

Wind Riders of the Lost River Range

Database Proposal



Team 4: Nirajan Bhattarai and Devon Breshears

INFO 5507

Database Design and Implementation

Spring 2021

Table of Contents

Executive Summary.....	4
Goals	4
Problems	4
Results.....	4
Recommendations	4
Introduction	5
Company Background	5
Problem.....	5
Entity-Relationship Diagram	6
Normalized Schema	7
Implementation	8
Database Table Creation.....	8
Accident Table.....	8
Credit Card Table.....	9
Customer Table.....	10
Discount Retailer Table.....	11
Employee Table.....	12
Enrolls In Table.....	13
Exercise Table.....	14
Glider Upkeep Table.....	15
Holds Account Table.....	16
iLogs Exercise Table.....	17
iLogs Pilot Table.	18
Instructor Log Book Table.....	19
Item Sold Table.....	20
Lesson Milestones Table.....	21
Milestones Table.....	22
Pilot Table.	23
Product Category Table.....	24
Product Order Table.....	25
Restock Line Item Table.....	26

Return Item Table	27
Returns Table	28
Sale Line Item Table	29
Salesperson Table	30
sLogs Pilot Table.....	31
Student Log Book Table.....	32
Supplier Table	33
Trainee Table.....	34
Trainer Table	35
Training Session Table.....	36
Transaction Table.....	37
Query Design.....	38
Q01.....	38
Q02.....	39
Q03.....	40
Q04.....	41
Q05.....	42
Q06.....	42
Q07.....	43
Q08.....	44
Q09.....	45
Q10.....	46
Stored Procedures.....	47
sProc #1:.....	47
sProc #2:.....	48
Triggers.....	49
Trigger #1	49
Trigger #2 Part A	50
Trigger #2 Part B.....	50
Summary Conclusion	51

Executive Summary

Goals

The goals of the database design were to overcome the limitations of the current system and to design a system that would have longevity and scalability for the company. Additional goals included ease of use for management and employees with robust record keeping and retrieval. A final goal was to ensure appropriate inventory management and ordering.

Problems

Within the database design, the goal was to ensure proper tracking and management of employees, customers, trainees, trainers, sales, and inventory. This was achieved by designing the system using a hierachal diagram of entity relationships. Establishing a relational database in this manner allowed for a clear assessment of protocols and objectives. Trial and error testing in designing the database led us through constructing the entity-relationship model, and the normalized schema. With that as a foundation, the creation and population of the relational tables was reasonably straightforward. Our team diligently tested and redesigned when necessary, resulting in a stable relational database system.

Results

The concept of the new system was to track customer sales and returns, along with payment methods, tracking and management of inventory and ordering, as well as establishing a system for returning products to inventory or to external retailers. Employees can be tracked in the various positions, be it salesperson or trainer, it will track sales commission for the salespeople, schedules, training sessions with the trainee, tracking of log books for the student and the instructor, records and tracking of milestones and progress for the trainee, any damage to the gliders and equipment. Training sessions can be scheduled and tracked for both the trainee and trainer, and detailed results can be managed. The new system also establishes measures and protocols for tracking checking, and ordering inventory.

Recommendations

The benefit of our database design is in the longevity and scalability that the previous software products did not provide. Our design incorporates the needs of Wind Riders of the Lost River Range (WRLRR) as well as makes predictions about how the company may expand. In addition, implementing our relational database management system (RDBMS) ensures data consistency and expandability should opportunities arise in the future. Our database takes into account the needs of the business as it stands while providing exponential tracking and management capabilities.

Introduction

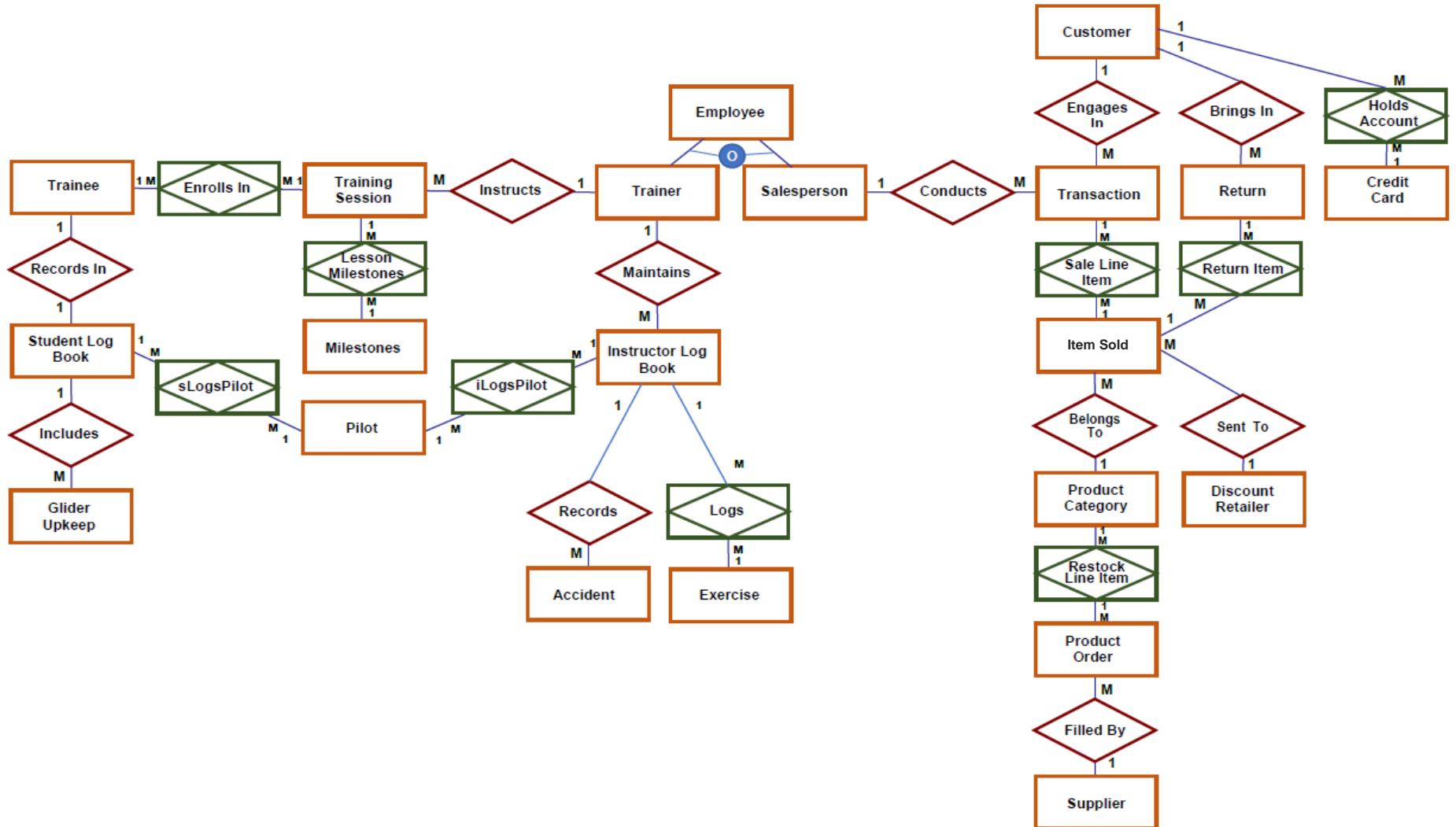
Company Background

Lost River Wind Riders started in 1990 as a cooperative giving enthusiasts access to free-flight sports. As interest increased, between 1990 and 2004, the company saw massive expansion, including the opportunity to expand into hang gliding and paragliding training. By 2015, the company was purchased, the company was renamed to differentiate from the past to WRLRR. Decisions were made to discontinue sales on specific models due to lack of interest and other regulatory issues, and maintenance and repair services were terminated for liability reasons. In addition, due to the system upgrade in 2005, better inventory controls and management became a critical issue for WRLRR.

Problem

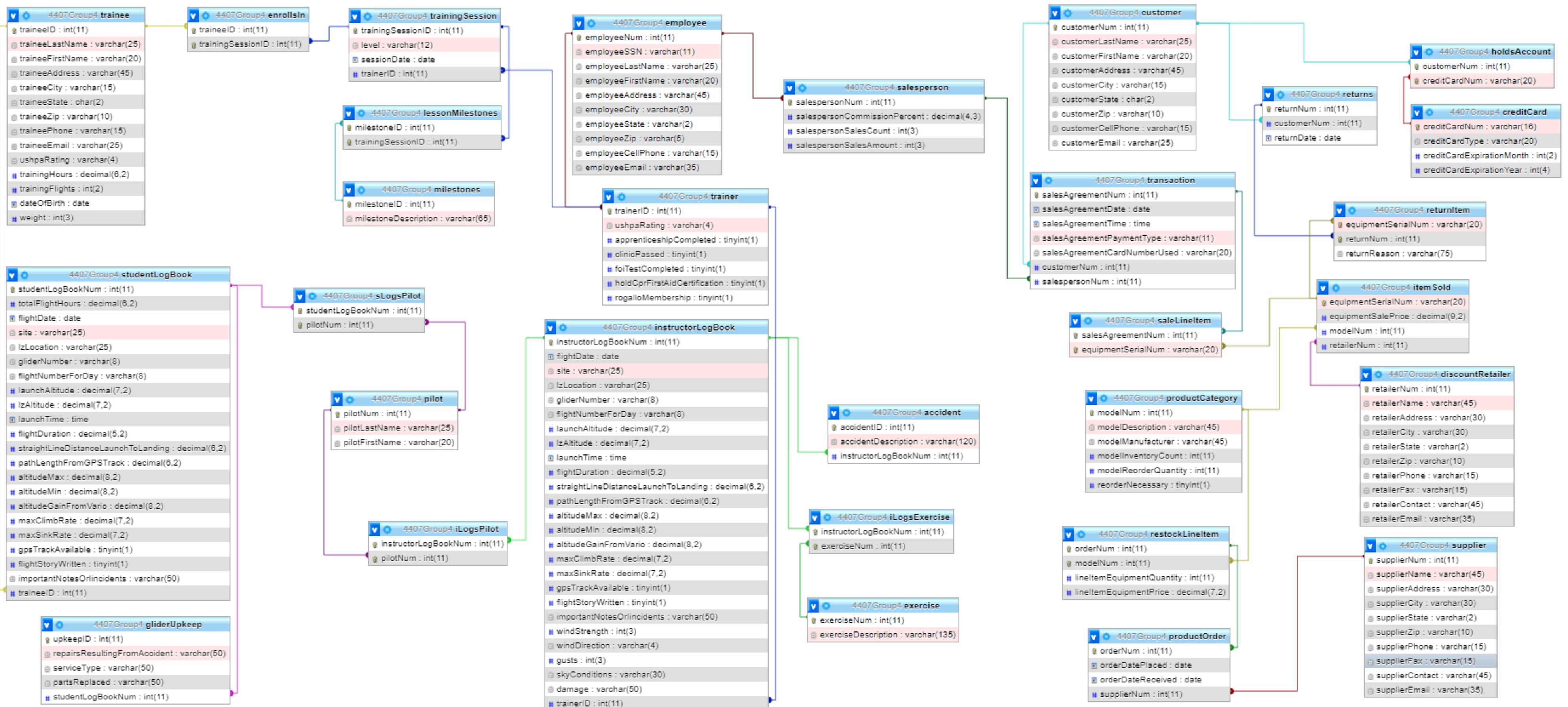
The single largest problem for WRLRR was the previous management of accounting and inventory controls using basic accounting programs and excel sheets. As the company grew, so too did their inventory management needs alongside additional back office services are required to sustain the business; and further, inventory, record keeping, and training controls are essential for the functionality of the company. Additionally, measures were needed to track customers, trainees, trainers, and commissions on services.

Entity-Relationship Diagram



- customer (customerNum, customerLastName, customerFirstName, customerAddress, customerCity, customerState, customerZip, customerHomePhone, customerCellPhone, customerWorkPhone, customerEmail, dateOfBirth, weight)
- holdsAccount (customerNum, creditCardNum)
- employee (employeeNum, employeeSSN, employeeLastName, employeeFirstName, employeeAddress, employeeCity, employeeState, employeeZip, employeeHomePhone, employeeCellPhone, employeeEmail)
- salesperson (salespersonNum, salespersonCommission%, salespersonSalesCount, salespersonSalesAmount)
- transaction (purchaseAgreementNum, purchaseAgreementDate, purchaseAgreementTime, purchaseAgreementPaymentType, purchaseAgreementCardNumberUsed, customerNum, employeeNum)
- return (returnNum, customerNum, returnDate)
- creditCard (creditCardNum, creditCardType, creditCardExpirationMonth, creditCardExpirationYear)
- saleLineItem (purchaseAgreementNum, equipmentSerialNum)
- returnLineItem (equipmentSerialNum, returnNum, returnReason)
- itemSold (equipmentSerialNum, equipmentSalePrice, modelNum, retailerNum)
- productCategory (modelNum, modelDescription, modelManufacturer, modelInventoryCount, modelReorderQuantity)
- discountRetailer (retailerNum, retailerName, retailerAddress, retailerCity, retailerState, supplierZip, retailerPhone, retailerFax, retailerContact, retailerEmail)
- restockLineItem (orderNum, modelNum, lineItemEquipmentQuantity, lineItemEquipmentPrice)
- productOrder (orderNum, orderDatePlaced, orderDateReceived, supplierNum)
- supplier (supplierNum, supplierName, supplierAddress, supplierCity, supplierState, supplierZip, supplierPhone, supplierFax, supplierContact, supplierEmail)
- trainer (trainerID, ushpaRating, apprenticeshipCompleted, clinicPassed, foitTestCompleted, holdCprFirstAidCertification, rogalloMembership)
- trainee (traineeID, traineeLastName, traineeFirstName, traineeAddress, traineeCity, traineeState, traineeZip, traineePhone, traineeEmail, ushpaRating, trainingHours, trainingFlights, dateOfBirth, weight)
- enrollsIn (traineeID, trainingSessionID)
- trainingSession (trainingSessionID, date, rating, trainerID)
- lessonMilestone (milestoneID, sessionNum)
- milestone (milestoneID, milestoneName, milestoneDescription)
- studentLogBook (studentLogBookNum, totalFlightHours, date, site, LZLocation, gliderNumber, flightNumberForDay, launchAltitude, LZAltitude, launchTime, flightDuration, straightLineDistanceLaunchToLanding, pathLengthFromGPSTrack, altitudeMax, altitudeMin, altitudeGainFromVario, maxClimbRate, maxSinkRate, gpsTrackAvailable, flightStoryWritten, importantNotesOrIncidents, traineeID)
- sLogsPilot (studentLogBookNum, pilotNum)
- pilot (pilotNum, pilotName)
- iLogsPilot (instructorLogBookNum, pilotNum)
- gliderUpkeep (upkeepID, repairsResultingFromAccident, serviceType, partsReplaced, studentLogBookNum)
- instructorLogBook (instructorLogBookID, employeeNum, date, site, LZLocation, gliderNumber, flightNumberForDay, launchAltitude, LZAltitude, launchTime, flightDuration, straightLineDistanceLaunchToLanding, pathLengthFromGPSTrack, altitudeMax, altitudeMin, altitudeGainFromVario, maxClimbRate, maxSinkRate, gpsTrackAvailable, flightStoryWritten, importantNotesOrIncidents, windStrength, windDirection, gusts, skyConditions, damage, trainerID)
- iLogsExercise (instructorLogBookNum, exerciseNum)
- exercise (exerciseNum, exerciseName, exerciseDescription)
- accident (accidentID, accidentDescription, instructorLogBookNum)
- Records In (StudentLogBookNum, 1)
- Maintains (InstructorLogBookNum, M)
- Engages In (CustomerNum, 1)
- Brings In (CustomerNum, M)
- Holds Account (CustomerNum, CreditCardNum, M)
- Conducts (SalespersonNum, TransactionNum, M)
- Belongs To (ItemSoldNum, ProductCategoryNum, 1)
- Sent To (ItemSoldNum, DiscountRetailerNum, 1)
- Logs (InstructorLogBookNum, AccidentID, M)
- Restock (ProductOrderNum, RestockLineItemNum, M)
- Filled By (ProductOrderNum, SupplierNum, 1)

