

Hunting for Trends in Data Using Microsoft Access

Jennifer LaFleur

National Institute for Computer-Assisted Reporting

jenster@aol.com

When Bob Imrie, an Associated Press reporter in Wausau, WI, went searching for information about hunting accidents, he ended up right on target. The state filed hard-copy reports on every accident detailing information such as weather, topography and the number of pieces of orange the victim was wearing. Imry had no way of digging out trends from the voluminous hard-copy reports and he had never used a computer database. But with some help over the phone from some nerdy colleagues, he created his own database to look at trends in hunting accidents in Wisconsin.

Here's what his database ended up looking like:

CASE	DATE	TIME	COUNTY	AREA	W
	11/28/93	1030	Langlade	north	chest
	11/26/93	700	Dunn	centrl	toe
43	11/20/93	650	Door	north	arm
44	11/20/93	845	Marinette	north	foot
45	11/20/93	915	Sauk	south	leg
46	11/20/93	930	Crawford	south	arm
47	11/20/93	1100	Columbia	south	ankle
48	11/20/93	1145	Washburn	north	toe
49	11/20/93	1330	Vernon	south	hand
50	11/21/93	1000	Sheboygan	south	hand
51	11/21/93	1045	Dunn	centrl	toe
52	11/21/93	1155	Portage	south	thigh
53	11/21/93	1330	Waukesha	south	head
54	11/22/93	1045	Walworth	south	arm
55	11/23/93	1600	Sauk	south	head
56	11/25/93	1215	Manitowoc	centrl	leg
57	11/25/93	1230	Dunn	centrl	hand
<6	11/21/92	730	Shawano	centrl	toe
37	11/21/92	830	Waushara	centrl	arm
38	11/21/92	900	Barron	north	shuldr
39	11/21/92	1235	Marathon	centrl	head
40	11/21/92	1300	Waupaca	centrl	thigh

The record layout, which is the map to any database, is below. It shows the fields Imry created in the database, whether they were characters or numbers and the width of each field:

Field	Field Name	Type	Width	Description
1	CASE	Character	10	This is an identifier he created
2	DATE	Date	8	Date of accident
3	TIME	Numeric	4	Time of accident
4	COUNTY	Character	11	County of accident
5	AREA	Character	6	Region of state (he created the areas)
6	WOUND	Character	6	Part of body wounded
7	INJURY	Character	5	Severity of injury
8	TYPE	Character	2	Self-inflicted, or other (si=self-inflicted, sp=second person)
9	CAUSE	Character	30	Description of cause
10	SAGE	Numeric	2	Age of shooter
11	VAGE	Numeric	2	Age of victim
12	FIREARM	Character	7	Type of firearm
13	FACTION	Character	6	Faction of gun
14	ALCOHOL	Character	3	Alcohol involved
15	ALCOLEV	Numeric	5	Alcohol level
16	WEATHER	Character	11	Weather
17	TOPOGRO	Character	10	Typography
18	SEXPER	Numeric	2	Years of shooter's experience
19	VEXPER	Numeric	2	Years of victim's experience
20	SGRADUATE	Character	3	Shooter graduate of safety school
21	VGRADUATE	Character	3	Victim graduate of safety school
22	SSEX	Character	1	Sex of shooter
23	VSEX	Character	1	Sex of victim
24	GUNBRND	Character	13	Brand of gun
25	GUNGUAGE	Character	10	Guage of gun
26	TEMP	Numeric	2	Temperature
27	MUZDIS	Numeric	5	Muzzle distance
28	LAND	Character	3	Private or public
29	VORANGE	Numeric	1	Number of pieces of orange worn by victim
30	GUNSIGHT	Character	5	Open, scope or other
31	VACTIVITY	Character	10	Victim's activity
32	LOCATION	Character	6	Location of victim
33	PRECIP	Character	3	Precipitation

To bring up the data yourself, go to FILE/OPEN and in the data directory, open a file called Hunting.mdb.



In Access a database may contain several individual data sets called “tables.” To open the Deer table, select it and click OPEN. Access will open the database to a browse window.

