustID` IN `Customer` IS +
'Customer Identification Number'
COMMENT ON `Company` IN `Customer` IS +
'Customer Company Name'
COMMENT ON `CustAddress` IN `Customer` IS +
'Customer Address'
COMMENT ON `CustCity` IN `Customer` IS +
'Customer City'
COMMENT ON `CustState` IN `Customer` IS +
'Customer State'
COMMENT ON `CustZip` IN `Customer` IS +
'Customer Zip Code'
COMMENT ON `CustPhone` IN `Customer` IS +
'Customer Phone Number'
COMMENT ON `CustFax` IN `Customer` IS +
'Customer Fax Number'
COMMENT ON `ModLevel` IN `Customer` IS +
'Customer Mod Level'
COMMENT ON `LastUpdateDate` IN `Customer` IS +
'Last Update Date'

```
COMMENT ON `LastUpdateTime` IN `Customer` IS +  
'Last Update Time'  
COMMENT ON `LastOrderDate` IN `Customer` IS +  
'Last Order Data'  
COMMENT ON `CustURL` IN `Customer` IS +  
'Customer WWW URL Address'  
COMMENT ON `CustEMail` IN `Customer` IS +  
'Customer E-Mail Address'  
COMMENT ON `CustStatus` IN `Customer` IS +  
'Customer Status (A/I)'  
COMMENT ON `PaymentTerm` IN `Customer` IS +  
'Payment Terms'  
COMMENT ON `CreditLimit` IN `Customer` IS +  
'Credit Limit'  
COMMENT ON TABLE `Customer` IS +  
'Customer Information'
```

Appendix D.

(To Unload Table Definition with Constraints, Indexes and Comments and Data)

COMMAND: UNLOAD ALL FOR Customer

```
SET QUOTES=NULL
SET QUOTES='
SET DELIMIT=NULL
SET DELIMIT=', '
SET LINEEND=NULL
SET LINEEND='p'
SET SEMI=NULL
SET SEMI=';'
SET PLUS=NULL
SET PLUS='+ '
SET SINGLE=NULL
SET SINGLE='_ '
SET MANY=NULL
SET MANY='% '
SET IDQUOTES=NULL
SET IDQUOTES='`'
SET CURRENCY '$' PREF 2 B
DISCONNECT
SET STATICDB OFF
SET ROWLOCKS ON
SET FASTLOCK OFF
CREATE SCHEMA AUTHOR RRBYW14 NONE
CREATE TABLE `Customer` +
(`CustID` INTEGER NOT NULL +
('Customer ID Number Cannot be NULL.')
```

, +
`Company` TEXT (40), +
`CustAddress` TEXT (30), +
`CustCity` TEXT (20), +
`CustState` TEXT (2), +
`CustZip` TEXT (10), +
`CustPhone` TEXT (12), +
`CustFax` TEXT (12), +
`ModLevel` INTEGER, +
`LastUpdateDate` DATE, +
`LastUpdateTime` TIME, +
`LastOrderDate` DATE, +
`CustURL` TEXT (60), +
`CustEmail` TEXT (60), +
`CustStatus` TEXT (1), +
`PaymentTerm` TEXT (20), +
`CreditLimit` CURRENCY)
SET CASE OFF
SET AUTOSKIP OFF
SET REVERSE ON
SET BELL ON
SET NULL ' '
SET DATE YEAR 30
SET DATE CENTURY 19

```

SET DATE SEQUENCE MMDDYYYY
SET TIME SEQUENCE HHMMSS
SET TOLERANCE 0.
SET ZERO ON
LOAD `Customer`
NONUM
100,'Computer Warehouse - II','123 My Way, Suite 100','Boston','CA','02178','+
617-341-2189','617-341-9999',1,09/10/2003,15:15:00,01/01/2005,'http://www.cw+
I.com','sales@cw.com','N','Net 20 Days',10000.00
101,'Microtech University - I','24700 Industrial Parkway','Boston','MA','0216+
2','617-423-8921','617-423-9999',2,09/09/2003,7:06:00,09/27/2003,'http://www+
.mui.com','sales@mui.com','N','Net 10 Days',20000.00
:
:
:
130,'Murrysville Technology Center','3930 Old William Penn Highway','Murrysvi+
lle','PA','15668-1854','724-733-1234','724-733-9999',1,05/22/2004,7:10:00,+
05/22/2004,'http://www.mtc.com','ap@mtc.com','A','Net 45 Days',100000.00
END
SET DATE FORMAT 'MM/DD/YYYY'
SET TIME FORMAT 'HH:MM:SS'
SET DATE SEQUENCE MMDDYY
SET TIME SEQUENCE HHMMSS
ALTER TABLE `Customer` ADD PRIMARY KEY +
(`CustID` ) +
('The customer id number must be unique for this table.',+
'This customer id number is being referenced in another table, sorry.6',+
'This value is being reference in another table.')
CREATE INDEX CustState ON `Customer` +
(`CustState` ASC SIZE 2 )
AUTONUM `CustID` IN `Customer` USING +
0. 0. NONUM
COMMENT ON `CustID` IN `Customer` IS +
'Customer Identification Number'
COMMENT ON `Company` IN `Customer` IS +
'Customer Company Name'
COMMENT ON `CustAddress` IN `Customer` IS +
'Customer Address'
COMMENT ON `CustCity` IN `Customer` IS +
'Customer City'
COMMENT ON `CustState` IN `Customer` IS +
'Customer State'
COMMENT ON `CustZip` IN `Customer` IS +
'Customer Zip Code'
COMMENT ON `CustPhone` IN `Customer` IS +
'Customer Phone Number'
COMMENT ON `CustFax` IN `Customer` IS +
'Customer Fax Number'
COMMENT ON `ModLevel` IN `Customer` IS +
'Customer Mod Level'
COMMENT ON `LastUpdateDate` IN `Customer` IS +
'Last Update Date'
COMMENT ON `LastUpdateTime` IN `Customer` IS +

```

```
'Last Update Time'  
COMMENT ON `LastOrderDate` IN `Customer` IS +  
'Last Order Data'  
COMMENT ON `CustURL` IN `Customer` IS +  
'Customer WWW URL Address'  
COMMENT ON `CustEMail` IN `Customer` IS +  
'Customer E-Mail Address'  
COMMENT ON `CustStatus` IN `Customer` IS +  
'Customer Status (A/I)'  
COMMENT ON `PaymentTerm` IN `Customer` IS +  
'Payment Terms'  
COMMENT ON `CreditLimit` IN `Customer` IS +  
'Credit Limit'  
COMMENT ON TABLE `Customer` IS +  
'Customer Information'
```

Appendix E.

(To Unload Table Definitions for All Tables)

```
COMMAND:          UNLOAD STRUCTURE FOR TABLES TABLEDEF

CREATE TABLE `Customer` +
(`CustID` INTEGER NOT NULL +
('Customer ID Number Cannot be NULL.')
```

, +
`Company` TEXT (40) , +
`CustAddress` TEXT (30) , +
`CustCity` TEXT (20) , +
`CustState` TEXT (2) , +
`CustZip` TEXT (10) , +
`CustPhone` TEXT (12) , +
`CustFax` TEXT (12) , +
`ModLevel` INTEGER , +
`LastUpdateDate` DATE , +
`LastUpdateTime` TIME , +
`LastOrderDate` DATE , +
`CustURL` TEXT (60) , +
`CustEmail` TEXT (60) , +
`CustStatus` TEXT (1) , +
`PaymentTerm` TEXT (20) , +
`CreditLimit` CURRENCY)

CREATE TABLE `Component` +
(`CompID` TEXT (6) NOT NULL +
('The component id number is a primary key, it's value cannot be NULL.'), +
`CompDesc` TEXT (31))

CREATE TABLE `SalesBonus` +
(`EmpID` INTEGER , +
`TransDate` DATE , +
`NetAmount` CURRENCY , +
`BonusPct` REAL , +
`Bonus`= +
(netamount* bonuspct) CURRENCY)

CREATE TABLE `PaymentTerms` +
(`PaymentTerm` TEXT (20))