Normalized Schema



Implementation

Database Table Creation

Accident Table.

```
CREATE TABLE accident (
    accidentID INT(11) NOT NULL AUTO_INCREMENT,
    accidentDescription VARCHAR(50) NOT NULL,
    instructorLogBookNum INT(11) NOT NULL,
    PRIMARY KEY (accidentID),
    FOREIGN KEY (instructorLogBookNum) REFERENCES instructorLogBook(instructorLogBookNum) ON DELETE CASCADE);
```

```
INSERT INTO `accident` (`accidentID`, `accidentDescription`, `instructorLogBookNum`) VALUES
(1000, 'Bent wing in wind gust while disengaging harness, knocked another off her feet', 40003);
```



The screenshot shows the MySQL Workbench interface with the 'accident' table selected. At the top, there's a toolbar with icons for Edit, Copy, and Delete. Below the toolbar, a row shows the current record: accidentID 1000, accidentDescription 'Bent wing in wind gust while disengaging harness, knocked another off her feet', and instructorLogBookNum 40003.

Table structure:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	accidentID	int(11)		No	None			AUTO_INCREMENT	Edit Rename Drop More
2	accidentDescription	varchar(120)	latin1_swedish_ci	No	None				Edit Rename Drop More
3	instructorLogBookNum	int(11)		No	None				Edit Rename Drop More

Indexes:

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	accidentID	0	A	No	
Edit Rename Drop	instructorLogBookNum	BTREE	No	No	instructorLogBookNum	0	A	No	

Credit Card Table.

```
CREATE TABLE creditCard (
    creditCardNum VARCHAR(20) NOT NULL,
    creditCardType VARCHAR(20) NOT NULL,
    creditCardExpirationMonth INT(2) NOT NULL,
    creditCardExpirationYear INT(4) NOT NULL,
    PRIMARY KEY (creditCardNum));
```

```
INSERT INTO `creditCard` (`creditCardNum`, `creditCardType`, `creditCardExpirationMonth`, `creditCardExpirationYear`) VALUES
('366667778888999', 'Discover', '11', '2022'),
('4120666734254132', 'Visa', '2', '2023'),
('4555554444333221', 'MasterCard', '7', '2024'),
('5140123408221452', 'Amex', '6', '2023'),
('8564858986567844', 'Visa', '11', '2024'),
('5268080043376894', 'MasterCard', '1', '2022');
```

			creditCardNum	creditCardType	creditCardExpirationMonth	creditCardExpirationYear	
<input type="checkbox"/>	Edit	Copy	Delete	366667778888999	Discover	11	2022
<input type="checkbox"/>	Edit	Copy	Delete	4120666734254132	Visa	2	2023
<input type="checkbox"/>	Edit	Copy	Delete	4555554444333221	MasterCard	7	2024
<input type="checkbox"/>	Edit	Copy	Delete	5140123408221452	Amex	6	2023
<input type="checkbox"/>	Edit	Copy	Delete	5268080043376894	MasterCard	1	2022
<input type="checkbox"/>	Edit	Copy	Delete	8564858986567844	Visa	11	2024

Table structure		Relation view										
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action			
<input type="checkbox"/>	1 creditCardNum	varchar(16)	latin1_swedish_ci		No	None			Change	Drop	More	
<input type="checkbox"/>	2 creditCardType	varchar(20)	latin1_swedish_ci		No	None			Change	Drop	More	
<input type="checkbox"/>	3 creditCardExpirationMonth	int(2)			No	None			Change	Drop	More	
<input type="checkbox"/>	4 creditCardExpirationYear	int(4)			No	None			Change	Drop	More	

Indexes												
Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment			
Edit	Rename	Drop	PRIMARY	BTREE	Yes	No	creditCardNum	6	A	No		

Customer Table.

```
CREATE TABLE customer (
    customerNum INT(11) NOT NULL AUTO_INCREMENT,
    customerLastName VARCHAR(25) NOT NULL,
    customerFirstName VARCHAR(20) NOT NULL,
    customerAddress VARCHAR(45) NOT NULL,
    customerCity VARCHAR(15) NOT NULL,
    customerState CHAR(2) NOT NULL,
    customerZip VARCHAR(10) NOT NULL,
    customerCellPhone VARCHAR(15) NOT NULL,
    customerEmail VARCHAR(25) NOT NULL,
    PRIMARY KEY (customerNum));
```

```
INSERT INTO `customer` (`customerNum`, `customerLastName`, `customerFirstName`, `customerAddress`, `customerCity`, `customerState`, `customerZip`, `customerCellPhone`, `customerEmail`) VALUES
(1, 'Jones', 'Walter', '428 S. Park St.', 'Walla Walla', 'WA', '99362', '(509) 529-3428', 'walljam@gmail.com'),
(2, 'Smith', 'Bert', '144 Jolly Ln.', 'Idaho Falls', 'ID', '83401', '(208) 455-6789', 'bert@hcom.com'),
(3, 'Mall', 'Janice', '997 SE Admiral Pl.', 'Walla Walla', 'WA', '99362', '(509) 529-7898', 'mallhound@gmail.com'),
(4, 'Dane', 'Jim \\"Great\\"', '996 Vista Del Sol Ct.', 'Mesquite', 'NV', '89027', '(702) 346-7200', 'greatdane@hotmail.com'),
(5, 'James', 'Edna', '208 W Aspen Valley Dr.', 'Panguitch', 'UT', '84759', '(801) 391-2678', 'auntedna@sparklight.net'),
(6, 'Etheridge', 'Walt', '1500 WestWind Ct.', 'Jackson', 'WY', '83002', '(307) 451-7634', 'weth@gmail.com'),
(7, 'Haskell', 'Edward', '1236 Mockingbird Ln.', 'Mayfield', 'NY', '12117', '(518) 661-2212', 'slickEddie@leavelt.com'),
(8, 'Wallace', 'Ed', '414 Elk Dr.', 'Panguitch', 'UT', '84759', '(801) 391-7425', 'edo@gmail.com');
```

Customer Table Data										
	customerNum	customerLastName	customerFirstName	customerAddress	customerCity	customerState	customerZip	customerCellPhone	customerEmail	
<input type="checkbox"/>	Edit	Copy	Delete	1	Jones	Walter	428 S. Park St.	Walla Walla	WA	99362 (509) 529-3428 walljam@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	2	Smith	Bert	144 Jolly Ln.	Idaho Falls	ID	83401 (208) 455-6789 bert@hcom.com
<input type="checkbox"/>	Edit	Copy	Delete	3	Mall	Janice	997 SE Admiral Pl.	Walla Walla	WA	99362 (509) 529-7898 mallhound@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	4	Dane	Jim "Great"	996 Vista Del Sol Ct.	Mesquite	NV	89027 (702) 346-7200 greatdane@hotmail.com
<input type="checkbox"/>	Edit	Copy	Delete	5	James	Edna	208 W Aspen Valley Dr.	Panguitch	UT	84759 (801) 391-2678 auntedna@sparklight.net
<input type="checkbox"/>	Edit	Copy	Delete	6	Etheridge	Walt	1500 WestWind Ct.	Jackson	WY	83002 (307) 451-7634 weth@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	7	Haskell	Edward	1236 Mockingbird Ln.	Mayfield	NY	12117 (518) 661-2212 slickEddie@leavelt.com
<input type="checkbox"/>	Edit	Copy	Delete	8	Wallace	Ed	414 Elk Dr.	Panguitch	UT	84759 (801) 391-7425 edo@gmail.com

Customer Table Structure											
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
<input type="checkbox"/>	1 customerNum	int(11)			No	None		AUTO_INCREMENT	<input type="button" value="Change"/>	<input type="button" value="Drop"/>	<input type="button" value="More"/>
<input type="checkbox"/>	2 customerLastName	varchar(25)	latin1_swedish_ci		No	None			<input type="button" value="Change"/>	<input type="button" value="Drop"/>	<input type="button" value="More"/>
<input type="checkbox"/>	3 customerFirstName	varchar(20)	latin1_swedish_ci		No	None			<input type="button" value="Change"/>	<input type="button" value="Drop"/>	<input type="button" value="More"/>
<input type="checkbox"/>	4 customerAddress	varchar(45)	latin1_swedish_ci		No	None			<input type="button" value="Change"/>	<input type="button" value="Drop"/>	<input type="button" value="More"/>
<input type="checkbox"/>	5 customerCity	varchar(15)	latin1_swedish_ci		No	None			<input type="button" value="Change"/>	<input type="button" value="Drop"/>	<input type="button" value="More"/>
<input type="checkbox"/>	6 customerState	char(2)	latin1_swedish_ci		No	None			<input type="button" value="Change"/>	<input type="button" value="Drop"/>	<input type="button" value="More"/>
<input type="checkbox"/>	7 customerZip	varchar(10)	latin1_swedish_ci		No	None			<input type="button" value="Change"/>	<input type="button" value="Drop"/>	<input type="button" value="More"/>
<input type="checkbox"/>	8 customerCellPhone	varchar(15)	latin1_swedish_ci		No	None			<input type="button" value="Change"/>	<input type="button" value="Drop"/>	<input type="button" value="More"/>
<input type="checkbox"/>	9 customerEmail	varchar(25)	latin1_swedish_ci		No	None			<input type="button" value="Change"/>	<input type="button" value="Drop"/>	<input type="button" value="More"/>

Customer Table Indexes										
Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment	
<input type="button" value="Edit"/>	<input type="button" value="Rename"/>	<input type="button" value="Drop"/>	PRIMARY	BTREE	Yes	No	customerNum	8	A	No

Discount Retailer Table.

```
CREATE TABLE discountRetailer (
    retailerNum INT(11) NOT NULL AUTO_INCREMENT,
    retailerName VARCHAR(45) NOT NULL,
    retailerAddress VARCHAR(30) NOT NULL,
    retailerCity VARCHAR(30) NOT NULL,
    retailerState VARCHAR(2) NOT NULL,
    retailerZip VARCHAR(10) NOT NULL,
    retailerPhone VARCHAR(15) NOT NULL,
    retailerFax VARCHAR(15) NOT NULL,
    retailerContact VARCHAR(45) NOT NULL,
    retailerEmail VARCHAR(35) NOT NULL,
    PRIMARY KEY (retailerNum));
```

```
INSERT INTO `discountRetailer` (`retailerNum`, `retailerName`, `retailerAddress`, `retailerCity`, `retailerState`, `retailerZip`, `retailerPhone`, `retailerFax`, `retailerContact`, `retailerEmail`) VALUES
(1, 'Moyes Resellers', '614 W. Ocean Blvd', 'Long Beach', 'CA', '90802-0000', '(818) 367-2430', '(818) 367-2441', 'Janet Moyes', 'janet@moyesonline.com'),
(2, 'Wing Seller', '121100 N. East, N Frontage Rd.', 'Draper', 'UY', '82020-0000', '801-382-2492', '801-382-2477', 'Ed O\'Keefe', 'edo@wingdeller.com'),
(3, 'Griffin Wings', '1500 West Harbor Ave.', 'Orange', 'CA', '92865-1001', '(714) 480-4295', '(714) 480-9870', 'Hermione Winn', 'winner@grieffinwings.com');
```

	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	1	retailerName	retailerAddress	retailerCity	retailerState	retailerZip	retailerPhone	retailerFax	retailerContact	retailerEmail
<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	2	Wing Seller	121100 N. East, N Frontage Rd.	Draper	UY	82020-0000	801-382-2492	801-382-2477	Ed O'Keefe	edo@wingdeller.com
<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	3	Griffin Wings	1500 West Harbor Ave.	Orange	CA	92865-1001	(714) 480-4295	(714) 480-9870	Hermione Winn	winner@grieffinwings.com

<input type="checkbox"/> Table structure		<input type="checkbox"/> Relation view											
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action				
<input type="checkbox"/>	1	retailerNum 	int(11)		No	None		AUTO_INCREMENT	<input type="checkbox"/> Change	<input type="checkbox"/> Drop	<input type="checkbox"/> More		
<input type="checkbox"/>	2	retailerName	varchar(45)	latin1_swedish_ci	No	None			<input type="checkbox"/> Change	<input type="checkbox"/> Drop	<input type="checkbox"/> More		
<input type="checkbox"/>	3	retailerAddress	varchar(30)	latin1_swedish_ci	No	None			<input type="checkbox"/> Change	<input type="checkbox"/> Drop	<input type="checkbox"/> More		
<input type="checkbox"/>	4	retailerCity	varchar(30)	latin1_swedish_ci	No	None			<input type="checkbox"/> Change	<input type="checkbox"/> Drop	<input type="checkbox"/> More		
<input type="checkbox"/>	5	retailerState	varchar(2)	latin1_swedish_ci	No	None			<input type="checkbox"/> Change	<input type="checkbox"/> Drop	<input type="checkbox"/> More		
<input type="checkbox"/>	6	retailerZip	varchar(10)	latin1_swedish_ci	No	None			<input type="checkbox"/> Change	<input type="checkbox"/> Drop	<input type="checkbox"/> More		
<input type="checkbox"/>	7	retailerPhone	varchar(15)	latin1_swedish_ci	No	None			<input type="checkbox"/> Change	<input type="checkbox"/> Drop	<input type="checkbox"/> More		
<input type="checkbox"/>	8	retailerFax	varchar(15)	latin1_swedish_ci	Yes	NULL			<input type="checkbox"/> Change	<input type="checkbox"/> Drop	<input type="checkbox"/> More		
<input type="checkbox"/>	9	retailerContact	varchar(45)	latin1_swedish_ci	No	None			<input type="checkbox"/> Change	<input type="checkbox"/> Drop	<input type="checkbox"/> More		
<input type="checkbox"/>	10	retailerEmail	varchar(35)	latin1_swedish_ci	No	None			<input type="checkbox"/> Change	<input type="checkbox"/> Drop	<input type="checkbox"/> More		