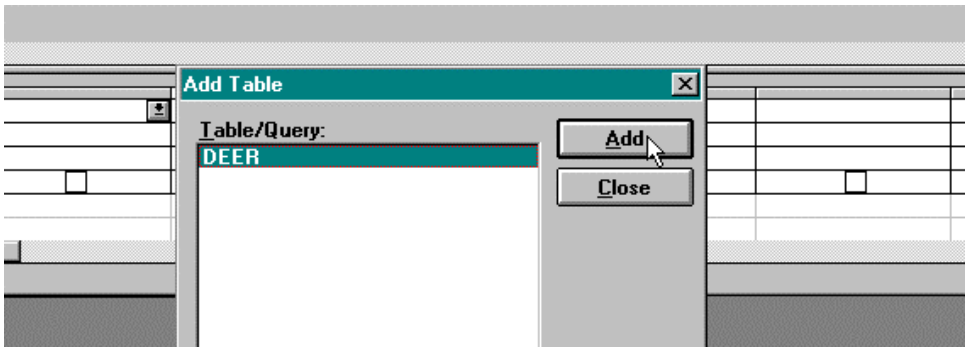
Interview your data.

Just as you do with people, you must get to know your data. Study it, look for patterns, look for special codes, upper case, lower case: all this will help when it’s time to ask questions.

Just looking at this database triggers many questions. But let’s start off with some basic queries. To bring up the query form, click on the Query tab in the main database window, and press the New button. You’ll then see the following window:

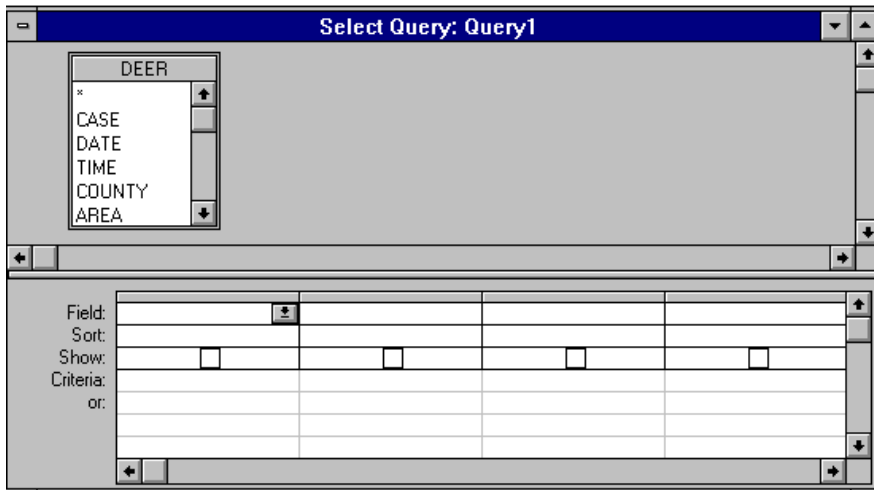


Click New Query. (Query wizard is a special tool to help you design special kinds of queries that we won’t be using). Access will bring you to a new window, and ask which tables you want to ask questions of. Choose the DEER table, and press the Add button ONCE!



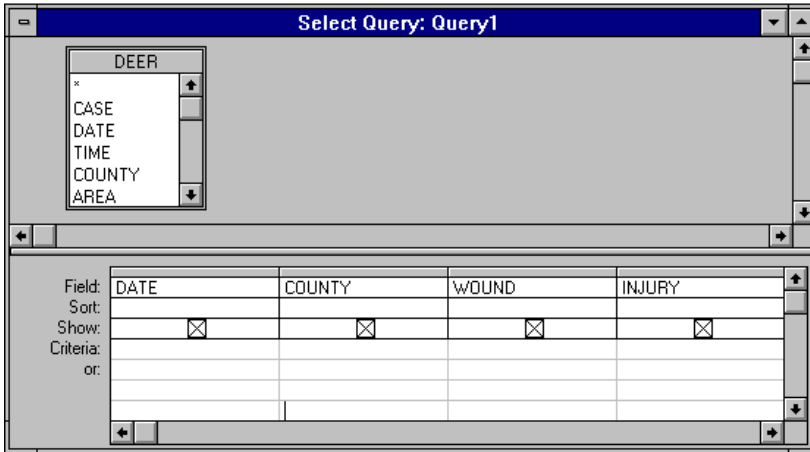
Then press the Close button.

Let's review the query form because it's the place you'll hang out the most in Access:



1. **Field...** Select Fields: this is where you pick the pieces of information you want to see
2. **Sort...** Put in order: do you want the most at the top, the oldest at the top?
3. **Criteria:** Pick the records you want. This allows you to limit filter out just some of your records.

These steps may seem overly fussy, but if you follow them in more complex queries you can save yourself some trouble. Let's practice the first step by doing a query to show the following fields: DATE, COUNTY, WOUND and INJURY. Do this by dragging each of these field names from the list of fields in the upstairs window area down into the downstairs window area. (You can also double-click on the name of the field.) When you choose a field, Access automatically adds an X to the checkbox. This means it will show in your result.



Click  from the button bar to run your query.