CREATE TABLE `CompUsed` +
(`Model` TEXT (6) , +
`CompID` TEXT (6))

CREATE TABLE `ProdLocation` +
(`Model` TEXT (6) , +
`Location` TEXT (4) , +
`OnHand` INTEGER NOT NULL +
('Value for Onhand cannot be null.'), +
`Cost` CURRENCY , +
`ReOrderPt` INTEGER)

CREATE TABLE `Levels` +
(`ModLevel` INTEGER NOT NULL +
('Value for Modlevel cannot be null.'), +
`Model` TEXT (6) NOT NULL +
('Value for Model cannot be null.'), +
`LevPerc` REAL)

```

CREATE TABLE `FormTable` +
(`ColumnZZ` INTEGER )
CREATE TABLE `BonusRate` +
(`MinAmount` CURRENCY , +
`MaxAmount` CURRENCY , +
`BonusPct` REAL )
CREATE TABLE `Product` +
(`Model` TEXT (6) NOT NULL +
('This is a Primary Key, it''s value cannot be NULL.') , +
`ProdName` TEXT (35) , +
`ProdDesc` TEXT (31) , +
`ListPrice` CURRENCY )
CREATE TABLE `InvoiceHeader` +
(`TransID` INTEGER NOT NULL +
('Value for column TransID cannot be null') , +
`CustID` INTEGER , +
`EmpID` INTEGER , +
`TransDate` DATE , +
`BillToCompany` TEXT (40) , +
`BillToAddress` TEXT (30) , +
`BillToCity` TEXT (20) , +
`BillToState` TEXT (2) , +
`BillToZip` TEXT (10) , +
`ShipToCompany` TEXT (40) , +
`ShipToAddress` TEXT (30) , +
`ShipToCity` TEXT (20) , +
`ShipToState` TEXT (2) , +
`ShipToZip` TEXT (10) , +
`NetAmount` CURRENCY , +
`Freight`= +
( netamount* .01) CURRENCY , +
`Tax`= +
( netamount* .081) CURRENCY , +
`InvoiceTotal`= +
( netamount+ freight+ tax) CURRENCY )
CREATE TABLE `PrintOptions` +
(`Option` TEXT (8) , +
`DisplaySequence` INTEGER , +
`DisplayName` TEXT (24) , +
`DisplayIcon` LONG VARBIT , +
`PrintOption` TEXT (8) , +
`ExtendedSyntax` TEXT (200) )
CREATE TABLE `InvoiceDetail` +
(`TransID` INTEGER , +
`DetailNum` INTEGER , +
`Model` TEXT (6) , +
`Units` INTEGER , +
`Price` CURRENCY , +
`Discount` REAL , +
`SalePrice`= +
(Price- (Price*Discount/100)) CURRENCY , +
`ExtPrice`= +
(Units* SalePrice) CURRENCY )

```

```

CREATE TABLE `ContactCallNotes` +
(`ContID` INTEGER , +
`EmpID` INTEGER , +
`CallDate` DATE , +
`CallTime` TIME NOT NULL +
('Value for column ContTime cannot be null') , +
`CallNotes` NOTE )
CREATE TABLE `LicenseInformation` +
(`LicenseNumber` TEXT (16) , +
`ConcurrentUsers` INTEGER , +
`CompanyName` TEXT (60) , +
`CompAddressLine1` TEXT (60) , +
`CompAddressLine2` TEXT (60) , +
`CompCity` TEXT (20) , +
`CompStateProvince` TEXT (20) , +
`CompPostalZipCode` TEXT (20) , +
`CompCountry` TEXT (20) , +
`ContactLastName` TEXT (20) , +
`ContactFirstName` TEXT (20) , +
`ContactMiddleName` TEXT (20) , +
`ContactPhone` TEXT (20) , +
`ContactFax` TEXT (20) , +
`ContactEMail` TEXT (60) , +
`ContactWEBAddress` TEXT (60) , +
`DateSold` DATE , +
`DateInstalled` DATE , +
`SupportDaysLeft` INTEGER )
CREATE TABLE `Contact` +
(`CustID` INTEGER , +
`ContID` INTEGER NOT NULL +
('Value for column ContID cannot be null') , +
`ContFName` TEXT (10) , +
`ContLName` TEXT (16) , +
`ContPhone` TEXT (14) , +
`ContFax` TEXT (14) , +
`ContCell` TEXT (14) , +
`ContPager` TEXT (14) , +
`ContEMail` TEXT (60) , +
`ContInfo` NOTE , +
`ContPhoto` LONG VARBIT , +
`LastContactDate` DATE )
CREATE TABLE `Titles` CASCADE +
(`EmpTID` INTEGER NOT NULL +
('The employee title id number cannot be NULL.') , +
`EmpTitle` TEXT (30) )
CREATE TABLE `Employee` CASCADE +
(`EmpID` INTEGER NOT NULL +
('Cannot have a NULL employee id number.') , +
`EmpTID` INTEGER , +
`EmpFName` TEXT (10) , +
`EmpLName` TEXT (16) , +
`EmpAddress` TEXT (30) , +
`EmpAddress2` TEXT (30) , +

```

```

`EmpCity` TEXT (20) , +
`EmpState` TEXT (2) , +
`EmpZip` TEXT (10) , +
`EmpPhone` TEXT (12) , +
`EmpExt` INTEGER , +
`HireDate` DATE , +
`EntryDate` DATE , +
`ActivEmp` TEXT (2) , +
`WebAddress` TEXT (60) DEFAULT 'C:\' , +
`EmailAddress` TEXT (60) , +
`EmployeePhoto` LONG VARBIT )
CREATE TABLE `StateAbr` +
(`State` TEXT (2) NOT NULL +
('Value for column State cannot be null') , +
`FullState` TEXT (30) )
CREATE TABLE `SecurityTable` +
(`UserID` TEXT (20) , +
`UserLogin` TEXT (20) , +
`UserPassword` TEXT (20) , +
`UserGroupID` INTEGER , +
`RememberPassword` TEXT (1) DEFAULT 'N' , +
`NetUserName` TEXT (30) , +
`ComputerName` TEXT (30) )
CREATE TABLE `AccessLog` +
(`UserLogin` TEXT (20) , +
`NetUserName` TEXT (30) , +
`ComputerName` TEXT (30) , +
`LogDate` DATE , +
`LogTime` TIME )
CREATE TABLE `AllColumns` +
(`CInteger` INTEGER , +
`CReal` REAL , +
`CText` TEXT (8) , +
`CDate` DATE , +
`CTime` TIME , +
`CCurrency` CURRENCY , +
`CDouble` DOUBLE , +
`CNote` NOTE , +
`CNumeric` NUMERIC (13, 0) , +
`CDateTime` DATETIME , +
`CVarChar` LONG VARCHAR , +
`CBitNote` BITNOTE (208) , +
`CVarBit` LONG VARBIT , +
`CComputedDate`= +
(IFNULL(CComputedDate, .#DATE, CComputedDate)
te)) DATE )
CREATE TABLE `EmployeeTest` +
(`EmpID` INTEGER , +
`EmpTID` INTEGER , +
`EmpFName` TEXT (10) , +
`EmpLName` TEXT (16) , +
`EmpAddress` TEXT (30) , +
`EmpAddress2` TEXT (30) , +

```

```
`EmpCity` TEXT (20) , +
`EmpState` TEXT (2) , +
`EmpZip` TEXT (10) , +
`EmpPhone` TEXT (12) , +
`EmpExt` INTEGER , +
`HireDate` DATE , +
`EntryDate` DATE , +
`ActivEmp` TEXT (2) , +
`WebAddress` TEXT (60) , +
`EmailAddress` TEXT (60) , +
`EmployeePhoto` LONG VARBIT , +
`RowBackgroundColor` TEXT (20) , +
`RowFontColor` TEXT (20) )
```

Appendix F.