Indexes 													
Action	<input type="checkbox"/> Edit	<input type="checkbox"/> Rename	<input type="checkbox"/> Drop	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment	
<input type="checkbox"/>		<input type="checkbox"/> Drop	PRIMARY	BTREE	Yes	No		retailerNum	3	A	No		

Employee Table.

```
CREATE TABLE employee (
    employeeNum INT(11) NOT NULL AUTO_INCREMENT,
    employeeSSN VARCHAR(11) NOT NULL,
    employeeLastName VARCHAR(25) NOT NULL,
    employeeFirstName VARCHAR(20) NOT NULL,
    employeeAddress VARCHAR(45) NOT NULL,
    employeeCity VARCHAR(30) NOT NULL,
    employeeState VARCHAR(2) NOT NULL,
    employeeZip VARCHAR(5) NOT NULL,
    employeeCellPhone VARCHAR(15) NOT NULL,
    employeeEmail VARCHAR(35) NOT NULL,
    PRIMARY KEY (employeeNum);
```

```
INSERT INTO `employee` (`employeeNum`, `employeeSSN`, `employeeLastName`, `employeeFirstName`, `employeeAddress`, `employeeCity`, `employeeState`, `employeeZip`, `employeeCellPhone`, `employeeEmail`) VALUES
(1, '514-92-0431', 'Hall', 'Meredith', '2700 W.', 'Howe', 'ID', '83244', '(208) 317-2291', 'hall_meredith@wrlrr.com'),
(2, '614-21-0442', 'Davis', 'Art', '109 S. Hickock St.', 'Mackay', 'ID', '83251', '(208) 317-2291', 'davis_art@wrlrr.com'),
(3, '212-77-7890', 'Teel', 'Lee', '811 W. College St.', 'Mackay', 'ID', '83251', '(208) 222-5555', 'teel_lee@wrlrr.com'),
(4, '554-92-3410', 'Fries', 'Jim', '3505 Little Lost River Hwy', 'Howe', 'ID', '83244', '(208) 555-2121', 'jim_fries@wrlrr.com'),
(5, '555-55-6532', 'Anderson', 'Chris', '225 W. Grand Ave.', 'Arco', 'ID', '83213', '(208) 317-6541', 'birdman@wrlrr.com'),
(6, '555-04-2005', 'Long', 'Tom', '5600 N 2700 W.', 'Howe', 'ID', '83244', '(208) 233-6721', 'longTom@wrlrr.com');
```

		employeeNum	employeeSSN	employeeLastName	employeeFirstName	employeeAddress	employeeCity	employeeState	employeeZip	employeeCellPhone	employeeEmail
	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	
<input type="checkbox"/>	1	employeeNum	int(11)			No	None		AUTO_INCREMENT		
<input type="checkbox"/>	2	employeeSSN	varchar(11)	latin1_swedish_ci		No	None				
<input type="checkbox"/>	3	employeeLastName	varchar(25)	latin1_swedish_ci		No	None				
<input type="checkbox"/>	4	employeeFirstName	varchar(20)	latin1_swedish_ci		No	None				
<input type="checkbox"/>	5	employeeAddress	varchar(45)	latin1_swedish_ci		No	None				
<input type="checkbox"/>	6	employeeCity	varchar(30)	latin1_swedish_ci		No	None				
<input type="checkbox"/>	7	employeeState	varchar(2)	latin1_swedish_ci		No	None				
<input type="checkbox"/>	8	employeeZip	varchar(5)	latin1_swedish_ci		No	None				
<input type="checkbox"/>	9	employeeCellPhone	varchar(15)	latin1_swedish_ci		No	None				
<input type="checkbox"/>	10	employeeEmail	varchar(35)	latin1_swedish_ci		No	None				

Indexes											
Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment		
	Rename		PRIMARY	BTREE	Yes	No	employeeNum	6	A		

Enrolls In Table.

```
CREATE TABLE enrollsIn (
    traineeID INT(11) NOT NULL,
    trainingSessionID INT(11) NOT NULL,
    PRIMARY KEY (traineeID, trainingSessionID));
```

```
INSERT INTO `enrollsIn` (`traineeID`, `trainingSessionID`) VALUES
(10001, 5),
(10001, 6),
(10002, 1),
(10002, 2),
(10002, 3),
(10002, 4);
```

				traineeID	trainingSessionID
<input type="checkbox"/>				10001	5
<input type="checkbox"/>				10001	6
<input type="checkbox"/>				10002	1
<input type="checkbox"/>				10002	2
<input type="checkbox"/>				10002	3
<input type="checkbox"/>				10002	4

#		Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
<input type="checkbox"/>		1 traineeID	int(11)			No	None					
<input type="checkbox"/>		2 trainingSessionID	int(11)			No	None					
Indexes												
Action		Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment		
			BTREE	Yes	No	traineeID trainingSessionID	6 6	A A	No No			

Exercise Table.

```
CREATE TABLE exercise (
    exerciseNum INT(11) NOT NULL AUTO_INCREMENT,
    exerciseDescription VARCHAR(35) NOT NULL,
    PRIMARY KEY (exerciseNum);
```

```
INSERT INTO `exercise` (`exerciseNum`, `exerciseDescription`) VALUES
(10001, 'Set-up and perform a pre-flight check on the glider'),
(10002, 'Unassisted take-off on a training hill'),
(10003, 'Safe, straight flight controlling airspeed and making minor corrections'),
(10004, 'Controlled stand-up landing or wheel landing if via aerotow'),
(10005, 'Land three times in a row within 100 feet of a target'),
(10006, 'Complete linked 90° turns. These turns must alternate from left to right or vice versa and demonstrate smooth control'),
(10007, 'Demonstrate fast and slow flight'),
(10008, 'Demonstrate steep and gentle 180° linked turns along a predetermined track'),
(10009, 'Land three times in a row within 50 feet of a target'),
(10010, 'Complete at least 10 flights with 75 feet or greater ground clearance'),
(10011, 'Demonstrate speed control in turns and various wind conditions'),
(10012, '60-minute flight'),
(10013, 'Soar in a thermal lift only for at least 30 minutes in one flight'),
(10014, 'Demonstrate figure eights around reselected pylons across the wind, with the course flown smoothly with equal radius turns'),
(10015, 'Land three times in a row within 25 feet of a spot after flights of at least one minute and at least 200 feet in altitude'),
(10016, 'Demonstrate stall and recovery');
```

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	exerciseNum	int(11)			No	None	AUTO_INCREMENT		
2	exerciseDescription	varchar(135)	latin1_swedish_ci		No	None			

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
	PRIMARY	BTREE	Yes	No	exerciseNum	16	A	No	

Glider Upkeep Table.

```
CREATE TABLE gliderUpkeep (
    upkeepID INT(11) NOT NULL AUTO_INCREMENT,
    repairsResultingFromAccident VARCHAR(50) NOT NULL,
    serviceType VARCHAR(50) NOT NULL,
    pilotFirstName VARCHAR(20) NOT NULL,
    partsReplaced VARCHAR(50) NOT NULL,
    studentLogBookNum INT(11) NOT NULL,
    PRIMARY KEY (upkeepID),
    FOREIGN KEY (studentLogBookNum) REFERENCES studentLogBook(studentLogBookNum) ON DELETE CASCADE);
```

```
INSERT INTO `gliderUpkeep`(`upkeepID`, `repairsResultingFromAccident`, `serviceType`, `partsReplaced`, `studentLogBookNum`) VALUES
(10001, 'Bent wing', 'Straighten wing', 'None', 110013);
```

The screenshot shows the MySQL Workbench interface with the 'Table structure' tab selected for the 'gliderUpkeep' table. The table has five columns: 'upkeepID', 'repairsResultingFromAccident', 'serviceType', 'partsReplaced', and 'studentLogBookNum'. A single row is inserted with values: upkeepID=10001, repairsResultingFromAccident='Bent wing', serviceType='Straighten wing', partsReplaced='None', and studentLogBookNum=110013.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	upkeepID	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	repairsResultingFromAccident	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
3	serviceType	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
4	partsReplaced	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
5	studentLogBookNum	int(11)			No	None			Change Drop More

Action	Edit	Rename	Drop	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
				PRIMARY	BTREE	Yes	No	upkeepID	0	A	No	
				studentLogBookNum	BTREE	No	No	studentLogBookNum	0	A	No	

Holds Account Table.

```
CREATE TABLE holdsAccount (
    customerNum INT(11) NOT NULL,
    creditCardNum VARCHAR(20) NOT NULL,
    PRIMARY KEY (customerNum, creditCardNum));
```

```
INSERT INTO `holdsAccount` (`customerNum`, `creditCardNum`) VALUES
(1, '4120666734254132'),
(2, '5140123408221452'),
(3, '8564858986567844'),
(4, '4555554444333221'),
(5, '3666677778888999'),
(1, '5268080043376894');
```

				customerNum	creditCardNum
<input type="checkbox"/>				1	4120666734254132
<input type="checkbox"/>				1	5268080043376894
<input type="checkbox"/>				2	5140123408221452
<input type="checkbox"/>				3	8564858986567844
<input type="checkbox"/>				4	4555554444333221
<input type="checkbox"/>				5	3666677778888999

Table structure		Relation view											
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	Change	Delete	More	
<input type="checkbox"/>	1 customerNum	int(11)		No	None								
<input type="checkbox"/>	2 creditCardNum	varchar(20)	latin1_swedish_ci	No	None								
Indexes													
Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment				
	Rename		PRIMARY	BTREE	Yes	No	customerNum	6	A			No	
							creditCardNum	6	A			No	

iLogs Exercise Table.

```
CREATE TABLE iLogsExercise (
    instructorLogBookNum INT(11) NOT NULL,
    exerciseNum INT(11) NOT NULL,
    PRIMARY KEY (instructorLogBookNum, exerciseNum));
```

```
INSERT INTO `iLogsExercise` (`instructorLogBookNum`, `exerciseNum`) VALUES
(40001, 10006),
(40002, 10007),
(40003, 10003);
```

The screenshot shows the MySQL Workbench interface with the 'Table structure' tab selected for the 'iLogsExercise' table. The table has two columns: 'instructorLogBookNum' and 'exerciseNum'. The data view shows three rows with values (40001, 10006), (40002, 10007), and (40003, 10003). The structure view shows the primary key constraint and the column definitions. The indexes view shows a primary index on 'instructorLogBookNum' and 'exerciseNum'.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	instructorLogBookNum	int(11)			No	None			Edit Change Drop More
2	exerciseNum	int(11)			No	None			Edit Change Drop More

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit	PRIMARY	BTREE	Yes	No	instructorLogBookNum	3	A	No	
					exerciseNum	3	A	No	

iLogs Pilot Table.

```
CREATE TABLE iLogsPilot (
    instructorLogBookNum INT(11) NOT NULL,
    pilotNum INT(11) NOT NULL,
    PRIMARY KEY (instructorLogBookNum, pilotNum));
```

```
INSERT INTO `iLogsPilot` (`instructorLogBookNum`, `pilotNum`) VALUES
(40002, 14785),
(40003, 17564);
```

The screenshot shows the MySQL Workbench interface with the 'iLogsPilot' table selected. The top section displays the table's data in a grid:

				instructorLogBookNum	pilotNum
<input type="checkbox"/>	Edit	Copy	Delete	40002	14785
<input type="checkbox"/>	Edit	Copy	Delete	40003	17564

The bottom section shows the 'Table structure' tab, displaying the columns and their properties:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	instructorLogBookNum	int(11)			No	None		Change	Drop More
2	pilotNum	int(11)			No	None		Change	Drop More

Below the table structure, the 'Indexes' section is shown, listing the primary key:

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit	PRIMARY	BTREE	Yes	No	instructorLogBookNum	2	A	No	
					pilotNum	2	A	No	

Instructor Log Book Table.

```
CREATE TABLE instructorLogBook (
instructorLogBookNum INT(11) NOT NULL AUTO_INCREMENT,
flightDate DATE NOT NULL,
site VARCHAR(25) NOT NULL,
lzLocation VARCHAR(25) NOT NULL,
gliderNumber VARCHAR(8) NOT NULL,
flightNumberForDay VARCHAR(8) NOT NULL,
launchAltitude DECIMAL(6,2) NOT NULL,
lzAltitude VARCHAR(8) NOT NULL,
launchTime TIME NOT NULL,
flightDuration DECIMAL(3,2) NOT NULL,
straightLineDistanceLaunchToLanding DECIMAL(5,2) NOT NULL,
pathLengthFromGPSTrack DECIMAL(5,2) NOT NULL,
altitudeMax DECIMAL(6,2) NOT NULL,
altitudeMin DECIMAL(6,2) NOT NULL,
altitudeGainFromVario DECIMAL(6,2) NOT NULL,
maxClimbRate DECIMAL(6,2) NOT NULL,
maxSinkRate DECIMAL(6,2) NOT NULL,
gpsTrackAvailable BOOLEAN NOT NULL NOT NULL,
importantNotesOrIncidents BOOLEAN NOT NULL,
windStrength INT(3) NOT NULL,
windDirection VARCHAR(4) NOT NULL,
gusts INT(3) NOT NULL,
skyConditions VARCHAR(20) NOT NULL,
damage VARCHAR(25) NOT NULL,
trainerID INT(11) NOT NULL,
PRIMARY KEY (instructorLogBookNum),
FOREIGN KEY (trainerID) REFERENCES trainer(trainerID) ON DELETE CASCADE)
```

```
INSERT INTO `instructorLogBook` (`instructorLogBookNum`, `flightDate`, `site`, `lzLocation`, `gliderNumber`, `flightNumberForDay`, `launchAltitude`, `lzAltitude`, `launchTime`, `flightDuration`, `straightLineDistanceLaunchToLanding`, `pathLengthFromGPSTrack`, `altitudeMax`, `altitudeMin`, `altitudeGainFromVario`, `maxClimbRate`, `maxSinkRate`, `gpsTrackAvailable`, `flightStoryWritten`, `importantNotesOrIncidents`, `windStrength`, `windDirection`, `gusts`, `skyConditions`, `damage`, `trainerID`) VALUES
```

(40001, '2020-08-14', 'Big Southern Butte', 'Site 2', '24B98', '224', 5200.00, 4150.00, '13:42:00', 1.50, 10.4, 12.8, 12000.50, 9400.32, 12896.42, 24.2, 49.5, TRUE, TRUE, 'Strong gusts above 10000', 35, 'NNE', 42, 'Clear', 'None', 6),

(40002, '2020-08-02', 'Big Southern Butte', 'Site 2', '24B75', '224', 5200.00, 4150.00, '11:15:00', 1.80, 10.2, 14.7, 13400.00, 9100.32, 13214.80, 31.6, 54.8, TRUE, TRUE, 'Paragliders interfered with flight path', 29, 'NE', 34, 'Partly cloudy', 'None', 6),

(40003, 2020-08-02, Big Southern Butte, Site 1, 266A9, 224, 5200.00, 4150.00, 12.30.00, 0.75, 10.4, 15.2, 10000.00, 8800.74, 12647.68, 20.7, 34.7, TRUE, TRUE, 'Difficulty turning into strong winds', 33, 'NNE', 36, 'Overcast, spotty sun', 'Wing damage after landing', 5);

	Edit	Copy	Delete	Flight Log Book Number	Flight Date	Site	Iz Location	Glider Number	Flight Number for Day	Launch Altitude	Iz Altitude	Launch Time	Flight Duration	Straight Line Distance Launch to Landing	Path Length from GPS Track
<input type="checkbox"/>	Edit	Copy	Delete	40001	2020-08-14	Big Southern Butte	Site 2	24B98	224	5200.00	4150.00	13:42:00	1.50	10.40	12.80
<input type="checkbox"/>	Edit	Copy	Delete	40002	2020-08-02	Big Southern Butte	Site 2	24B75	224	5200.00	4150.00	11:15:00	1.80	10.20	14.70
<input type="checkbox"/>	Edit	Copy	Delete	40003	2020-08-02	Big Southern Butte	Site 1	266A9	224	5200.00	4150.00	12:30:00	0.75	10.40	13.20
altitudeMax	altitudeMin	altitudeGainFromVario	maxClimbRate	maxSinkRate	gpsTrackAvailable	flightStoryWritten	importantNotesOrIncidents	windStrength	windDirection	gusts	skyConditions	damage	trainerID		
12000.50	9400.32	12896.42	24.20	49.50	1	1	Strong gusts above 10000	35	NNE	42	Clear	None	6		
13400.00	9100.32	13214.80	31.60	54.80	1	1	Paragliders interfered with flight path	29	NE	34	Partly cloudy	None	6		
10000.00	8800.74	12647.68	20.70	34.70	1	1	Difficulty turning into strong winds	33	NNE	36	Overcast, spotty sun	Wing damage after landing	5		

#		Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	instructorLogBookNum 	int(11)			No	None		AUTO_INCREMENT	 Change  Drop  More
<input type="checkbox"/>	2	flightDate	date			No	None			 Change  Drop  More
<input type="checkbox"/>	3	site	varchar(25)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	4	lzLocation	varchar(25)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	5	gliderNumber	varchar(8)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	6	flightNumberForDay	varchar(8)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	7	launchAltitude	decimal(7,2)			No	None			 Change  Drop  More
<input type="checkbox"/>	8	lzAltitude	decimal(7,2)			No	None			 Change  Drop  More
<input type="checkbox"/>	9	launchTime	time			No	None			 Change  Drop  More
<input type="checkbox"/>	10	flightDuration	decimal(5,2)			No	None			 Change  Drop  More
<input type="checkbox"/>	11	straightLineDistanceLaunchToLanding	decimal(6,2)			No	None			 Change  Drop  More
<input type="checkbox"/>	12	pathLengthFromGPSTrack	decimal(6,2)			No	None			 Change  Drop  More
<input type="checkbox"/>	13	altitudeMax	decimal(8,2)			No	None			 Change  Drop  More
<input type="checkbox"/>	14	altitudeMin	decimal(8,2)			No	None			 Change  Drop  More
<input type="checkbox"/>	15	altitudeGainFromVario	decimal(8,2)			No	None			 Change  Drop  More
<input type="checkbox"/>	16	maxClimbRate	decimal(7,2)			No	None			 Change  Drop  More
<input type="checkbox"/>	17	maxSinkRate	decimal(7,2)			No	None			 Change  Drop  More
<input type="checkbox"/>	18	gpsTrackAvailable	tinyint(1)			No	None			 Change  Drop  More
<input type="checkbox"/>	19	flightStoryWritten	tinyint(1)			No	None			 Change  Drop  More
<input type="checkbox"/>	20	importantNotesOrIncidents	varchar(50)	latin1_swedish_ci		Yes	NULL			 Change  Drop  More
<input type="checkbox"/>	21	windStrength	int(3)			No	None			 Change  Drop  More
<input type="checkbox"/>	22	windDirection	varchar(4)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	23	gusts	int(3)			No	None			 Change  Drop  More
<input type="checkbox"/>	24	skyConditions	varchar(30)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	25	damage	varchar(50)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	26	trainerID 	int(11)			No	None			 Change  Drop  More

Item Sold Table.

```
CREATE TABLE itemSold (
    equipmentSerialNum VARCHAR(20) NOT NULL,
    equipmentSalePrice DECIMAL(9,2) NOT NULL,
    equipmentType VARCHAR(20) NOT NULL,
    equipmentDescription VARCHAR(20) NOT NULL,
    modelNum INT(11) NOT NULL,
    retailerNum INT(11) NOT NULL,
    PRIMARY KEY (equipmentSerialNum),
    FOREIGN KEY (modelNum) REFERENCES productCategory(modelNum) ON DELETE CASCADE,
    FOREIGN KEY (retailerNum) REFERENCES discountRetailer(retailerNum) ON DELETE CASCADE);
```

```
INSERT INTO `itemPurchased` (`equipmentSerialNum`, `equipmentPurchasePrice`, `equipmentSalePrice`, `equipmentPurchaseDate`, `modelNum`, `retailerNum`)
VALUES
('563421', 2400.00, 2640.00, '2018-04-01', 5, NULL),
('123456', 4300.00, 4730.00, '2018-04-10', 8, NULL),
('132457', 4200.00, 4620.00, '2018-04-10', 9, 1),
('111111', 5000.00, 5500.00, '2018-04-21', 4, NULL),
('129876', 5200.00, 5720.00, '2019-01-06', 3, NULL),
('378927', 3700.00, 4070.00, '2019-02-04', 1, NULL),
('563422', 3700.00, 4070.00, '2019-04-01', 1, NULL),
('677811', 4800.00, 5280.00, '2019-04-22', 2, NULL),
('017767', 4700.00, 5170.00, '2019-04-22', 6, 2),
('639831', 5400.00, 5940.00, '2020-02-15', 7, NULL),
('266453', 5400.00, 5940.00, '2020-02-15', 7, 1),
('288545', 2600.00, 2860.00, '2020-03-04', 11, NULL),
('333657', 2600.00, 2860.00, '2020-03-04', 11, NULL),
('194311', 5000.00, 5500.00, '2020-04-21', 10, NULL);
```

	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	equipmentSerialNum	equipmentSalePrice	modelNum	retailerNum
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	017767	5170.00	6	2
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	111111	5500.00	4	NULL
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	123456	4730.00	8	NULL
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	129876	5720.00	3	NULL
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	132457	4620.00	9	1
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	194311	5500.00	10	NULL
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	266453	5940.00	7	1
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	288545	2860.00	11	NULL
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	333657	2860.00	11	NULL
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	378927	4070.00	1	NULL
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	402984	4070.00	1	NULL
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	563421	2640.00	5	NULL
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	563422	4070.00	1	NULL
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	639831	5940.00	7	NULL
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	677811	5280.00	2	NULL

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	equipmentSerialNum	varchar(20)	latin1_swedish_ci		No	None			
2	equipmentSalePrice	decimal(9,2)			No	None			
3	modelNum	int(11)			No	None			
4	retailerNum	int(11)			Yes	NULL			

Indexes

Action				Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null
				PRIMARY	BTREE	Yes	No	equipmentSerialNum	14	A	No
				modelNum	BTREE	No	No	modelNum	14	A	No
				retailerNum	BTREE	No	No	retailerNum	7	A	Yes

Lesson Milestones Table.

```
CREATE TABLE lessonMilestones (
    milestoneID INT(11) NOT NULL ,
    trainingSessionID INT(11) NOT NULL,
    PRIMARY KEY (milestoneID, trainingSessionID));
```

```
INSERT INTO `lessonMilestones` (`milestoneID`, `trainingSessionID`) VALUES
(1, 1), (2, 1), (3, 1), (4, 1), (5, 1), (1, 2), (2, 2), (3, 2), (4, 2), (5, 2), (6, 3), (7, 3), (8, 3), (9, 3), (10, 3), (6, 4), (7, 4),
(8, 4), (9, 4), (10, 5), (11, 5), (12, 5), (13, 5), (14, 5), (15, 5), (10, 6), (11, 6), (12, 6), (13, 6), (14, 6), (15, 6);
```

The screenshot shows the MySQL Workbench interface with the 'lessonMilestones' table selected. At the top, there's a grid view of the table data. Below it, the 'Table structure' tab is active, displaying the schema with two columns: 'milestoneID' and 'trainingSessionID'. The 'Indexes' tab at the bottom shows a primary key index named 'PRIMARY'.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	milestoneID	int(11)			No	None			Change Drop More
2	trainingSessionID	int(11)			No	None			Change Drop More

Below the table structure, there are several navigation and search buttons: Print, Propose table structure, Move columns, Normalize, Add, Go, Check all, With selected:, Browse, Change, Drop, Primary, Unique, Index, Spatial, Fulltext.

At the bottom, there's a search bar for 'Add' with '1' entered, and dropdown menus for 'column(s)' and 'after trainingSessionID'.

Milestones Table.

```
CREATE TABLE milestones (
  milestoneID INT(11) NOT NULL AUTO_INCREMENT,
  milestoneDescription VARCHAR(25) NOT NULL,
  PRIMARY KEY (milestoneID);

INSERT INTO `milestones` (`milestoneID`, `milestoneDescription`) VALUES
(1, 'flight theory'),
(2, 'basic aerodynamics'),
(3, 'care of glider and flight equipment'),
(4, 'glider assembly'),
(5, 'pre-flight safety inspections'),
(6, 'ground handling in varying velocities of wind and turbulence'),
(7, 'low altitude flight skills'),
(8, 'cross-wind take offs'),
(9, 'cross-wind landings'),
(10, 'safe and effective turning skills'),
(11, '360 degree turns'),
(12, 'stall recognition'),
(13, 'slip recognition'),
(14, 'glide angle control'),
(15, 'in-flight wind analysis'),
(16, 'emergency procedures'),
(17, 'consistent landing approaches'),
(18, 'staying aloft for hours at a time'),
(19, 'right-of-way rules'),
(20, 'flying close to other gliders'),
(21, 'flying weather forecasting'),
(22, 'Federal Aviation Regulations'),
(23, 'aeronautical chart usage');
```

				milestoneID	milestoneDescription
<input type="checkbox"/>				1	flight theory
<input type="checkbox"/>				2	basic aerodynamics
<input type="checkbox"/>				3	care of glider and flight equipment
<input type="checkbox"/>				4	glider assembly
<input type="checkbox"/>				5	pre-flight safety inspections
<input type="checkbox"/>				6	ground handling in varying velocities of wind and ...
<input type="checkbox"/>				7	low altitude flight skills
<input type="checkbox"/>				8	cross-wind take offs
<input type="checkbox"/>				9	cross-wind landings
<input type="checkbox"/>				10	safe and effective turning skills
<input type="checkbox"/>				11	360 degree turns
<input type="checkbox"/>				12	stall recognition
<input type="checkbox"/>				13	slip recognition
<input type="checkbox"/>				14	glide angle control
<input type="checkbox"/>				15	in-flight wind analysis
<input type="checkbox"/>				16	emergency procedures
<input type="checkbox"/>				17	consistent landing approaches
<input type="checkbox"/>				18	staying aloft for hours at a time
<input type="checkbox"/>				19	right-of-way rules
<input type="checkbox"/>				20	flying close to other gliders
<input type="checkbox"/>				21	flying weather forecasting
<input type="checkbox"/>				22	Federal Aviation Regulations
<input type="checkbox"/>				23	aeronautical chart usage

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 milestoneID	int(11)			No	None		AUTO_INCREMENT	
<input type="checkbox"/>	2 milestoneDescription	varchar(65)	latin1_swedish_ci		No	None			

Check all With selected: Browse

Print Propose table structure Move columns Normalize

Add 1 column(s) after milestoneDescription

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
	PRIMARY	BTREE	Yes	No	milestoneID	23	A	No	

Pilot Table.

```
CREATE TABLE pilot (
    pilotNum INT(11) NOT NULL AUTO_INCREMENT,
    pilotLastName VARCHAR(25) NOT NULL,
    pilotFirstName VARCHAR(20) NOT NULL,
    PRIMARY KEY (pilotNum);
```

```
INSERT INTO `pilot` (`pilotNum`, `pilotLastName`, `pilotFirstName`) VALUES
(14875, 'Harrelson', 'Jared'),
(17564, 'Dwight', 'Brandon');
```

Screenshot of MySQL Workbench showing the 'pilot' table structure and data.