(To Unload Table Definitions, Constraints, Indexes and Comments for All Tables)

COMMAND: UNLOAD STRUCTURE FOR TABLES

```
CREATE TABLE `Customer` +
(`CustID` INTEGER NOT NULL +
('Customer ID Number Cannot be NULL.')
```

, +
`Company` TEXT (40), +
`CustAddress` TEXT (30), +
`CustCity` TEXT (20), +
`CustState` TEXT (2), +
`CustZip` TEXT (10), +
`CustPhone` TEXT (12), +
`CustFax` TEXT (12), +
`ModLevel` INTEGER, +
`LastUpdateDate` DATE, +
`LastUpdateTime` TIME, +
`LastOrderDate` DATE, +
`CustURL` TEXT (60), +
`CustEmail` TEXT (60), +
`CustStatus` TEXT (1), +
`PaymentTerm` TEXT (20), +
`CreditLimit` CURRENCY)
ALTER TABLE `Customer` ADD PRIMARY KEY +
(`CustID`) +
('The customer id number must be unique for this table.',+
'This customer id number is being referenced in another table, sorry.6',+
'This value is being reference in another table.')
CREATE INDEX CustState ON `Customer` +
(`CustState` ASC SIZE 2)
AUTONUM `CustID` IN `Customer` USING +
0. 0. NONUM
COMMENT ON `CustID` IN `Customer` IS +
'Customer Identification Number'
COMMENT ON `Company` IN `Customer` IS +
'Customer Company Name'
COMMENT ON `CustAddress` IN `Customer` IS +
'Customer Address'
COMMENT ON `CustCity` IN `Customer` IS +
'Customer City'
COMMENT ON `CustState` IN `Customer` IS +
'Customer State'
COMMENT ON `CustZip` IN `Customer` IS +
'Customer Zip Code'
COMMENT ON `CustPhone` IN `Customer` IS +
'Customer Phone Number'
COMMENT ON `CustFax` IN `Customer` IS +
'Customer Fax Number'
COMMENT ON `ModLevel` IN `Customer` IS +
'Customer Mod Level'
COMMENT ON `LastUpdateDate` IN `Customer` IS +
'Last Update Date'

```

COMMENT ON `LastUpdateTime` IN `Customer` IS +
'Last Update Time'
COMMENT ON `LastOrderDate` IN `Customer` IS +
'Last Order Data'
COMMENT ON `CustURL` IN `Customer` IS +
'Customer WWW URL Address'
COMMENT ON `CustEMail` IN `Customer` IS +
'Customer E-Mail Address'
COMMENT ON `CustStatus` IN `Customer` IS +
'Customer Status (A/I)'
COMMENT ON `PaymentTerm` IN `Customer` IS +
'Payment Terms'
COMMENT ON `CreditLimit` IN `Customer` IS +
'Credit Limit'
COMMENT ON TABLE `Customer` IS +
'Customer Information'
CREATE TABLE `Component` +
(`CompID` TEXT (6) NOT NULL +
('The component id number is a primary key, it''s value cannot be NULL.'), +
`CompDesc` TEXT (31) )
ALTER TABLE `Component` ADD PRIMARY KEY +
(`CompID` ) +
('Compid must be a unique value for this table.',+
'Cannot delete this component id number, it is being referenced in another
table+
.',+
'Cannot change this value. It''s current value is being referenced in
another t+
able.')
COMMENT ON `CompID` IN `Component` IS +
'Component identification number'
COMMENT ON `CompDesc` IN `Component` IS +
'Component description'
COMMENT ON TABLE `Component` IS +
'Component Identification Number and Description'
CREATE TABLE `SalesBonus` +
(`EmpID` INTEGER , +
`TransDate` DATE , +
`NetAmount` CURRENCY , +
`BonusPct` REAL , +
`Bonus`= +
( netamount* bonuspct) CURRENCY )
ALTER TABLE `SalesBonus` ADD FOREIGN KEY +
( `EmpID` )+
REFERENCES `Employee` +
('Cannot have an employee id that does not exist in the Employee table.',+
'Cannot change employee id number to a value that does not exist in the
employee+
table.')
COMMENT ON `EmpID` IN `SalesBonus` IS +
'Employee identification number'
COMMENT ON `TransDate` IN `SalesBonus` IS +
'Transaction date'

```

```

COMMENT ON `NetAmount` IN `SalesBonus` IS +
'Net amount of transaction'
COMMENT ON `BonusPct` IN `SalesBonus` IS +
'Bonus percentage'
COMMENT ON `Bonus` IN `SalesBonus` IS +
'Amount of bonus'
COMMENT ON TABLE `SalesBonus` IS +
'Sales Bonus Information'
CREATE TABLE `PaymentTerms` +
(`PaymentTerm` TEXT (20) )
COMMENT ON `PaymentTerm` IN `PaymentTerms` IS +
'Payment Term'
COMMENT ON TABLE `PaymentTerms` IS +
'Payment Terms'
CREATE TABLE `CompUsed` +
(`Model` TEXT (6) , +
`CompID` TEXT (6) )
ALTER TABLE `CompUsed` ADD FOREIGN KEY +
( `Model` )+
REFERENCES `Product` +
('This value must first exist in the Product table.',+
'You cannot update this value to one that does not exist in the Product
table.')
ALTER TABLE `CompUsed` ADD FOREIGN KEY +
( `CompID` )+
REFERENCES `Component` +
('Cannot use a component id number that does not exist in the Component
table.',+
'The value you are trying to update to does not exist in the Component
table.')
COMMENT ON `Model` IN `CompUsed` IS +
'Product model number'
COMMENT ON `CompID` IN `CompUsed` IS +
'Component identification number'
COMMENT ON TABLE `CompUsed` IS +
'Components Used in a Model'
CREATE TABLE `ProdLocation` +
(`Model` TEXT (6) , +
`Location` TEXT (4) , +
`OnHand` INTEGER NOT NULL +
('Value for Onhand cannot be null.'), +
`Cost` CURRENCY , +
`ReOrderPt` INTEGER )
ALTER TABLE `ProdLocation` ADD FOREIGN KEY +
( `Model` )+
REFERENCES `Product` +
('The value of the product number must already exist in the Model table.',+
'Cannot change this value to one that does not exist in the Model table. Add
th+
is value to the Model table first.')
COMMENT ON `Model` IN `ProdLocation` IS +
'Product model number'
COMMENT ON `Location` IN `ProdLocation` IS +