Data View:

			pilotNum	pilotLastName	pilotFirstName	
<input type="checkbox"/>	Edit	Copy	Delete	14875	Harrelson	Jared
<input type="checkbox"/>	Edit	Copy	Delete	17564	Dwight	Brandon

Table Structure View:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	pilotNum	int(11)	latin1_swedish_ci		No	None		AUTO_INCREMENT	Change Drop More
2	pilotLastName	varchar(25)	latin1_swedish_ci		No	None			Change Drop More
3	pilotFirstName	varchar(20)	latin1_swedish_ci		No	None			Change Drop More

Buttons: [Check all](#) [With selected:](#) [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Spatial](#) [Fulltext](#)

Actions: [Print](#) [Propose table structure](#) [Move columns](#) [Normalize](#)

Add: column(s) after pilotFirstName [Go](#)

Indexes View:

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit	PRIMARY	BTREE	Yes	No	pilotNum	2	A	No	

Product Category Table.

```
CREATE TABLE productCategory (
    modelNum INT(11) NOT NULL AUTO_INCREMENT,
    modelDescription VARCHAR(45) NOT NULL,
    modelManufacturer VARCHAR(45) NOT NULL,
    modelInventoryCount INT(11) NOT NULL,
    modelReorderQuantity INT(11) NOT NULL,
    PRIMARY KEY (modelNum);
```

```
INSERT INTO `productCategory` (`modelNum`, `modelDescription`, `modelManufacturer`, `modelInventoryCount`, `modelReorderQuantity`) VALUES
(1, 'Paraglider - Cayenne', 'Skywalk', 1, 1),
(2, 'Hang Glider - Eagle 180', 'Aeros', 3, 1),
(3, 'Hang Glider - Pulse', 'Airwave', 2, 1),
(4, 'Kiteboard - Reactor 2012', 'Airush', 3, 2),
(5, 'Kiteboard - Spitfire 2010', 'Caution', 4, 2),
(6, 'Hang Glider - Stalker', 'Aeros', 2, 1),
(7, 'Hang Glider - Stealth 151', 'Aeros', 2, 2),
(8, 'Hang Glider - Talon 140', 'Wills Wing', 0, 1),
(9, 'Hang Glider - Talon 160', 'Wills Wing', 0, 1),
(10, 'Paraglider - Tequila', 'Skywalk', 0, 1),
(11, 'Kiteboard - Wasp 3', 'Cabrinha', 1, 2);
```

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	modelNum	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	modelDescription	varchar(45)	latin1_swedish_ci		No	None			Change Drop More
3	modelManufacturer	varchar(45)	latin1_swedish_ci		No	None			Change Drop More
4	modelInventoryCount	int(11)			No	None			Change Drop More
5	modelReorderQuantity	int(11)			No	None			Change Drop More
6	reorderNecessary	tinyint(1)			No	None			Change Drop More

Check all With selected: Primary Index Fulltext

Print Propose table structure Move columns Normalize

Add 1 column(s) after reorderNecessary Go

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	modelNum	11	A	No	

Product Order Table.

```
CREATE TABLE productOrder (
    orderNum INT(11) NOT NULL AUTO_INCREMENT,
    orderDatePlaced DATE NOT NULL,
    orderDateReceived DATE NOT NULL,
    supplierNum INT(11) NOT NULL,
    PRIMARY KEY (orderNum),
    FOREIGN KEY (supplierNum) REFERENCES supplier(supplierNum) ON DELETE CASCADE);
```

```
INSERT INTO productOrder (orderNum, orderDatePlaced, orderDateReceived, supplierNum) VALUES
```

```
(1, '2018-04-01', '2018-04-11', 11),
(2, '2018-04-10', '2018-04-21', 1),
(3, '2018-04-10', '2018-04-17', 10),
(4, '2018-04-21', '2018-04-30', 3),
(5, '2019-01-06', '2019-01-12', 4),
(6, '2019-02-04', '2019-02-21', 4),
(7, '2019-04-22', '2019-05-01', 2),
(8, '2019-04-22', '2019-04-29', 2),
(9, '2020-02-15', '2020-03-01', 9),
(11, '2020-03-21', '2020-03-30', 4),
(12, '2020-04-04', NULL, 6)
```

```
ON DUPLICATE KEY UPDATE
```

```
orderDatePlaced = VALUES(orderDatePlaced),
orderDateReceived = VALUES(orderDateReceived),
supplierNum = VALUES(supplierNum);
```

				orderNum	orderDatePlaced	orderDateReceived	supplierNum
<input type="checkbox"/>				1	2018-04-01	2018-04-11	11
<input type="checkbox"/>				2	2018-04-10	2018-04-21	1
<input type="checkbox"/>				3	2018-04-10	2018-04-17	10
<input type="checkbox"/>				4	2018-04-21	2018-04-30	3
<input type="checkbox"/>				5	2019-01-06	2019-01-12	4
<input type="checkbox"/>				6	2019-02-04	2019-02-21	4
<input type="checkbox"/>				7	2019-04-22	2019-05-01	2
<input type="checkbox"/>				8	2019-04-22	2019-04-29	2
<input type="checkbox"/>				9	2020-02-15	2020-03-01	9
<input type="checkbox"/>				11	2020-03-21	2020-03-30	4
<input type="checkbox"/>				12	2020-04-04	NULL	6
<input type="checkbox"/>				14	2020-05-23	2020-06-11	2
<input type="checkbox"/>				15	2020-05-30	2020-06-15	10

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	orderNum	int(11)			No	None		AUTO_INCREMENT	More
2	orderDatePlaced	date			No	None			More
3	orderDateReceived	date			Yes	NULL			More
4	supplierNum	int(11)			No	None			More

Check all With selected:

Add 1 column(s) after supplierNum Go

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
	PRIMARY	BTREE	Yes	No	orderNum	12	A	No	
	supplierNum	BTREE	No	No	supplierNum	12	A	No	

Restock Line Item Table.

```
CREATE TABLE restockLineItem (
    orderNum INT(11) NOT NULL,
    modelNum INT(11) NOT NULL,
    lineItemEquipmentQuantity INT(11) NOT NULL,
    lineItemEquipmentPrice DECIMAL(7,2) NOT NULL,
    PRIMARY KEY (orderNum, modelNum);
```

```
INSERT INTO `restockLineItem`(`orderNum`, `modelNum`, `lineItemEquipmentQuantity`, `lineItemEquipmentPrice`) VALUES
(1, 5, 1, '2400.00'),
(2, 8, 1, '4300.00'),
(3, 9, 1, '4200.00'),
(4, 4, 1, '5000.00'),
(5, 3, 1, '5200.00'),
(6, 1, 1, '3700.00'),
(7, 1, 1, '3700.00'),
(7, 2, 2, '4800.00'),
(8, 6, 1, '4700.00'),
(9, 7, 1, '5400.00'),
(10, 7, 1, '5400.00'),
(11, 10, 1, '5000.00'),
(12, 11, 1, '2600.00'),
(13, 11, 1, '2600.00');
```

			orderNum	modelNum	lineItemEquipmentQuantity	lineItemEquipmentPrice
<input type="checkbox"/>		Edit			1	5 1 2400.00
<input type="checkbox"/>		Edit			2	8 1 4300.00
<input type="checkbox"/>		Edit			3	9 1 4200.00
<input type="checkbox"/>		Edit			4	4 1 5000.00
<input type="checkbox"/>		Edit			5	3 1 5200.00
<input type="checkbox"/>		Edit			6	1 1 3700.00
<input type="checkbox"/>		Edit			7	1 1 3700.00
<input type="checkbox"/>		Edit			7	2 2 4800.00
<input type="checkbox"/>		Edit			8	6 1 4700.00
<input type="checkbox"/>		Edit			9	7 1 5400.00
<input type="checkbox"/>		Edit			10	7 1 5400.00
<input type="checkbox"/>		Edit			11	10 1 5000.00
<input type="checkbox"/>		Edit			12	11 1 2600.00
<input type="checkbox"/>		Edit			13	11 1 2600.00
<input type="checkbox"/>		Edit			14	1 1 3700.00
<input type="checkbox"/>		Edit			15	8 1 4300.00

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 orderNum	int(11)			No	None			
<input type="checkbox"/>	2 modelNum	int(11)			No	None			
<input type="checkbox"/>	3 lineItemEquipmentQuantity	int(11)			No	None			
<input type="checkbox"/>	4 lineItemEquipmentPrice	decimal(7,2)			No	None			

Check all With selected: Browse Primary Unique Index Spatial Fulltext

Print Propose table structure Add 1 column(s) after lineItemEquipmentPrice Go

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
			PRIMARY	BTREE	Yes	No	orderNum	14	A No
							modelNum	14	A No

Return Item Table.

```
CREATE TABLE returnItem (
    equipmentSerialNum VARCHAR(20) NOT NULL,
    returnNum INT(11) NOT NULL,
    returnReason VARCHAR(75) NOT NULL,
    PRIMARY KEY (equipmentSerialNum, returnNum));
```

```
INSERT INTO `returnItem` (`equipmentSerialNum`, `returnNum`, `returnReason`) VALUES
('266453', 3, 'Cracked Wing'),
('132457', 1, 'Broken'),
('017767', 2, 'Dorky fuselage');
```

Screenshot of the MySQL Workbench interface showing the 'returnItem' table structure and data.

Data View:

			equipmentSerialNum	returnNum	returnReason	
<input type="checkbox"/>				017767	2	Dorky fuselage
<input type="checkbox"/>				132457	1	Broken
<input type="checkbox"/>				266453	3	Cracked Wing

Table Structure View:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	
1	equipmentSerialNum	varchar(20)	latin1_swedish_ci		No	None				
2	returnNum	int(11)			No	None				
3	returnReason	varchar(75)	latin1_swedish_ci		No	None				

Buttons below the table structure:

- Check all
- With selected:

Buttons at the bottom of the interface:

- Print
- Propose table structure
- Move columns
- Normalize
- Add 1 column(s) after returnReason

Indexes View:

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment	
			PRIMARY	BTREE	Yes	No	equipmentSerialNum	3	A	No
							returnNum	3	A	No

Returns Table.

```
CREATE TABLE returns (
    returnNum INT(11) NOT NULL AUTO_INCREMENT,
    customerNum INT(11) NOT NULL,
    returnDate DATE NOT NULL,
    PRIMARY KEY (returnNum),
    FOREIGN KEY (customerNum) REFERENCES customer(customerNum) ON DELETE CASCADE);
```

```
INSERT INTO `returns` (`returnNum`, `customerNum`, `returnDate`) VALUES
(1, 3, '2018-07-30'),
(2, 6, '2019-07-08'),
(3, 1, '2020-06-22');
```

The screenshot shows the MySQL Workbench interface with the 'returns' table selected. At the top, there's a toolbar with various icons for database management. Below the toolbar, the table structure is displayed in a grid format with columns: #, Name, Type, Collation, Attributes, Null, Default, Comments, Extra, and Action. The data grid shows three rows of data with columns: returnNum, customerNum, and returnDate. At the bottom, there are sections for indexes and a search bar.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	returnNum	int(11)			No	None		AUTO_INCREMENT	Edit Change Drop More
2	customerNum	int(11)			No	None			Edit Change Drop More
3	returnDate	date			No	None			Edit Change Drop More

Indexes									
Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit	Rename	PRIMARY	BTREE	Yes	returnNum	3	A	No	
Edit	Rename	Drop	customerNum	BTREE	No	customerNum	3	A	No

Sale Line Item Table.

```
CREATE TABLE saleLineItem (
    purchaseAgreementNum INT(11) NOT NULL,
    equipmentSerialNum VARCHAR(20) NOT NULL,
    PRIMARY KEY (purchaseAgreementNum, equipmentSerialNum));
```

```
INSERT INTO `saleLineItem` (`salesAgreementNum`, `equipmentSerialNum`) VALUES
(10000000, '111111'),
(10000000, '123456'),
(10000001, '563421'),
(10000002, '132457'),
(10000003, '129876'),
(10000004, '378927'),
(10000004, '563422'),
(10000005, '677811'),
(10000005, '017767'),
(10000006, '639831'),
(10000007, '194311'),
(10000008, '266453'),
(10000009, '288545'),
(10000010, '333657');
```

				salesAgreementNum	equipmentSerialNum
<input type="checkbox"/>				10000000	111111
<input type="checkbox"/>				10000000	123456
<input type="checkbox"/>				10000001	563421
<input type="checkbox"/>				10000002	132457
<input type="checkbox"/>				10000003	129876
<input type="checkbox"/>				10000004	378927
<input type="checkbox"/>				10000004	563422
<input type="checkbox"/>				10000005	017767
<input type="checkbox"/>				10000005	677811
<input type="checkbox"/>				10000006	639831
<input type="checkbox"/>				10000007	194311
<input type="checkbox"/>				10000008	266453
<input type="checkbox"/>				10000009	288545
<input type="checkbox"/>				10000010	333657
<input type="checkbox"/>				10000011	402984

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	
<input type="checkbox"/>	1 salesAgreementNum	int(11)			No	None				
<input type="checkbox"/>	2 equipmentSerialNum	varchar(20)	latin1_swedish_ci		No	None				

Check all With selected:

Add 1 column(s) after equipmentSerialNum

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment	
			PRIMARY	BTREE	Yes	No	salesAgreementNum	14	A	No
							equipmentSerialNum	14	A	No

Salesperson Table.

```
CREATE TABLE salesperson (
    salespersonNum INT(11) NOT NULL,
    salesmanCommissionPercent DECIMAL(4,2) NOT NULL,
    salesmanSalesCount INT(3) NOT NULL,
    salesmanSalesAmount INT(3) NOT NULL,
    PRIMARY KEY (salespersonNum),
    FOREIGN KEY (salespersonNum) REFERENCES employee(employeeNum) ON DELETE CASCADE);
```

```
INSERT INTO `salesperson` (`salespersonNum`, `salespersonCommissionPercent`, `salespersonSalesCount`, `salespersonSalesAmount`) VALUES
(1, 0.03, 74, 444000.00),
(2, 0.025, 22, 136400.00),
(3, 0.025, 51, 295800.00),
(4, 0.025, 46, 271400.00),
(6, 0.025, 34, 207400.00);
```

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	salespersonNum	int(11)			No	None			Change Drop More
2	salespersonCommissionPercent	decimal(4,3)			No	None			Change Drop More
3	salespersonSalesCount	int(3)			No	None			Change Drop More
4	salespersonSalesAmount	int(3)			No	None			Change Drop More

Check all With selected: [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Spatial](#) [Fulltext](#)

[Print](#) [Propose table structure](#) [Move columns](#) [Normalize](#)

[Add](#) column(s) after salespersonSalesAmount [Go](#)

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit	PRIMARY	BTREE	Yes	No	salespersonNum	5	A	No	

sLogs Pilot Table.

```
CREATE TABLE sLogsPilot (
    studentLogBookNum INT(11) NOT NULL,
    pilotNum INT(11) NOT NULL,
    PRIMARY KEY (studentLogBookNum, pilotNum));
```

```
INSERT INTO `sLogsPilot` (`studentLogBookNum`, `pilotNum`) VALUES
(10012, 14875),
(10013, 17564);
```

The screenshot shows the MySQL Workbench interface for the `sLogsPilot` table. At the top, there is a data grid displaying two rows of data:

			studentLogBookNum	pilotNum		
<input type="checkbox"/>	<input type="checkbox"/>	Edit	Copy	Delete	10012	14875
<input type="checkbox"/>	<input type="checkbox"/>	Edit	Copy	Delete	10013	17564

Below the data grid is the **Table structure** tab, which displays the columns and their properties:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	studentLogBookNum	int(11)			No	None			<input type="checkbox"/> Change <input type="checkbox"/> Drop <input type="checkbox"/> More
2	pilotNum	int(11)			No	None			<input type="checkbox"/> Change <input type="checkbox"/> Drop <input type="checkbox"/> More

Below the table structure are several buttons: Check all, With selected: Browse, Change, Drop, Primary, Unique, Index, Spatial, Fulltext.

At the bottom left, there are buttons for Print, Propose table structure, Move columns, Normalize, Add (with a dropdown menu for 1 column(s) after pilotNum), and Go.

Below the table structure is the **Indexes** tab, which lists the primary key index:

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
<input type="checkbox"/> Edit <input type="checkbox"/> Rename <input type="checkbox"/> Drop	PRIMARY	BTREE	Yes	No	studentLogBookNum pilotNum	2 2	A A	No No	

Student Log Book Table.

```
CREATE TABLE studentLogBook (
    studentLogBookNum INT(11) NOT NULL AUTO_INCREMENT,
    totalFlightHours INT(2) NOT NULL,
    flightDate DATE NOT NULL,
    site VARCHAR(25) NOT NULL,
    lzLocation VARCHAR(25) NOT NULL,
    gliderNumber VARCHAR(8) NOT NULL,
    flightNumberForDay VARCHAR(8) NOT NULL,
    launchAltitude DECIMAL(6,2) NOT NULL,
    lzAltitude VARCHAR(8) NOT NULL,
    launchTime TIME NOT NULL,
    flightDuration DECIMAL(3,2) NOT NULL,
    straightLineDistanceLaunchToLanding DECIMAL(5,2) NOT NULL,
    pathLengthFromGPSTrack DECIMAL(5,2) NOT NULL,
    altitudeMax DECIMAL(6,2) NOT NULL,
    altitudeMin DECIMAL(6,2) NOT NULL,
    altitudeGainFromVario DECIMAL(6,2) NOT NULL,
    maxClimbRate DECIMAL(6,2) NOT NULL,
    maxSinkRate DECIMAL(6,2) NOT NULL,
    gpsTrackAvailable BOOLEAN NOT NULL,
    importantNotesOrIncidents BOOLEAN NOT NULL,
    traineeID INT(11) NOT NULL,
    PRIMARY KEY (studentLogBookNum),
    FOREIGN KEY (traineeID) REFERENCES trainee(traineeID) ON DELETE CASCADE);
```

```
INSERT INTO `studentLogBook` (`studentLogBookNum`, `totalFlightHours`, `flightDate`, `site`, `lzLocation`, `gliderNumber`, `flightNumberForDay`, `launchAltitude`, `lzAltitude`, `launchTime`, `flightDuration`, `straightLineDistanceLaunchToLanding`, `pathLengthFromGPSTrack`, `altitudeMax`, `altitudeMin`, `altitudeGainFromVario`, `maxClimbRate`, `maxSinkRate`, `gpsTrackAvailable`, `flightStoryWritten`, `importantNotesOrIncidents`, `traineeID`) VALUES
(110011, 120, '2020-08-14', 'Big Southern Butte', 'Site 2', '24B98', '224', 5200.00, 4150.00, '13:42:00', 1.50, 10.4, 12.8, 12000.50, 9400.32, 12896.42, 24.2, 49.5, TRUE, TRUE, 'Strong gusts above 10000', 10001),
(110012, 116, '2020-08-02', 'Big Southern Butte', 'Site 2', '24B75', '224', 5200.00, 4150.00, '11:15:00', 1.80, 10.2, 14.7, 13400.00, 9100.38, 13214.80, 31.6, 54.8, TRUE, TRUE, 'Paragliders interfered with flight path', 10001),
(110013, 52, '2020-08-02', 'Big Southern Butte', 'Site 1', '266A9', '224', 5200.00, 4150.00, '12:30:00', 0.75, 10.4, 13.2, 10000.00, 8800.74, 12647.68, 20.7, 34.7, TRUE, TRUE, 'Difficulty turning into strong winds', 10002);
```

		studentLogBookNum	totalFlightHours	flightDate	site	lzLocation	gliderNumber	flightNumberForDay	launchAltitude	lzAltitude	launchTime	flightDuration	straightLineDistanceLaunchToLanding	pathLengthFromGPSTrack	altitudeMax	altitudeMin	altitudeGainFromVario	maxClimbRate	maxSinkRate	gpsTrackAvailable	flightStoryWritten	importantNotesOrIncidents	traineeID
		110011	120.00	2020-08-14	Big Southern Butte	Site 2	24B98	224	5200.00	4150.00	13:42:00	1.50	10.40								Strong gusts above 10000	10001	
		110012	116.00	2020-08-02	Big Southern Butte	Site 2	24B75	224	5200.00	4150.00	11:15:00	1.80	10.20								Paragliders interfered with flight path	10001	
		110013	52.00	2020-08-02	Big Southern Butte	Site 1	266A9	224	5200.00	4150.00	12:30:00	0.75	10.40								Difficulty turning into strong winds	10002	
12.80			12000.50	9400.32	12896.42		24.20	49.50	1		1										Strong gusts above 10000	10001	
14.70			13400.00	9100.38	13214.80		31.60	54.80	1		1										Paragliders interfered with flight path	10001	
13.20			10000.00	8800.74	12647.68		20.70	34.70	1		1										Difficulty turning into strong winds	10002	

Table structure		Relation view								
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	
1	studentLogBookNum	int(11)			No	None	AUTO_INCREMENT		Change	Drop
2	totalFlightHours	decimal(6,2)			No	None			Change	Drop
3	flightDate	date			No	None			Change	Drop
4	site	varchar(25)	latin1_swedish_ci		No	None			Change	Drop
5	lzLocation	varchar(25)	latin1_swedish_ci		No	None			Change	Drop
6	gliderNumber	varchar(8)	latin1_swedish_ci		No	None			Change	Drop
7	flightNumberForDay	varchar(8)	latin1_swedish_ci		No	None			Change	Drop
8	launchAltitude	decimal(7,2)			No	None			Change	Drop
9	lzAltitude	decimal(7,2)			No	None			Change	Drop
10	launchTime	time			No	None			Change	Drop
11	flightDuration	decimal(5,2)			No	None			Change	Drop
12	straightLineDistanceLaunchToLanding	decimal(6,2)			No	None			Change	Drop
13	pathLengthFromGPSTrack	decimal(6,2)			No	None			Change	Drop
14	altitudeMax	decimal(8,2)			No	None			Change	Drop
15	altitudeMin	decimal(8,2)			No	None			Change	Drop
16	altitudeGainFromVario	decimal(8,2)			No	None			Change	Drop
17	maxClimbRate	decimal(7,2)			No	None			Change	Drop
18	maxSinkRate	decimal(7,2)			No	None			Change	Drop
19	gpsTrackAvailable	tinyint(1)			No	None			Change	Drop
20	flightStoryWritten	tinyint(1)			No	None			Change	Drop
21	importantNotesOrIncidents	varchar(50)	latin1_swedish_ci		Yes	NULL			Change	Drop
22	traineeID	int(11)			No	None			Change	Drop

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit	Rename	PRIMARY	BTREE	Yes	No	studentLogBookNum	3	A	No
Edit	Rename	traineeID	BTREE	No	No	traineeID	3	A	No

Supplier Table.

```
CREATE TABLE supplier (
    supplierNum INT(11) NOT NULL AUTO_INCREMENT,
    supplierName VARCHAR(45) NOT NULL,
    supplierAddress VARCHAR(30) NOT NULL,
    supplierCity VARCHAR(30) NOT NULL,
    supplierState VARCHAR(2) NOT NULL,
    supplierZip VARCHAR(10) NOT NULL,
    supplierPhone VARCHAR(15) NOT NULL,
    supplierFax VARCHAR(15) NOT NULL,
    supplierContact VARCHAR(45) NOT NULL,
    supplierEmail VARCHAR(35) NOT NULL,
    PRIMARY KEY (supplierNum));
```

```
INSERT INTO `supplier` (`supplierNum`, `supplierName`, `supplierAddress`, `supplierCity`, `supplierState`, `supplierZip`, `supplierPhone`, `supplierFax`, `supplierContact`, `supplierEmail`) VALUES
(1, 'Wills Wing', '500 West Blueridge Ave', 'Orange', 'CA', '92865-1001', '(714) 998-0647', '(714) 217-8822', 'Linda Meier', 'linda@willswing.com'),
(2, 'Aeros', '1125 Harborview Dr.', 'Kill Devil Hills', 'CO', '27948-0000', '(252) 480-3552', '(336) 217-8822', 'Gill Witt', 'gw@justfly.com'),
(3, 'Airwave', '500 West Blueridge Ave.', 'Orange', 'CA', '92865-1001', '(714) 480-6359', '(714) 998-0647', 'Mike Meier', 'mike@airwave.com'),
(4, 'Skywalk', '3345 Porter', 'Timbuktu', 'NM', '66345-0000', '(223) 222-5555', '(223) 223-5555', 'Cliff Trip', 'ct@skwalkms.com'),
(5, 'Flightstar', 'P.O. Box 760', 'Ellington', 'CA', '06029-0000', '(960) 875-8185', '(860) 870-5499', 'Jim High', 'JH@flyflightstar.com'),
(6, 'Quicksilver', '42241 Sarah Way', 'Temecula', 'FL', '92590-2100', '(951) 506-0061', '(951) 506-1589', 'Todd Ellefson', 'quicksilversales@pe.net'),
(7, 'Phantom', '2225 Flight Road', 'Three Rivers', 'MN', '74665-0000', '(269) 271-4568', '(269) 592-5908', 'Eric Pederson', 'epederson@phantomaero.com'),
(8, 'Slingshot', '1111 Beach St.', 'Santa Cruz', 'CA', '95223-1234', '(408) 222-1234', NULL, 'Martin Vari', 'mvari@sskiteboarding.com'),
(9, 'Cabrhina', '400 Hana Highway', 'Maui', 'HI', '96732-0101', '(808) 877-7443', '(808) 877-2149', 'Niel Pryde', 'npryde@cabrhina.com'),
(10, 'Airush', '111 Hana Highway', 'Kahalui', 'HI', '96732-0101', '(808) 877-1087', '(808) 871-2144', 'Joe Dent', 'jd@airush_boarding.com'),
(11, 'Caution', '430 Westline Drive', 'Alameda', 'CA', '94501-0000', '(510) 522-9463', NULL, 'Jody Mills', 'jody@caution_kitesurfing.com'),
(12, 'Xnation', '10030 North Temple', 'Salt Lake City', 'UT', '88989-1023', '(800) 688-2802', NULL, 'Lance Harris', 'Lance@xnation.biz');
```

		supplierNum	supplierName	supplierAddress	supplierCity	supplierState	supplierZip	supplierPhone	supplierFax	supplierContact	supplierEmail
<input type="checkbox"/>		1	Wills Wing	500 West Blueridge Ave	Orange	CA	92865-1001	(714) 998-0647	(714) 217-8822	Linda Meier	linda@willswing.com
<input type="checkbox"/>		2	Aeros	1125 Harborview Dr.	Kill Devil Hills	CO	27948-0000	(252) 480-3552	(336) 217-8822	Gill Witt	gw@justfly.com
<input type="checkbox"/>		3	Airwave	500 West Blueridge Ave.	Orange	CA	92865-1001	(714) 480-6359	(714) 998-0647	Mike Meier	mike@airwave.com
<input type="checkbox"/>		4	Skywalk	3345 Porter	Timbuktu	NM	66345-0000	(223) 222-5555	(223) 223-5555	Cliff Trip	ct@skwalkms.com
<input type="checkbox"/>		5	Flightstar	P.O. Box 760	Ellington	CA	06029-0000	(960) 875-8185	(860) 870-5499	Jim High	JH@flyflightstar.com
<input type="checkbox"/>		6	Quicksilver	42241 Sarah Way	Temecula	FL	92590-2100	(951) 506-0061	(951) 506-1589	Todd Ellefson	quicksilversales@pe.net
<input type="checkbox"/>		7	Phantom	2225 Flight Road	Three Rivers	MN	74665-0000	(269) 271-4568	(269) 592-5908	Eric Pederson	epederson@phantomaero.com
<input type="checkbox"/>		8	Slingshot	1111 Beach St.	Santa Cruz	CA	95223-1234	(408) 222-1234	NULL	Martin Vari	mvari@sskiteboarding.com
<input type="checkbox"/>		9	Cabrhina	400 Hana Highway	Maui	HI	96732-0101	(808) 877-7443	(808) 877-2149	Niel Pryde	npryde@cabrhina.com
<input type="checkbox"/>		10	Airush	111 Hana Highway	Kahalui	HI	96732-0101	(808) 877-1087	(808) 871-2144	Joe Dent	jd@airush_boarding.com
<input type="checkbox"/>		11	Caution	430 Westline Drive	Alameda	CA	94501-0000	(510) 522-9463	NULL	Jody Mills	jody@caution_kitesurfing.com
<input type="checkbox"/>		12	Xnation	10030 North Temple	Salt Lake City	UT	88989-1023	(800) 688-2802	NULL	Lance Harris	Lance@xnation.biz

Table structure		Relation view									
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
<input type="checkbox"/>	1 supplierNum	int(11)			No	None		AUTO_INCREMENT			
<input type="checkbox"/>	2 supplierName	varchar(45)	latin1_swedish_ci		No	None					
<input type="checkbox"/>	3 supplierAddress	varchar(30)	latin1_swedish_ci		No	None					
<input type="checkbox"/>	4 supplierCity	varchar(30)	latin1_swedish_ci		No	None					
<input type="checkbox"/>	5 supplierState	varchar(2)	latin1_swedish_ci		No	None					
<input type="checkbox"/>	6 supplierZip	varchar(10)	latin1_swedish_ci		No	None					
<input type="checkbox"/>	7 supplierPhone	varchar(15)	latin1_swedish_ci		No	None					
<input type="checkbox"/>	8 supplierFax	varchar(15)	latin1_swedish_ci		Yes	NULL					
<input type="checkbox"/>	9 supplierContact	varchar(45)	latin1_swedish_ci		No	None					
<input type="checkbox"/>	10 supplierEmail	varchar(35)	latin1_swedish_ci		No	None					

Indexes											
Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment		
	Rename		PRIMARY	BTREE	Yes	No	supplierNum	12	A	No	