```

```

'Product locations'
COMMENT ON `OnHand` IN `ProdLocation` IS +
'Product quantity on hand'
COMMENT ON `Cost` IN `ProdLocation` IS +
'Product unit cost'
COMMENT ON TABLE `ProdLocation` IS +
'Product Location, Quantity on Hand, and Cost'
RULES 'Value for onhand cannot be less than minimum.' +
      FOR `ProdLocation` SUCCEEDS +
      WHERE ProdLocation.Onhand >= 1
CREATE TABLE `Levels` +
(`ModLevel` INTEGER NOT NULL +
('Value for Modlevel cannot be null.')
```

```

), +
`Model` TEXT (6) NOT NULL +
('Value for Model cannot be null.')
```

```

), +
`LevPerc` REAL
)
ALTER TABLE `Levels` ADD PRIMARY KEY +
(`ModLevel` +
`Model`)
ALTER TABLE `Levels` ADD FOREIGN KEY +
(`Model`)+
REFERENCES `Product` +
('Cannot have a pricing level for a model not in the Model table.',+
'Cannot change to a model that is not in the model table.')
COMMENT ON TABLE `Levels` IS +
'Pricing Level Per Model by Percent'
CREATE TABLE `FormTable` +
(`ColumnZZ` INTEGER )
COMMENT ON TABLE `FormTable` IS +
'Dummy Table for Forms'
CREATE TABLE `BonusRate` +
(`MinAmount` CURRENCY , +
`MaxAmount` CURRENCY , +
`BonusPct` REAL )
COMMENT ON `MinAmount` IN `BonusRate` IS +
'Minimum transaction net amount'
COMMENT ON `MaxAmount` IN `BonusRate` IS +
'Maximum transaction net amount'
COMMENT ON `BonusPct` IN `BonusRate` IS +
'Bonus percentage'
COMMENT ON TABLE `BonusRate` IS +
'Rates for Bonuses'
CREATE TABLE `Product` +
(`Model` TEXT (6) NOT NULL +
('This is a Primary Key, it's value cannot be NULL.')
```

```

), +
`ProdName` TEXT (35) , +
`ProdDesc` TEXT (31) , +
`ListPrice` CURRENCY )
ALTER TABLE `Product` ADD PRIMARY KEY +
(`Model` ) +
('This table requires a unique model number.',+
'This value is being referenced another table.',+
'Cannot change this value, it is being referenced in another table.')
```

```

COMMENT ON `Model` IN `Product` IS +
'Product model number'
COMMENT ON `ProdName` IN `Product` IS +
'Product name'
COMMENT ON `ProdDesc` IN `Product` IS +
'Product description'
COMMENT ON `ListPrice` IN `Product` IS +
'List price of product'
COMMENT ON TABLE `Product` IS +
'Model Information, Including List Price'
CREATE TABLE `InvoiceHeader` +
(`TransID` INTEGER NOT NULL +
('Value for column TransID cannot be null') , +
`CustID` INTEGER , +
`EmpID` INTEGER , +
`TransDate` DATE , +
`BillToCompany` TEXT (40) , +
`BillToAddress` TEXT (30) , +
`BillToCity` TEXT (20) , +
`BillToState` TEXT (2) , +
`BillToZip` TEXT (10) , +
`ShipToCompany` TEXT (40) , +
`ShipToAddress` TEXT (30) , +
`ShipToCity` TEXT (20) , +
`ShipToState` TEXT (2) , +
`ShipToZip` TEXT (10) , +
`NetAmount` CURRENCY , +
`Freight`= +
(netamount* .01) CURRENCY , +
`Tax`= +
(netamount* .081) CURRENCY , +
`InvoiceTotal`= +
(netamount+ freight+ tax) CURRENCY )
ALTER TABLE `InvoiceHeader` ADD PRIMARY KEY +
(`TransID` ) +
('Values for rows in InvoiceHeader must be unique',+
'Cannot delete - values exist in another table',+
'Cannot update - values exist in another table')
ALTER TABLE `InvoiceHeader` ADD FOREIGN KEY +
( `EmpID` )+
REFERENCES `Employee` +
('Cannot insert - value does not exist in Employee',+
'Cannot update - value does not exist in Employee')
ALTER TABLE `InvoiceHeader` ADD FOREIGN KEY +
( `CustID` )+
REFERENCES `Customer` +
('Cannot insert - value does not exist in Customer',+
'Cannot update - value does not exist in Customer')
COMMENT ON TABLE `InvoiceHeader` IS +
'Invoice Header Information'
CREATE TABLE `PrintOptions` +
(`Option` TEXT (8) , +
`DisplaySequence` INTEGER , +

```

```

`DisplayName` TEXT (24) , +
`DisplayIcon` LONG VARBIT , +
`PrintOption` TEXT (8) , +
`ExtendedSyntax` TEXT (200) )
COMMENT ON `Option` IN `PrintOptions` IS +
'Option'
COMMENT ON `DisplaySequence` IN `PrintOptions` IS +
'Display Sequence'
COMMENT ON `DisplayName` IN `PrintOptions` IS +
'Display Name'
COMMENT ON `DisplayIcon` IN `PrintOptions` IS +
'Display Icon'
COMMENT ON `PrintOption` IN `PrintOptions` IS +
'Print Option'
COMMENT ON `ExtendedSyntax` IN `PrintOptions` IS +
'Extended Syntax'
COMMENT ON TABLE `PrintOptions` IS +
'Print Options'
CREATE TABLE `InvoiceDetail` +
(`TransID` INTEGER , +
`DetailNum` INTEGER , +
`Model` TEXT (6) , +
`Units` INTEGER , +
`Price` CURRENCY , +
`Discount` REAL , +
`SalePrice`= +
(Price- (Price*Discount/100)) CURRENCY , +
`ExtPrice`= +
(Units* SalePrice) CURRENCY )
ALTER TABLE `InvoiceDetail` ADD FOREIGN KEY +
( `TransID` )+
REFERENCES `InvoiceHeader` +
('Cannot insert - value does not exist in InvoiceHeader',+
'Cannot update - value does not exist in InvoiceHeader')
ALTER TABLE `InvoiceDetail` ADD FOREIGN KEY +
( `Model` )+
REFERENCES `Product` +
('Cannot insert - value does not exist in Product',+
'Cannot update - value does not exist in Product')
COMMENT ON TABLE `InvoiceDetail` IS +
'Invoice Detail Information'
CREATE TABLE `ContactCallNotes` +
(`ContID` INTEGER , +
`EmpID` INTEGER , +
`CallDate` DATE , +
`CallTime` TIME NOT NULL +
('Value for column ContTime cannot be null') , +
`CallNotes` NOTE )
ALTER TABLE `ContactCallNotes` ADD FOREIGN KEY +
( `EmpID` )+
REFERENCES `Employee` +
('Cannot insert - value does not exist in Employee',+
'Cannot update - value does not exist in Employee')

```

```

ALTER TABLE `ContactCallNotes` ADD FOREIGN KEY +
  ( `ContID` )+
  REFERENCES `Contact` +
('Cannot insert - value does not exist in Contact',+
 'Cannot update - value does not exist in Contact')
COMMENT ON `ContID` IN `ContactCallNotes` IS +
'Contact ID'
COMMENT ON `EmpID` IN `ContactCallNotes` IS +
'Employee (Sales Rep ID)'
COMMENT ON `CallDate` IN `ContactCallNotes` IS +
'Call Date'
COMMENT ON `CallTime` IN `ContactCallNotes` IS +
'Call Time'
COMMENT ON `CallNotes` IN `ContactCallNotes` IS +
'Call Notes'
COMMENT ON TABLE `ContactCallNotes` IS +
'Contact Call Notes'
CREATE TABLE `LicenseInformation` +
(`LicenseNumber` TEXT (16) , +
 `ConcurrentUsers` INTEGER , +
 `CompanyName` TEXT (60) , +
 `CompAddressLine1` TEXT (60) , +
 `CompAddressLine2` TEXT (60) , +
 `CompCity` TEXT (20) , +
 `CompStateProvince` TEXT (20) , +
 `CompPostalZipCode` TEXT (20) , +
 `CompCountry` TEXT (20) , +
 `ContactLastName` TEXT (20) , +
 `ContactFirstName` TEXT (20) , +
 `ContactMiddleName` TEXT (20) , +
 `ContactPhone` TEXT (20) , +
 `ContactFax` TEXT (20) , +
 `ContactEMail` TEXT (60) , +
 `ContactWEBAddress` TEXT (60) , +
 `DateSold` DATE , +
 `DateInstalled` DATE , +
 `SupportDaysLeft` INTEGER )
COMMENT ON `LicenseNumber` IN `LicenseInformation` IS +
'License Number'
COMMENT ON `ConcurrentUsers` IN `LicenseInformation` IS +
'Number of Concurrent Users Allowed'
COMMENT ON `CompanyName` IN `LicenseInformation` IS +
'Company Name'
COMMENT ON `CompAddressLine1` IN `LicenseInformation` IS +
'Address Line1'
COMMENT ON `CompAddressLine2` IN `LicenseInformation` IS +
'Address Line 2'
COMMENT ON `CompCity` IN `LicenseInformation` IS +
'City'
COMMENT ON `CompStateProvince` IN `LicenseInformation` IS +
'State/Province'
COMMENT ON `CompPostalZipCode` IN `LicenseInformation` IS +
'Postal/Zip Code'