Trainee Table.

```
CREATE TABLE trainee (
    traineeID INT(11) NOT NULL AUTO_INCREMENT,
    traineeLastName VARCHAR(25) NOT NULL,
    traineeFirstName VARCHAR(20) NOT NULL,
    traineeAddress VARCHAR(45) NOT NULL,
    traineeCity VARCHAR(15) NOT NULL,
    traineeState CHAR(2) NOT NULL,
    traineeZip VARCHAR(10) NOT NULL,
    traineePhone VARCHAR(15) NOT NULL,
    traineeEmail VARCHAR(25) NOT NULL,
    ushpaRating VARCHAR(4) NOT NULL,
    trainingHours DECIMAL(5,1) NOT NULL,
    trainingFlights INT(2) NOT NULL,
    dateOfBirth DATE NOT NULL,
    equipmentType VARCHAR(20) NOT NULL,
    weight INT(3) NOT NULL,
    PRIMARY KEY (traineeID));
```

```
INSERT INTO `trainee` ('traineeID', 'traineeLastName', 'traineeFirstName', 'traineeAddress', 'traineeCity', 'traineeState', 'traineeZip', 'traineePhone', 'traineeEmail', 'ushpaRating', 'trainingHours', 'trainingFlights', 'dateOfBirth', 'weight') VALUES
(10001, 'Etheridge', 'Walt', '1500 WestWind', 'Jackson', 'WY', '83002', '(307) 451-7634', 'weth@gmail.com', 'H-1', 120, 80, '1995-04-14', 185),
(10002, 'Wallace', 'Jimmy', '414 Elk Dr', 'Panguitch', 'UT', '84759', '(801) 391-9210', 'JWallace414@gmail.com', 'H-0', 52, 40, '2000-09-06', 160);
```

	traineeID	traineeLastName	traineeFirstName	traineeAddress	traineeCity	traineeState	traineeZip	traineePhone	traineeEmail	ushpaRating	trainingHours	trainingFlights	dateOfBirth	weight			
	Edit	Copy	Delete	10001	Etheridge	Walt	1500 WestWind	Jackson	WY	83002	(307) 451-7634	weth@gmail.com	H-1	120.00	80	1995-04-14	185
	Edit	Copy	Delete	10002	Wallace	Jimmy	414 Elk Dr	Panguitch	UT	84759	(801) 391-9210	JWallace414@gmail.com	H-0	52.00	40	2000-09-06	160

Table structure		Relation view													
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	Change	Drop	More			
1	traineeID	int(11)			No	None		AUTO_INCREMENT							
2	traineeLastName	varchar(25)	latin1_swedish_ci		No	None									
3	traineeFirstName	varchar(20)	latin1_swedish_ci		No	None									
4	traineeAddress	varchar(45)	latin1_swedish_ci		No	None									
5	traineeCity	varchar(15)	latin1_swedish_ci		No	None									
6	traineeState	char(2)	latin1_swedish_ci		No	None									
7	traineeZip	varchar(10)	latin1_swedish_ci		No	None									
8	traineePhone	varchar(15)	latin1_swedish_ci		No	None									
9	traineeEmail	varchar(25)	latin1_swedish_ci		No	None									
10	ushpaRating	varchar(4)	latin1_swedish_ci		No	None									
11	trainingHours	decimal(6,2)			No	None									
12	trainingFlights	int(2)			No	None									
13	dateOfBirth	date			No	None									
14	weight	int(3)			No	None									

Indexes											
Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment		
Edit	Rename	Drop	PRIMARY	BTREE	Yes	No	traineeID	2	A	No	

Trainer Table.

```
CREATE TABLE trainer (
    trainerID INT(11) NOT NULL,
    ushpaRating VARCHAR(4) NOT NULL,
    apprenticeshipCompleted BOOLEAN NOT NULL,
    clinicPassed BOOLEAN NOT NULL,
    foiTestCompleted BOOLEAN NOT NULL,
    holdCprFirstAidCertification BOOLEAN NOT NULL,
    rogalloMembership BOOLEAN NOT NULL,
    PRIMARY KEY (trainerID),
    FOREIGN KEY (trainerID) REFERENCES employee(employeeNum) ON DELETE CASCADE);
```

```
INSERT INTO `trainer` (`trainerID`, `ushpaRating`, `apprenticeshipCompleted`, `clinicPassed`, `foiTestCompleted`, `holdCprFirstAidCertification`, `rogalloMembership`) VALUES
(5, 'H-4', TRUE, TRUE, TRUE, TRUE),
(6, 'H-5', TRUE, TRUE, TRUE, TRUE);
```

		trainerID	ushpaRating	apprenticeshipCompleted	clinicPassed	foiTestCompleted	holdCprFirstAidCertification	rogalloMembership
		Edit	Copy	Delete				
	5	H-4	1	1	1	1	1	1
	6	H-5	1	1	1	1	1	1

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	trainerID	int(11)			No	None			Change Drop More
2	ushpaRating	varchar(4)	latin1_swedish_ci		No	None			Change Drop More
3	apprenticeshipCompleted	tinyint(1)			No	None			Change Drop More
4	clinicPassed	tinyint(1)			No	None			Change Drop More
5	foiTestCompleted	tinyint(1)			No	None			Change Drop More
6	holdCprFirstAidCertification	tinyint(1)			No	None			Change Drop More
7	rogalloMembership	tinyint(1)			No	None			Change Drop More

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit	Rename	Drop							
	PRIMARY	BTREE	Yes	No	trainerID	2	A	No	

Training Session Table.

```
CREATE TABLE trainingSession (
```

```
    trainingSessionID INT(11) NOT NULL AUTO_INCREMENT,
```

```
    sessionDate DATE NOT NULL,
```

```
    trainerID INT(11) NOT NULL,
```

```
    PRIMARY KEY (trainingSessionID),
```

```
    FOREIGN KEY (trainerID) REFERENCES trainer(trainerID) ON DELETE CASCADE);
```

```
INSERT INTO `trainingSession` (`trainingSessionID`, `level`, `sessionDate`, `trainerID`) VALUES
```

```
(1, 'novice', '2020-06-08', 5),
```

```
(2, 'novice', '2020-06-09', 5),
```

```
(3, 'novice', '2020-06-10', 5),
```

```
(4, 'novice', '2020-06-11', 5),
```

```
(5, 'intermediate', '2020-06-18', 6),
```

```
(6, 'intermediate', '2020-06-19', 6);
```

		trainingSessionID				level	sessionDate	trainerID
		Edit	Copy	Delete	1	novice	2020-06-08	5
		Edit	Copy	Delete	2	novice	2020-06-09	5
		Edit	Copy	Delete	3	novice	2020-06-10	5
		Edit	Copy	Delete	4	novice	2020-06-11	5
		Edit	Copy	Delete	5	intermediate	2020-06-18	6
		Edit	Copy	Delete	6	intermediate	2020-06-19	6

		Table structure									
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	More	
1	trainingSessionID	int(11)			No	None	AUTO_INCREMENT		Change	Drop	More
2	level	varchar(12)	latin1_swedish_ci		No	None			Change	Drop	More
3	sessionDate	date			No	None			Change	Drop	More
4	trainerID	int(11)			No	None			Change	Drop	More

Indexes											
Action	Rename	Drop	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit	Rename	Drop	PRIMARY	BTREE	Yes	No	trainingSessionID	6	A	No	
Edit	Rename	Drop	trainerID	BTREE	No	No	trainerID	6	A	No	

Transaction Table.

```
CREATE TABLE transaction (
    purchaseAgreementNum INT(11) NOT NULL AUTO_INCREMENT,
    purchaseAgreementDate DATE NOT NULL,
    purchaseAgreementTime TIME NOT NULL,
    purchaseAgreementPaymentType VARCHAR(11) NOT NULL,
    purchaseAgreementCardNumberUsed VARCHAR(20) NOT NULL,
    customerNum INT(11) NOT NULL,
    salespersonNum INT(11) NOT NULL,
    PRIMARY KEY (purchaseAgreementNum),
    FOREIGN KEY (customerNum) REFERENCES customer(customerNum) ON DELETE CASCADE,
    FOREIGN KEY (salespersonNum) REFERENCES salesperson(salespersonNum) ON DELETE CASCADE);
```

```
INSERT INTO `transaction` (`salesAgreementNum`, `salesAgreementDate`, `salesAgreementTime`, `salesAgreementPaymentType`, `salesAgreementCardNumberUsed`, `customerNum`, `salespersonNum`) VALUES
(10000000, '2018-06-16', '08:34:00', 'Credit Card', '4120666734254132', 1, 2),
(10000001, '2018-07-05', '16:45:00', 'Credit Card', '5140123408221452', 2, 3),
(10000002, '2018-07-14', '14:02:00', 'Credit Card', '8564858986567844', 3, 1),
(10000003, '2019-06-30', '12:13:00', 'Credit Card', '4555554444333221', 4, 2),
(10000004, '2019-06-06', '13:31:00', 'Credit Card', '3666677778888999', 5, 2),
(10000005, '2019-07-02', '12:37:00', 'Cash', NULL, 6, 4),
(10000006, '2020-05-30', '11:23:00', 'Cash', NULL, 7, 3),
(10000007, '2020-06-21', '10:00:00', 'Cash', NULL, 8, 3),
(10000008, '2020-06-21', '11:21:00', 'Credit Card', '5268080043376894', 1, 3),
(10000009, '2020-06-22', '15:18:00', 'Credit Card', '5268080043376894', 1, 3),
(10000010, '2020-07-24', '15:32:00', 'Credit Card', '4555554444333221', 4, 3);
```

			salesAgreementNum	salesAgreementDate	salesAgreementTime	salesAgreementPaymentType	salesAgreementCardNumberUsed	customerNum	salespersonNum	
<input type="checkbox"/>	Edit	Copy	Delete	10000000	2018-06-16	08:34:00	Credit Card	4120666734254132	1	2
<input type="checkbox"/>	Edit	Copy	Delete	10000001	2018-07-05	16:45:00	Credit Card	5140123408221452	2	3
<input type="checkbox"/>	Edit	Copy	Delete	10000002	2018-07-14	14:02:00	Credit Card	8564858986567844	3	1
<input type="checkbox"/>	Edit	Copy	Delete	10000003	2019-06-30	12:13:00	Credit Card	4555554444333221	4	2
<input type="checkbox"/>	Edit	Copy	Delete	10000004	2019-06-06	13:31:00	Credit Card	3666677778888999	5	2
<input type="checkbox"/>	Edit	Copy	Delete	10000005	2019-07-02	12:37:00	Credit Card	5268080043376894	6	4
<input type="checkbox"/>	Edit	Copy	Delete	10000006	2020-05-30	11:23:00	Cash	NULL	7	3
<input type="checkbox"/>	Edit	Copy	Delete	10000007	2020-06-21	10:00:00	Cash	NULL	8	3
<input type="checkbox"/>	Edit	Copy	Delete	10000008	2020-06-21	11:21:00	Credit Card	5268080043376894	1	3
<input type="checkbox"/>	Edit	Copy	Delete	10000009	2020-06-22	15:18:00	Credit Card	5268080043376894	1	3
<input type="checkbox"/>	Edit	Copy	Delete	10000010	2020-07-24	15:32:00	Credit Card	4555554444333221	4	3
<input type="checkbox"/>	Edit	Copy	Delete	10000011	2020-06-01	14:42:00	Credit Card	3666677778888999	5	2
<input type="checkbox"/>	Edit	Copy	Delete	10000015	2021-04-14	11:00:57	Credit Card	5268080043376894	2	1

[Table structure](#) [Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 salesAgreementNum 	int(11)			No	None	AUTO_INCREMENT		Change Drop More
<input type="checkbox"/>	2 salesAgreementDate	date			No	None			Change Drop More
<input type="checkbox"/>	3 salesAgreementTime	time			No	current_timestamp()			Change Drop More
<input type="checkbox"/>	4 salesAgreementPaymentType	varchar(11)	latin1_swedish_ci		No	None			Change Drop More
<input type="checkbox"/>	5 salesAgreementCardNumberUsed	varchar(20)	latin1_swedish_ci		Yes	NULL			Change Drop More
<input type="checkbox"/>	6 customerNum 	int(11)			No	None			Change Drop More
<input type="checkbox"/>	7 salespersonNum 	int(11)			No	None			Change Drop More

Indexes 										
Action		Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit	Rename	Drop	PRIMARY	BTREE	Yes	No	salesAgreementNum	A	No	
Edit	Rename	Drop	customerNum	BTREE	No	No	customerNum	A	No	
Edit	Rename	Drop	salespersonNum	BTREE	No	No	salespersonNum	A	No	

Query Design

Q01. List the model description and manufacturer for all products in the product category table whose manufacturer name starts with an "A" and has a third letter "r". (Your answer should include Aeros, Airwave, and Airush.)

```
SELECT productCategory.modelDescription, productCategory.modelManufacturer  
FROM productCategory  
WHERE productCategory.modelManufacturer LIKE "A_r%"
```

modelDescription	modelManufacturer
Hang Glider - Eagle 180	Aeros
Hang Glider - Pulse	Airwave
Kiteboard - Reactor 2012	Airush
Hang Glider - Stalker	Aeros
Hang Glider - Stealth 151	Aeros

Q02. List all salesperson information, including salesperson number, last name, first name, sales count, and sales total (salespersonSalesAmount) and commission earned in descending order by commission earned.

```
SELECT salesperson.salespersonNum, employee.employeeLastName, employee.employeeFirstName, salesperson.salespersonSalesCount,  
salesperson.salespersonSalesAmount, salesperson.salespersonCommissionPercent*salesperson.salespersonSalesAmount AS `Commission Earned`  
FROM employee, salesperson  
WHERE employee.employeeNum = salesperson.salespersonNum  
ORDER BY `Commission Earned` DESC
```

salespersonNum	employeeLastName	employeeFirstName	salespersonSalesCount	salespersonSalesAmount	Commission Earned
1	Hall	Meredith	74	444000	13320.000
3	Teel	Lee	51	295800	7395.000
4	Fries	Jim	46	271400	6785.000
6	Long	Tom	34	207400	5185.000
2	Davis	Art	22	136400	3410.000

Q03. List the customer number, customer name (concatenated like “First Last”), sales agreement number, and sale price for each transaction. Give the concatenated name a meaningful alias.

```
SELECT customer.customerNum, CONCAT(customer.customerFirstName, ' ', customer.customerLastName) AS `Customer Name`,
transaction.salesAgreementNum, itemSold.equipmentSalePrice, SUM(equipmentSalePrice) AS `Purchase Total`
FROM customer
Inner JOIN transaction ON customer.customerNum = transaction.customerNum
INNER JOIN saleLineItem ON transaction.salesAgreementNum = saleLineItem.salesAgreementNum
INNER JOIN itemSold ON saleLineItem.equipmentSerialNum = itemSold.equipmentSerialNum
GROUP BY customerNum, salesAgreementNum
```

customerNum	Customer Name	salesAgreementNum	equipmentSalePrice	Purchase Total
1	Walter Jones	10000000	5500.00	10230.00
1	Walter Jones	10000008	5940.00	5940.00
1	Walter Jones	10000009	2860.00	2860.00
2	Bert Smith	10000001	2640.00	2640.00
3	Janice Mall	10000002	4620.00	4620.00
4	Jim "Great" Dane	10000003	5720.00	5720.00
4	Jim "Great" Dane	10000010	2860.00	2860.00
5	Edna James	10000004	4070.00	8140.00
5	Edna James	10000011	4070.00	4070.00
6	Walt Etheridge	10000005	5170.00	10450.00
7	Edward Haskell	10000006	5940.00	5940.00
8	Ed Wallace	10000007	5500.00	5500.00