```

```

COMMENT ON `CompCountry` IN `LicenseInformation` IS +
'Country'
COMMENT ON `ContactLastName` IN `LicenseInformation` IS +
'Contact Last Name'
COMMENT ON `ContactFirstName` IN `LicenseInformation` IS +
'Contact First Name'
COMMENT ON `ContactMiddleName` IN `LicenseInformation` IS +
'Contact Middle Name'
COMMENT ON `ContactPhone` IN `LicenseInformation` IS +
'Contact Phone Number'
COMMENT ON `ContactFax` IN `LicenseInformation` IS +
'Contact Phone Number'
COMMENT ON `ContactEMail` IN `LicenseInformation` IS +
'Contact E-Mail Address'
COMMENT ON `ContactWEBAddress` IN `LicenseInformation` IS +
'ContactWEBAddress'
COMMENT ON `DateSold` IN `LicenseInformation` IS +
'Date Product Sold'
COMMENT ON `DateInstalled` IN `LicenseInformation` IS +
'Date Product Installed'
COMMENT ON `SupportDaysLeft` IN `LicenseInformation` IS +
'Support Days Left'
COMMENT ON TABLE `LicenseInformation` IS +
'Running R:BASE Your Way Cutomer License Information'
CREATE TABLE `Contact` +
(`CustID` INTEGER , +
`ContID` INTEGER NOT NULL +
('Value for column ContID cannot be null') , +
`ContFName` TEXT (10) , +
`ContLName` TEXT (16) , +
`ContPhone` TEXT (14) , +
`ContFax` TEXT (14) , +
`ContCell` TEXT (14) , +
`ContPager` TEXT (14) , +
`ContEMail` TEXT (60) , +
`ContInfo` NOTE , +
`ContPhoto` LONG VARBIT , +
`LastContactDate` DATE )
ALTER TABLE `Contact` ADD PRIMARY KEY +
(`ContID` ) +
('Values for rows in Contact must be unique',+
'Cannot delete - values exist in another table',+
'Cannot update - values exist in another table')
ALTER TABLE `Contact` ADD FOREIGN KEY +
( `CustID` )+
REFERENCES `Customer`
AUTONUM `ContID` IN `Contact` USING +
1039. 1. NONUM
COMMENT ON `CustID` IN `Contact` IS +
'Customer Identification Number'
COMMENT ON `ContID` IN `Contact` IS +
'Contact Automated ID'
COMMENT ON `ContFName` IN `Contact` IS +

```

```

'Customer Contact First Name'
COMMENT ON `ContLName` IN `Contact` IS +
'Customer Contact Last Name'
COMMENT ON `ContPhone` IN `Contact` IS +
'Customer Contact Phone Number'
COMMENT ON `ContFax` IN `Contact` IS +
'Customer Contact Fax Number'
COMMENT ON `ContCell` IN `Contact` IS +
'Customer Contact Cellular Number'
COMMENT ON `ContPager` IN `Contact` IS +
'Customer Contact Pager Number'
COMMENT ON `ContEMail` IN `Contact` IS +
'Customer Contact E-Mail Address'
COMMENT ON `ContInfo` IN `Contact` IS +
'Customer Contact Notes'
COMMENT ON `ContPhoto` IN `Contact` IS +
'Customer Contact Photo'
COMMENT ON `LastContactDate` IN `Contact` IS +
'Last Contact Date'
COMMENT ON TABLE `Contact` IS +
'Customer Contact Information'
CREATE TABLE `Titles` CASCADE +
(`EmpTID` INTEGER NOT NULL +
('The employee title id number cannot be NULL. '), +
`EmpTitle` TEXT (30) )
ALTER TABLE `Titles` ADD PRIMARY KEY +
(`EmpTID` ) +
('Employee title id number must be unique.',+
'Cannot delete this title id number, it is currently being referenced.',+
'Cannot change this value, it is being referenced in another table. ')
AUTONUM `EmpTID` IN `Titles` USING +
9. 1. NONUM
COMMENT ON TABLE `Titles` IS +
'Employee Titles List'
CREATE TABLE `Employee` CASCADE +
(`EmpID` INTEGER NOT NULL +
('Cannot have a NULL employee id number. '), +
`EmpTID` INTEGER , +
`EmpFName` TEXT (10) , +
`EmpLName` TEXT (16) , +
`EmpAddress` TEXT (30) , +
`EmpAddress2` TEXT (30) , +
`EmpCity` TEXT (20) , +
`EmpState` TEXT (2) , +
`EmpZip` TEXT (10) , +
`EmpPhone` TEXT (12) , +
`EmpExt` INTEGER , +
`HireDate` DATE , +
`EntryDate` DATE , +
`ActivEmp` TEXT (2) , +
`WebAddress` TEXT (60) DEFAULT 'C:\' , +
`EmailAddress` TEXT (60) , +
`EmployeePhoto` LONG VARBIT )

```

```

ALTER TABLE `Employee` ADD PRIMARY KEY +
(`EmpID` ) +
('There must be a unique id number per employee.',+
'You cannot delete an employee id that is being referenced in another
table.',+
'Cannot change employee id value that is referenced in another table.')
ALTER TABLE `Employee` ADD FOREIGN KEY +
( `EmpTID` )+
REFERENCES `Titles` +
('You must enter an employee id number that exists in the Titles table.',+
'You cannot update the title id to this value, use a value that exists in the
Ti+
tles table.')
AUTONUM `EmpID` IN `Employee` USING +
116. 1. NONUM
COMMENT ON `EmpID` IN `Employee` IS +
'Employee Identification Number'
COMMENT ON `EmpTID` IN `Employee` IS +
'Employee Title ID'
COMMENT ON `EmpFName` IN `Employee` IS +
'Employee First Name'
COMMENT ON `EmpLName` IN `Employee` IS +
'Employee Last Name'
COMMENT ON `EmpAddress` IN `Employee` IS +
'Employee Home Address Line 1'
COMMENT ON `EmpAddress2` IN `Employee` IS +
'Employee Home Address Line 2'
COMMENT ON `EmpCity` IN `Employee` IS +
'Employee Home City'
COMMENT ON `EmpState` IN `Employee` IS +
'Employee Home State'
COMMENT ON `EmpZip` IN `Employee` IS +
'Employee Home Zip Code'
COMMENT ON `EmpPhone` IN `Employee` IS +
'Employee Home Phone Number'
COMMENT ON `EmpExt` IN `Employee` IS +
'Employee Work Phone Extension Number'
COMMENT ON `HireDate` IN `Employee` IS +
'Employee Date of Hire'
COMMENT ON `EntryDate` IN `Employee` IS +
'Date Employee Information Entered'
COMMENT ON `ActivEmp` IN `Employee` IS +
'Employee Current Status (A)ctive, (I)nactive'
COMMENT ON `WebAddress` IN `Employee` IS +
'Favorite WEB Address'
COMMENT ON `EmailAddress` IN `Employee` IS +
'E-Mail Address'
COMMENT ON `EmployeePhoto` IN `Employee` IS +
'Employee Photo'
COMMENT ON TABLE `Employee` IS +
'Employee Information'
CREATE TABLE `StateAbr` +
(`State` TEXT (2) NOT NULL +

```

```

('Value for column State cannot be null') , +
`FullState` TEXT (30) )
ALTER TABLE `StateAbr` ADD PRIMARY KEY +
(`State` ) +
('Values for rows in StateAbr must be unique',+
'Cannot delete - values exist in another table',+
'Cannot update - values exist in another table')
COMMENT ON `State` IN `StateAbr` IS +
'State abbreviations'
COMMENT ON `FullState` IN `StateAbr` IS +
'State name'
COMMENT ON TABLE `StateAbr` IS +
'State Names and Abbreviations'
CREATE TABLE `SecurityTable` +
(`UserID` TEXT (20) , +
`UserLogin` TEXT (20) , +
`UserPassword` TEXT (20) , +
`UserGroupID` INTEGER , +
`RememberPassword` TEXT (1) DEFAULT 'N' , +
`NetUserName` TEXT (30) , +
`ComputerName` TEXT (30) )
COMMENT ON TABLE `SecurityTable` IS +
'RRBYW14 - Application MainMenu Security/Login'
CREATE TABLE `AccessLog` +
(`UserLogin` TEXT (20) , +
`NetUserName` TEXT (30) , +
`ComputerName` TEXT (30) , +
`LogDate` DATE , +
`LogTime` TIME )
COMMENT ON `UserLogin` IN `AccessLog` IS +
'User Name'
COMMENT ON `NetUserName` IN `AccessLog` IS +
'Network User Name'
COMMENT ON `ComputerName` IN `AccessLog` IS +
'Computer Name'
COMMENT ON `LogDate` IN `AccessLog` IS +
'Log Date'
COMMENT ON `LogTime` IN `AccessLog` IS +
'LogTime'
COMMENT ON TABLE `AccessLog` IS +
'JobTrack - User Access Transaction Table'
CREATE TABLE `AllColumns` +
(`CInteger` INTEGER , +
`CReal` REAL , +
`CText` TEXT (8) , +
`CDate` DATE , +
`CTime` TIME , +
`CCurrency` CURRENCY , +
`CDouble` DOUBLE , +
`CNote` NOTE , +
`CNumeric` NUMERIC (13, 0) , +
`CDateTime` DATETIME , +
`CVarChar` LONG VARCHAR , +

```

```

`CBitNote` BITNOTE (208) , +
`CVarBit` LONG VARBIT , +
`CComputedDate`= +
(IFNULL(CComputedDate, .#DATE, CComputedDate)
te)) DATE )
COMMENT ON TABLE `AllColumns` IS +
'One Table with All Column Data Types for Testing'
CREATE TABLE `EmployeeTest` +
(`EmpID` INTEGER , +
`EmpTID` INTEGER , +
`EmpFName` TEXT (10) , +
`EmpLName` TEXT (16) , +
`EmpAddress` TEXT (30) , +
`EmpAddress2` TEXT (30) , +
`EmpCity` TEXT (20) , +
`EmpState` TEXT (2) , +
`EmpZip` TEXT (10) , +
`EmpPhone` TEXT (12) , +
`EmpExt` INTEGER , +
`HireDate` DATE , +
`EntryDate` DATE , +
`ActivEmp` TEXT (2) , +
`WebAddress` TEXT (60) , +
`EmailAddress` TEXT (60) , +
`EmployeePhoto` LONG VARBIT , +
`RowBackgroundColor` TEXT (20) , +
`RowFontColor` TEXT (20) )
COMMENT ON TABLE `EmployeeTest` IS +
'Employee Information'

```

Appendix G.

(To Unload Multiple Table Definitions, Constraints, Indexes and Comments using Wild Card)

COMMAND: UNLOAD STRUCTURE FOR TABLES CO%

```
CREATE TABLE `Component` +
(`CompID` TEXT (6) NOT NULL +
('The component id number is a primary key, it's value cannot be NULL.') , +
`CompDesc` TEXT (31) )
ALTER TABLE `Component` ADD PRIMARY KEY +
(`CompID` ) +
('Compid must be a unique value for this table.',+
'Cannot delete this component id number, it is being referenced in another
table+
.',+
'Cannot change this value. It's current value is being referenced in
another t+
able.')
COMMENT ON `CompID` IN `Component` IS +
'Component identification number'
COMMENT ON `CompDesc` IN `Component` IS +
'Component description'
COMMENT ON TABLE `Component` IS +
'Component Identification Number and Description'
CREATE TABLE `CompUsed` +
(`Model` TEXT (6) , +
`CompID` TEXT (6) )
ALTER TABLE `CompUsed` ADD FOREIGN KEY +
( `Model` )+
REFERENCES `Product` +
('This value must first exist in the Product table.',+
'You cannot update this value to one that does not exist in the Product
table.')
ALTER TABLE `CompUsed` ADD FOREIGN KEY +
( `CompID` )+
REFERENCES `Component` +
('Cannot use a component id number that does not exist in the Component
table.',+
'The value you are trying to update to does not exist in the Component
table.')
COMMENT ON `Model` IN `CompUsed` IS +
'Product model number'
COMMENT ON `CompID` IN `CompUsed` IS +
'Component identification number'
COMMENT ON TABLE `CompUsed` IS +
'Components Used in a Model'
CREATE TABLE `ContactCallNotes` +
(`ContID` INTEGER , +
`EmpID` INTEGER , +
`CallDate` DATE , +
`CallTime` TIME NOT NULL +
('Value for column ContTime cannot be null') , +
`CallNotes` NOTE )
```

```

ALTER TABLE `ContactCallNotes` ADD FOREIGN KEY +
  ( `EmpID` )+
  REFERENCES `Employee` +
  ('Cannot insert - value does not exist in Employee',+
  'Cannot update - value does not exist in Employee')
ALTER TABLE `ContactCallNotes` ADD FOREIGN KEY +
  ( `ContID` )+
  REFERENCES `Contact` +
  ('Cannot insert - value does not exist in Contact',+
  'Cannot update - value does not exist in Contact')
COMMENT ON `ContID` IN `ContactCallNotes` IS +
  'Contact ID'
COMMENT ON `EmpID` IN `ContactCallNotes` IS +
  'Employee (Sales Rep ID)'
COMMENT ON `CallDate` IN `ContactCallNotes` IS +
  'Call Date'
COMMENT ON `CallTime` IN `ContactCallNotes` IS +
  'Call Time'
COMMENT ON `CallNotes` IN `ContactCallNotes` IS +
  'Call Notes'
COMMENT ON TABLE `ContactCallNotes` IS +
  'Contact Call Notes'
CREATE TABLE `Contact` +
  (`CustID` INTEGER , +
  `ContID` INTEGER NOT NULL +
  ('Value for column ContID cannot be null') , +
  `ContFName` TEXT (10) , +
  `ContLName` TEXT (16) , +
  `ContPhone` TEXT (14) , +
  `ContFax` TEXT (14) , +
  `ContCell` TEXT (14) , +
  `ContPager` TEXT (14) , +
  `ContEMail` TEXT (60) , +
  `ContInfo` NOTE , +
  `ContPhoto` LONG VARBIT , +
  `LastContactDate` DATE )
ALTER TABLE `Contact` ADD PRIMARY KEY +
  (`ContID` ) +
  ('Values for rows in Contact must be unique',+
  'Cannot delete - values exist in another table',+
  'Cannot update - values exist in another table')
ALTER TABLE `Contact` ADD FOREIGN KEY +
  ( `CustID` )+
  REFERENCES `Customer`
AUTONUM `ContID` IN `Contact` USING +
  1039. 1. NONUM
COMMENT ON `CustID` IN `Contact` IS +
  'Customer Identification Number'
COMMENT ON `ContID` IN `Contact` IS +
  'Contact Automated ID'
COMMENT ON `ContFName` IN `Contact` IS +
  'Customer Contact First Name'
COMMENT ON `ContLName` IN `Contact` IS +

```

```
'Customer Contact Last Name'  
COMMENT ON `ContPhone` IN `Contact` IS +  
'Customer Contact Phone Number'  
COMMENT ON `ContFax` IN `Contact` IS +  
'Customer Contact Fax Number'  
COMMENT ON `ContCell` IN `Contact` IS +  
'Customer Contact Cellular Number'  
COMMENT ON `ContPager` IN `Contact` IS +  
'Customer Contact Pager Number'  
COMMENT ON `ContEMail` IN `Contact` IS +  
'Customer Contact E-Mail Address'  
COMMENT ON `ContInfo` IN `Contact` IS +  
'Customer Contact Notes'  
COMMENT ON `ContPhoto` IN `Contact` IS +  
'Customer Contact Photo'  
COMMENT ON `LastContactDate` IN `Contact` IS +  
'Last Contact Date'  
COMMENT ON TABLE `Contact` IS +  
'Customer Contact Information'
```

Appendix H.

(To Unload Definition of a Given View)

COMMAND: UNLOAD STRUCTURE FOR VIEW CustomerView

```
CREATE VIEW `CustomerView` +  
(Company,CustID,CustAddress,CustCity,CustState,Cus+  
tZip,CustCSZ) AS +  
SELECT Company,CustID,CustAddress,CustCity,CustSta+  
te,CustZip (CustCity+', '&CustState&CustZip) FROM C+  
ustomer ORDER BY Company ASC  
COMMENT ON TABLE `CustomerView` IS +  
'Sorted List of Customers'
```

Appendix I.