Q04. List the customer name (concatenated like “First Last”), serial number, model number, model description, and transaction date for all items having a transaction date of June 21, 2020.

```
SELECT CONCAT(customer.customerFirstName, ' ', customer.customerLastName) AS 'Customer Name', itemSold.equipmentSerialNum, itemSold.modelNum,
productCategory.modelDescription, transaction.salesAgreementDate
FROM customer, transaction, itemSold, saleLineItem, productCategory
WHERE customer.customerNum = transaction.customerNum
AND transaction.salesAgreementNum = saleLineItem.salesAgreementNum
AND saleLineItem.equipmentSerialNum = itemSold.equipmentSerialNum
AND itemSold.modelNum = productCategory.modelNum
```

Customer Name	equipmentSerialNum	modelNum	modelDescription	salesAgreementDate
Walter Jones	111111	4	Kiteboard - Reactor 2012	2018-06-16
Walter Jones	123456	8	Hang Glider - Talon 140	2018-06-16
Walter Jones	266453	7	Hang Glider - Stealth 151	2020-06-21
Walter Jones	288545	11	Kiteboard - Wasp 3	2020-06-22
Bert Smith	563421	5	Kiteboard - Spitfire 2010	2018-07-05
Janice Mall	132457	9	Hang Glider - Talon 160	2018-07-14
Jim "Great" Dane	129876	3	Hang Glider - Pulse	2019-06-30
Jim "Great" Dane	333657	11	Kiteboard - Wasp 3	2020-07-24
Edna James	378927	1	Paraglider - Cayenne	2019-06-06
Edna James	563422	1	Paraglider - Cayenne	2019-06-06
Edna James	402984	1	Paraglider - Cayenne	2020-06-01
Walt Etheridge	017767	6	Hang Glider - Stalker	2019-07-02
Walt Etheridge	677811	2	Hang Glider - Eagle 180	2019-07-02
Edward Haskell	639831	7	Hang Glider - Stealth 151	2020-05-30
Ed Wallace	194311	10	Paraglider - Tequila	2020-06-21

Q05. List all orders that were placed over 1 month ago and have not been received. Include order number, order date placed, order date received, supplier number, supplier name, supplier contact, and supplier phone. List by order date placed from oldest to most recent. INTERVAL is useful here.

```
SELECT productOrder.orderNum, productOrder.orderDatePlaced, productOrder.orderDateReceived, supplier.supplierNum, supplier.supplierName,
supplier.supplierContact, supplier.supplierPhone
FROM productOrder, supplier
WHERE productOrder.supplierNum = supplier.supplierNum
AND productOrder.orderDatePlaced < CURRENT_DATE - INTERVAL 1 MONTH
AND productOrder.orderDateReceived IS NULL
ORDER BY productOrder.orderDatePlaced ASC
```

orderNum	orderDatePlaced	orderDateReceived	supplierNum	supplierName	supplierContact	supplierPhone
12	2020-04-04	NULL	6	Quicksilver	Todd Ellefson	(951) 506-0061

Q06. List supplier number and supplier name for all suppliers for which there are no current orders. Sort the list in ascending order by supplier name.

```
SELECT DISTINCT supplier.supplierNum, supplier.supplierName
FROM supplier, productOrder
WHERE productOrder.orderDateReceived IS NOT NULL
ORDER BY supplier.supplierName
```

supplierNum	supplierName
2	Aeros
10	Airush
3	Airwave
9	Cabrhma
11	Caution
5	Flightstar
7	Phantom
6	Quicksilver
4	Skywalk
8	Slingshot
1	Wills Wing
12	Xnation

Q07. List all sales in the past two years. Include sales date, customer number, customer name (concatenated like “First Last”), model number, model description, model manufacturer, and sales price. Sort the output by sales date.

```
SELECT transaction.salesAgreementDate AS `Sales Date`, customer.customerNum, CONCAT(customer.customerFirstName, ' ', customer.customerLastName) AS 'Customer', productCategory.modelNum AS `Model #`, productCategory.modelDescription AS `Model Description`, productCategory.modelManufacturer AS 'Manufacturer', itemSold.equipmentSalePrice AS Price
FROM customer, transaction, itemSold, productCategory, saleLineItem
WHERE customer.customerNum = transaction.customerNum
AND transaction.salesAgreementNum = saleLineItem.salesAgreementNum
AND saleLineItem.equipmentSerialNum = itemSold.equipmentSerialNum
AND productCategory.modelNum = itemSold.modelNum
AND transaction.salesAgreementDate > CURRENT_DATE - INTERVAL 24 MONTH
ORDER BY `Sales Date`
```

Sales Date	customerNum	Customer	Model #	Model Description	Manufacturer	Price
2019-06-06	5	Edna James	1	Paraglider - Cayenne	Skywalk	4070.00
2019-06-06	5	Edna James	1	Paraglider - Cayenne	Skywalk	4070.00
2019-06-30	4	Jim "Great" Dane	3	Hang Glider - Pulse	Airwave	5720.00
2019-07-02	6	Walt Etheridge	6	Hang Glider - Stalker	Aeros	5170.00
2019-07-02	6	Walt Etheridge	2	Hang Glider - Eagle 180	Aeros	5280.00
2020-05-30	7	Edward Haskell	7	Hang Glider - Stealth 151	Aeros	5940.00
2020-06-01	5	Edna James	1	Paraglider - Cayenne	Skywalk	4070.00
2020-06-21	1	Walter Jones	7	Hang Glider - Stealth 151	Aeros	5940.00
2020-06-21	8	Ed Wallace	10	Paraglider - Tequila	Skywalk	5500.00
2020-06-22	1	Walter Jones	11	Kiteboard - Wasp 3	Cabrinha	2860.00
2020-07-24	4	Jim "Great" Dane	11	Kiteboard - Wasp 3	Cabrinha	2860.00

Q08. Stage 1: For each product category, list the model number, description, and the total number of sales from that category.

```
CREATE OR REPLACE VIEW `Del 5 Q8a` AS  
SELECT productCategory.modelNum, productCategory.modelDescription, COUNT(itemSold.modelNum) AS `Total Sales`  
FROM productCategory, itemSold  
WHERE itemSold.modelNum = productCategory.modelNum  
GROUP BY productCategory.modelNum
```

Stage2: list the model number, description, and the total number of sales for the product category that has the highest number of sales.

```
SELECT* FROM `Del 5 Q8a`  
ORDER BY MAX(`Total Sales`)
```

modelNum	modelDescription	Total Sales
1	Paraglider - Cayenne	3

Q09. List the model number, description, and number of suppliers for those product categories that were supplied by multiple suppliers.

Part 1: For each product category, list the model number, description, order number, supplier number, and equipment price, sorting by model number and then by order number.

```
SELECT productCategory.modelNum, productCategory.modelDescription, productOrder.orderNum, supplier.supplierNum,
restockLineItem.lineItemEquipmentPrice
FROM productCategory,restockLineItem,productOrder, supplier
WHERE productCategory.modelNum=restockLineItem.modelNum
AND restockLineItem.orderNum=productOrder.orderNum
AND productOrder.supplierNum=supplier.supplierNum
ORDER BY productCategory.modelNum, productOrder.orderNum
```

modelNum	modelDescription	orderNum	supplierNum	lineItemEquipmentPrice
1	Paraglider - Cayenne	6	4	3700.00
1	Paraglider - Cayenne	7	2	3700.00
1	Paraglider - Cayenne	14	2	3700.00
2	Hang Glider - Eagle 180	7	2	4800.00
3	Hang Glider - Pulse	5	4	5200.00
4	Kiteboard - Reactor 2012	4	3	5000.00
5	Kiteboard - Spitfire 2010	1	11	2400.00
6	Hang Glider - Stalker	8	2	4700.00
7	Hang Glider - Stealth 151	9	9	5400.00
8	Hang Glider - Talon 140	2	1	4300.00
8	Hang Glider - Talon 140	15	10	4300.00
9	Hang Glider - Talon 160	3	10	4200.00
10	Paraglider - Tequila	11	4	5000.00
11	Kiteboard - Wasp 3	12	6	2600.00

Part 2: List the model number, description, and number of suppliers for those product categories that were supplied by multiple suppliers.

```
SELECT productCategory.modelNum, productCategory.modelDescription, COUNT(DISTINCT supplier.supplierNum) AS Total_supplier
FROM productCategory,restockLineItem,productOrder, supplier
WHERE productCategory.modelNum=restockLineItem.modelNum
AND restockLineItem.orderNum=productOrder.orderNum
AND productOrder.supplierNum=supplier.supplierNum
GROUP BY productCategory.modelDescription
HAVING COUNT(Total_supplier)>1
ORDER BY modelNum ASC
```

modelNum	modelDescription	Total_supplier
1	Paraglider - Cayenne	2
8	Hang Glider - Talon 140	2

Q10. Using a prepared statement, write a query that uses a placeholder that accepts a customer number via a variable and then lists all credit cards associated with that customer. Include customer number, customer name (concatenated like “First Last”), card number, card type (Visa, Mastercard, etc.), and expiration year and month, in the format mm/yyyy.

```
SET @a = 4;
PREPARE customerNum_in FROM
"SELECT customer.customerNum, CONCAT(customer.customerFirstName, ' ', customer.customerLastName) AS `Name`,\r\ncreditCard.creditCardNum,
creditCard.creditCardType, CONCAT(creditCard.creditCardExpirationMonth, '/', creditCard.creditCardExpirationYear) AS `Exp Date`\r\nFROM customer,
holdsAccount, creditCard\r\nWHERE customer.customerNum=holdsAccount.customerNum\r\nAND holdsAccount.creditCardNum=
creditCard.creditCardNum\r\nAND customer.customerNum LIKE ?";
EXECUTE customerNum_in USING @a;
DEALLOCATE PREPARE customerNum_in
```

customerNum	Name	creditCardNum	creditCardType	Exp Date
4	Jim "Great" Dane	4555554444333221	MasterCard	7/2024

Stored Procedures

sProc #1: Develop Stored Program

```
DELIMITER $$  
CREATE DEFINER='4407Group4'@'%' PROCEDURE `reordernecessary`()  
BEGIN  
    Alter Table productCategory  
    ADD COLUMN IF NOT EXISTS reorderNecessary BOOL NOT NULL;  
    SELECT productCategory.modelInventoryCount, productCategory.modelReorderQuantity, productCategory.reorderNecessary  
    FROM productCategory;  
    UPDATE productCategory  
    SET productCategory.reorderNecessary = 1  
    WHERE productCategory.modelInventoryCount <= productCategory.modelReorderQuantity;  
END$$  
DELIMITER ;
```

sProc #2: Convert Existing Stored Program

```

DELIMITER $$

CREATE DEFINER='4407Group4'@`%` FUNCTION `ValidateCreditCard`(`creditCardNum` VARCHAR(16)) RETURNS varchar(10) CHARSET latin1
BEGIN
    DECLARE counter INT;
    DECLARE sum INT;
    DECLARE number INT;
    DECLARE tmp INT;
    IF creditCardNum IS NOT NULL AND char_length(rtrim(creditCardNum))=16 THEN
        SET counter=1;
    END IF;
    WHILE counter<=char_length(rtrim(creditCardNum)) AND (substring(creditCardNum,counter,1))=1 REGEXP '^-[0-9]+$'
        DO
            SET counter=counter+1;
        END WHILE;
    SET sum=0;
    SET number=0;
    SET counter=char_length(rtrim(creditCardNum));
    WHILE counter>0
        DO
            IF counter>1
                THEN
                    SET tmp=(ASCII(substring(creditCardNum,counter-1,1))-48)*2;
                    IF tmp>9 THEN
                        SET sum=sum+tmp-9;
                    ELSE
                        SET sum=sum+tmp;
                    END IF;
                END IF;
                SET number=number+(ASCII(substring(creditCardNum,counter,1))-48);
                SET counter=counter-2;
            END WHILE;
            SET sum=(sum+number) % 10;
            IF sum=0 AND creditCardNum IS NOT NULL AND char_length(rtrim(creditCardNum))=16 THEN RETURN 'Valid'; ELSE RETURN 'Not Valid!';
        END IF;
    END$$
DELIMITER ;

```

Triggers

Trigger #1

```
CREATE TRIGGER `trigger1` BEFORE UPDATE ON `productCategory`
FOR EACH ROW BEGIN
IF NEW.modelInventoryCount <= OLD.modelReorderQuantity THEN
SET NEW.reorderNecessary = 1;
ELSE SET NEW.reorderNecessary = 0;
IF NEW.modelReorderQuantity > OLD.modelInventoryCount THEN
SET NEW.reorderNecessary = 1;
ELSE SET NEW.reorderNecessary =0;
END IF;
END IF;
END
```

Trigger #2 Part A

#According to MySQL documentation, we needed to seperate the Insert and Update triggers.

```
CREATE TRIGGER `updateTheCard` BEFORE UPDATE ON `transaction`
FOR EACH ROW BEGIN
IF NEW.salesAgreementPaymentType = 'Credit Card' THEN
SET @b= ValidateCreditCard(NEW.salesAgreementCardNumberUsed);
IF @b= 'Not Valid!' THEN
SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Error: The credit Card number is not valid';
END IF;
END IF;
END
```

Trigger #2 Part B

#According to MySQL documentation, we needed to seperate the Insert and Update triggers.

```
CREATE TRIGGER `InsertTheCard` BEFORE INSERT ON `transaction`
FOR EACH ROW BEGIN
IF NEW.salesAgreementPaymentType = 'Credit Card' THEN
SET @a= ValidateCreditCard(NEW.salesAgreementCardNumberUsed);
IF @a= 'Not Valid!' THEN
SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Error: The credit Card number is not valid';
END IF;
END IF;
END
```

Summary Conclusion

The company has shown that growth and opportunity are critical to the success and direction of the business. The database that we have designed focuses on this as a strategic advantage. The seamless design is easy to use for management and employees while still being an incredibly powerful addition to the tools of WRLRR. This agile and dynamic database will give the company confidence in current operations and freedom to expand with the peace of mind that our relational database system can grow and help launch to a new horizon.