(To Structure of a Given View - Traditional)

COMMAND: UNLOAD STRUCTURE FOR CustomerView

```
SET QUOTES=NULL
SET QUOTES='
SET DELIMIT=NULL
SET DELIMIT=', '
SET LINEEND=NULL
SET LINEEND='p'
SET SEMI=NULL
SET SEMI=';'
SET PLUS=NULL
SET PLUS='+'
SET SINGLE=NULL
SET SINGLE='_ '
SET MANY=NULL
SET MANY='% '
SET IDQUOTES=NULL
SET IDQUOTES='`'
SET CURRENCY '$' PREF 2 B
CREATE VIEW `CustomerView` +
(Company,CustID,CustAddress,CustCity,CustState,Cus+
tZip,CustCSZ) AS +
SELECT Company,CustID,CustAddress,CustCity,CustSta+
te,CustZip (CustCity+', '&CustState&CustZip) FROM C+
ustomer ORDER BY Company ASC
COMMENT ON TABLE `CustomerView` IS +
'Sorted List of Customers'
```

Appendix J.

(To Unload Definition of aLL Views)

```
COMMAND:          UNLOAD STRUCTURE FOR VIEWS

CREATE VIEW `CustomerView` +
(Company,CustID,CustAddress,CustCity,CustState,Cust+
tZip,CustCSZ) AS +
SELECT Company,CustID,CustAddress,CustCity,CustSta+
te,CustZip (CustCity+', '&CustState&CustZip) FROM C+
ustomer ORDER BY Company ASC
COMMENT ON TABLE `CustomerView` IS +
'Sorted List of Customers'
CREATE VIEW `ListOfReportsView` +
(ReportName,ReportTableName) AS +
SELECT Sys_Report_Name,Sys_Table_Name FROM Sys_Rep+
orts3 ORDER BY Sys_Report_Name
COMMENT ON TABLE `ListOfReportsView` IS +
'List of Reports to be Shown in Combo/List Boxes'
CREATE VIEW `ProdAlias` +
(TransDate,BillToCompany,BillToAddress,BillToCity,+
BillToState,BillToZip,ShipToCompany,ShipToAddress+
,ShipToCity,ShipToState, ShipToZip,Transid,DetailN+
um,Model,Units,Price,Discount,SalePrice,ExtPrice, +
CustID,Company,CustAddress,CustCity,CustState,Cust+
Zip,CustPhone,ModLevel, EmpID,EmpFName,EmpLName,Pr+
odName) AS +
SELECT T1.TransDate,T1.BillToCompany,T1.BillToAddr+
ess,T1.BillToCity,T1.BillToState,T1.BillToZip,T1.S+
hipToCompany,T1.ShipToAddress,T1.ShipToCity,T1.Shi+
pToState,T1.ShipToZip,T2.Transid,T2.DetailNum,T2.M+
odel,T2.Units,T2.Price,T2.Discount,T2.SalePrice,T2+
.ExtPrice,T3.CustID,T3.Company,T3.CustAddress,T3.C+
ustCity,T3.CustState,T3.CustZip,T3.CustPhone,T3.Mo+
dLevel,T4.EmpID,T4.EmpFName,T4.EmpLName,T5.ProdNam+
e FROM InvoiceHeader T1,InvoiceDetail T2,Customer +
T3,Employee T4,Product T5 WHERE T1.TransID = T2.Tr+
ansID AND T1.CustID = T3.CustID AND T1.EmpID = T4.+
EmpID AND T2.Model = T5.Model
COMMENT ON TABLE `ProdAlias` IS +
'Products Sold by Date - View'
CREATE VIEW `ProdView` +
(TransDate, TransID,DetailNum,Model,Units,Price,D+
iscount,SalePrice,ExtPrice, CustID,Company,CustAd+
dress,CustCity,CustState,CustZip,CustPhone,ModLeve+
l, LastUpdateDate,LastUpdateTime,LastOrderDate,Cu+
stURL,CustEmail,CustStatus, PaymentTerm,CreditLim+
it, EmpID,EmpFName,EmpLName,ProdName) AS +
SELECT T1.TransDate,T2.TransID,T2.DetailNum,T2.Mod+
el,T2.Units,T2.Price,T2.Discount,T2.SalePrice,T2.E+
xtPrice,T3.CustID,T3.Company,T3.CustAddress,T3.Cus+
tCity,T3.CustState,T3.CustZip,T3.CustPhone,T3.ModL+
evel,T3.LastUpdateDate,T3.LastUpdateTime,T3.LastOr+
```

```

derDate,T3.CustURL,T3.CustEMail,T3.CustStatus,T3.P+
aymentTerm,T3.CreditLimit,T4.EmpID,T4.EmpFName,T4.+
EmpLName,T5.ProdName FROM InvoiceHeader T1,Invoice+
Detail T2,Customer T3,Employee T4,Product T5 WHERE+
((T1.TransID = T2.TransID) AND (T1.CustID = T3.C+
ustID ) AND (T1.EmpID = T4.EmpID) AND (T2.Model =+
T5.Model)))
COMMENT ON TABLE `ProdView` IS +
'Sales by Products - View'
CREATE VIEW `SalesByEmployee` +
(EmpLName,EmpFName,EmpID,TransID,TransDate,NetAmou+
nt) AS +
SELECT T1.EmpLName,T1.EmpFName,T2.EmpID,T2.TransID+
,T2.TransDate,T2.NetAmount FROM Employee T1,Invoic+
eHeader T2 WHERE T2.EmpID = T1.EmpID
COMMENT ON TABLE `SalesByEmployee` IS +
'Products Sold by Employee - View'
CREATE VIEW `SalesData` +
(Company,CustID,TransID,EmpID,TransDate,NetAmount,+
Freight,Tax,InvoiceTotal) AS +
SELECT T1.Company,T2.CustID,T2.TransID,T2.EmpID,T2+
.TransDate,T2.NetAmount,T2.Freight,T2.Tax,T2.Invoi+
ceTotal FROM InvoiceHeader T2,Customer T1 WHERE T1+
.CustID = T2.CustID ORDER BY T1.CustID ASC
COMMENT ON TABLE `SalesData` IS +
'Sales Data by Company'
CREATE VIEW `SalesDataByCompany` +
(Company,CustID,TransID,EmpID,TransDate, NetAmount+
,Freight,Tax,InvoiceTotal) AS +
SELECT T1.Company,T2.CustID,T2.TransID,T2.EmpID,T2+
.TransDate,T2.NetAmount,T2.Freight,T2.Tax,T2.Invoi+
ceTotal FROM InvoiceHeader T2,Customer T1 WHERE T1+
.CustID = T2.CustID ORDER BY T1.Company ASC
COMMENT ON TABLE `SalesDataByCompany` IS +
'Sales Data by Company'
CREATE VIEW `SalesInf` +
(EmpID,TransID,TransDate,NetAmount) AS +
SELECT T1.EmpID,T2.TransID,T2.TransDate,T2.NetAmou+
nt FROM Employee T1,InvoiceHeader T2 WHERE T2.EmpI+
D = T1.EmpID
COMMENT ON TABLE `SalesInf` IS +
'Products Sold by Employee - View'
CREATE VIEW `TCrossTabView` +
(EmpLName,Company,ExtPrice,ProdName) AS +
SELECT T1.EmpLName,T2.Company,T4.ExtPrice,T5.ProdN+
ame FROM Employee T1,Customer T2,InvoiceHeader T3,+
InvoiceDetail T4,Product T5 WHERE T3.EmpID = T1.Em+
pID AND T3.CustID = T2.CustID AND T4.TransID = T3.+
TransID AND T5.Model = T4.Model
COMMENT ON TABLE `TCrossTabView` IS +
'CrossTab Data - View'
CREATE VIEW `YT3` +
(CustID,Company,NetAmount,Freight,Tax,InvoiceTotal+

```

```

) AS +
SELECT T35.CustID,T9.Company,(SUM(T35.NetAmount)),+
(SUM(T35.Freight)),(SUM(T35.Tax)),(SUM(T35.Invoice+
Total)) FROM InvoiceHeader T35,Customer T9 WHERE T+
35.CustID = T9.CustID ORDER BY Company ASC GROUP B+
Y T35.CustID,T9.Company
COMMENT ON TABLE `YT3` IS +
'YTD Invoice Summary'
CREATE VIEW `NewCustomer` +
(CustID, Company, CustAddress, CustCSZ, CustPhone,+
CustURL, CustEMail) AS +
SELECT T5.CustID,T5.Company,T5.CustAddress,(T5.Cus+
tCity+', '+T5.CustState&T5.CustZip),T5.CustPhone,T5+
.CustURL,T5.CustEMail FROM Customer T5
COMMENT ON TABLE `NewCustomer` IS +
'Test View for Sub-Reports'
CREATE VIEW `NewInvoiceHeader` +
(TransID,CustID,EmpID,TransDate,BillToCompany,Bill+
ToAddress,BillToCSZ, ShipToCompany,ShipToCSZ,ShipT+
oAddress,NetAmount,Freight,Tax,InvoiceTotal) AS +
SELECT T10.TransID,T10.CustID,T10.EmpID,T10.TransD+
ate,T10.BillToCompany,T10.BillToAddress,(T10.BillT+
oCity + ',' + T10.BillToState & T10.BillToZip),T10+
.ShipToCompany,T10.ShipToAddress,(T10.ShipToCity+'+
','+T10.ShipToState&T10.ShipToZip),T10.NetAmount,Tl+
0.Freight,T10.Tax,T10.InvoiceTotal FROM InvoiceHea+
der T10
COMMENT ON TABLE `NewInvoiceHeader` IS +
'Test View for Sub-Reports'
CREATE VIEW `NewInvoiceDetails` +
(TransID,DetailNum,Model,ProdName,ProdDesc,ListPri+
ce,Units,Price,Discount, SalePrice,ExtPrice) AS +
SELECT t10.TransID,t10.DetailNum,t21.Model,t21.Pro+
dName,t21.ProdDesc,t21.ListPrice,t10.Units,t10.Pri+
ce,t10.Discount,t10.SalePrice,t10.ExtPrice FROM In+
voiceDetail t10,Product t21 WHERE t10.Model = t21.+
Model
COMMENT ON TABLE `NewInvoiceDetails` IS +
'Test View for Sub-Reports'
CREATE VIEW `NewEmployee` +
(EmpID,EmpFullName,EmpAddress,EmpCity,EmpState,Emp+
Zip,EmpPhone, EMailAddress,EmpFullAddress) AS +
SELECT T9.EmpID,(T9.EmpFName & T9.EmpLName),T9.Emp+
Address,T9.EmpCity,T9.EmpState,T9.EmpZip,T9.EmpPho+
ne,T9.EMailAddress,(T9.EmpCity+', '+T9.EmpState&T9.+
EmpZip) FROM Employee T9
COMMENT ON TABLE `NewEmployee` IS +
'Test View for Sub-Reports'
CREATE VIEW `NewContact` +
(CustID,ContFullName,ContPhone) AS +
SELECT CustID,(ContFName&ContLName),ContPhone FROM+
Contact
COMMENT ON TABLE `NewContact` IS +

```

```

'Quick List of Customer Contacts'
CREATE VIEW `CustomerList` +
(CustID, Company) AS +
SELECT CustID,Company FROM Customer ORDER BY CustID
D
CREATE VIEW `YTDInvoiceTotal` +
(CustID,Company,YTDNetAmount,YTDFreight,YTDTax,YTD+
InvoiceTotal) AS +
SELECT CustID,BillToCompany,(SUM(NetAmount)),(SUM(+
Freight)),(SUM(Tax)),(SUM(InvoiceTotal)) FROM Invo+
iceHeader GROUP BY CustID,BillToCompany
COMMENT ON TABLE `YTDInvoiceTotal` IS +
'Year-To-Date Invoice Totals by Customer'
CREATE VIEW `QuarterlySummary` +
(Quarter,TotalSale) AS +
SELECT 1,(SUM(InvoiceTotal)) FROM InvoiceHeader WH+
ERE TransDate BETWEEN 01/01/2006 AND 03/31/2006 UN+
ION SELECT 2,(SUM(InvoiceTotal)) FROM InvoiceHeade+
r WHERE TransDate BETWEEN 04/01/2006 AND 06/30/200+
6 UNION SELECT 3,(SUM(InvoiceTotal)) FROM InvoiceH+
eader WHERE TransDate BETWEEN 07/01/2006 AND 09/30+
/2006 UNION SELECT 4,(SUM(InvoiceTotal)) FROM Invo+
iceHeader WHERE TransDate BETWEEN 10/01/2006 AND 1+
2/31/2006
COMMENT ON TABLE `QuarterlySummary` IS +
'2006 Quarterly Summary View'

```

Appendix K.

(To Unload Definition of Multiple Views using Wild Card)

COMMAND: UNLOAD STRUCTURE FOR VIEWS New%

```
CREATE VIEW `NewCustomer` +
(CustID, Company, CustAddress, CustCSZ, CustPhone,+
CustURL, CustEmail) AS +
SELECT T5.CustID,T5.Company,T5.CustAddress,(T5.Cus+
tCity+', '+T5.CustState&T5.CustZip),T5.CustPhone,T5+
.CustURL,T5.CustEmail FROM Customer T5
COMMENT ON TABLE `NewCustomer` IS +
'Test View for Sub-Reports'
CREATE VIEW `NewInvoiceHeader` +
(TransID,CustID,EmpID,TransDate,BillToCompany,Bill+
ToAddress,BillToCSZ, ShipToCompany,ShipToCSZ,ShipT+
oAddress,NetAmount,Freight,Tax,InvoiceTotal) AS +
SELECT T10.TransID,T10.CustID,T10.EmpID,T10.TransD+
ate,T10.BillToCompany,T10.BillToAddress,(T10.BillT+
oCity + ',' + T10.BillToState & T10.BillToZip),T10+
.ShipToCompany,T10.ShipToAddress,(T10.ShipToCity+'+
'+T10.ShipToState&T10.ShipToZip),T10.NetAmount,Tl+
0.Freight,T10.Tax,T10.InvoiceTotal FROM InvoiceHea+
der T10
COMMENT ON TABLE `NewInvoiceHeader` IS +
'Test View for Sub-Reports'
CREATE VIEW `NewInvoiceDetails` +
(TransID,DetailNum,Model,ProdName,ProdDesc,ListPri+
ce,Units,Price,Discount, SalePrice,ExtPrice) AS +
SELECT t10.TransID,t10.DetailNum,t21.Model,t21.Pro+
dName,t21.ProdDesc,t21.ListPrice,t10.Units,t10.Pri+
ce,t10.Discount,t10.SalePrice,t10.ExtPrice FROM In+
voiceDetail t10,Product t21 WHERE t10.Model = t21.+
Model
COMMENT ON TABLE `NewInvoiceDetails` IS +
'Test View for Sub-Reports'
CREATE VIEW `NewEmployee` +
(EmpID,EmpFullName,EmpAddress,EmpCity,EmpState,Emp+
Zip,EmpPhone, EMailAddress,EmpFullAddress) AS +
SELECT T9.EmpID,(T9.EmpFName & T9.EmpLName),T9.Emp+
Address,T9.EmpCity,T9.EmpState,T9.EmpZip,T9.EmpPho+
ne,T9.EMailAddress,(T9.EmpCity+', '+T9.EmpState&T9.+
EmpZip) FROM Employee T9
COMMENT ON TABLE `NewEmployee` IS +
'Test View for Sub-Reports'
CREATE VIEW `NewContact` +
(CustID,ContFullName,ContPhone) AS +
SELECT CustID,(ContFName&ContLName),ContPhone FROM+
Contact
COMMENT ON TABLE `NewContact` IS +
'Quick List of Customer Contacts'
```