Your result should look like this:

Select Query: Query1				
DATE	COUNTY	WOUND	INJURY	
11/28/93	Langlade	chest	minor	
11/26/93	Dunn	toe	minor	
11/20/93	Door	arm	minor	
11/20/93	Marinette	foot	major	
11/20/93	Sauk	leg	major	
11/20/93	Crawford	arm	minor	
11/20/93	Columbia	ankle	major	
11/20/93	Washburn	toe	minor	
11/20/93	Vernon	hand	minor	
11/21/93	Sheboygan	hand	minor	
11/21/93	Dunn	toe	major	
11/21/93	Portage	thigh	minor	
11/21/93	Waukesha	head	major	
11/22/93	Walworth	arm	minor	
11/23/93	Sauk	head	fatal	
11/25/93	Manitowoc	leg	major	
11/25/93	Dunn	hand	minor	
11/21/92	Shawano	toe	minor	

Record: 1 of 255

The second step to any query is to sort it. In this case, we could organize the records by DATE. To go from your results back to your query window, hit the VIEW/QUERY DESIGN. Or use these buttons to toggle between the answer and the query form:



^ ^
design table

To sort the data click in the box right below the field name. In this case, click below date. The query form also says “sort” on the left end of this line.

Field:	DATE	C
Sort:	<input type="button" value="v"/>	
Show:	Ascending	
Criteria:	Descending	
or:	(not sorted)	

Pick the kind of sort you want.

Run the query. Your answer should appear in descending date order (or the most recent first).

Limit the records

Step three of any query lets you take a bite out of your data. Say you didn't really care about all the deer hunting accidents in Wisconsin, you just wanted to see them for Jackson County. That's where the CRITERIA line of the query comes in handy. Click on the criteria line, in the COUNTY field and type: Jackson

Access is sometimes case-sensitive, so if Jackson is upper case -- type it that way in your query.

Other comparisons:

If you're screening your data, but you don't know the whole word in a field, Access will let you estimate using a *, which is called a "wildcard." A wildcard stands for any combination of letters or numbers.

Typing this in the cause field would pull out all causes beginning with Careless:

Careless* → Access will change this to Like "Careless*"

This would pull out all causes containing the word wife:

wife

Let's get Boolean

Sometimes you want more from a selection criteria than just one county. You might want the records for Jackson and Door counties. That's where OR comes in handy. Type OR between each item. In this case, type Jackson OR Door in the Criteria:

Field:	DATE	COUNTY	WOUND	INJURY
Sort:	Descending			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		Jackson or Door		
or:				

Now run the query again. You told Access you wanted all the records where the COUNTY was either Jackson or Door. This might be useful if you covered a multiple county area. Using the OR to connect criteria is called Boolean logic. Another Boolean word is AND. AND is more restrictive than OR: with AND both lines must be true in order for a record to be selected. We could use the AND if we wanted all causes containing both the word wife AND careless.

CAUSE
<input checked="" type="checkbox"/>
wife and Careless*

This example also uses the *. This is a “wildcard” that stands for any combination of letters and numbers in Access. Careless* stands for any cause starting with Careless. *wife* stands for any cause containing the word wife.

You also can use criteria in two fields at the same time. For example, we might want just self-inflicted accidents from Jackson County. Click in the field area below COUNTY and type Jackson right over what you have. Then click in the TYPE column and put **si** in the same line as Jackson for self-inflicted.

Field:	DATE	COUNTY	WOUND	INJURY
Sort:	Descending			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		"Jackson" Or "Door"		"si"
or:				

If you wanted Jackson or Door county, OR self-inflicted wounds, you would move the "si" down to the line that says, "or".

Practice Exercises

1. Do a query to show all the injuries occurring physically below, and including, the ankle. *Hint: here's a list of all possible wounds: abdomen, ankle, arm, back, butt, chest, ear, eye, face, finger, foot, groin, hand, head, hip, knee, leg, neck, scrotum, shoulder, stomach, thigh, thumb, toe.*
2. Do a query to show all self-inflicted accidents caused by shooters older than 50.
3. How old was the oldest victim in Jackson County? What wound did they have? How old was the youngest? What wound?
4. How many accidents occurred overall on the first day of deer hunting season last year (November 19, 1994)?

Answers

1. Do a query to show all the injuries occurring physically below, and including, the ankle.

Field:	DATE	COUNTY	WOUND	INJURY
Sort:	Descending			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			foot or ankle or toe	
or:				

Select Query: Query1				
	DATE	COUNTY	WOUND	INJURY
▶	11/27/94	Green	foot	minor
	11/22/94	Dane	ankle	major
	11/20/94	Buffalo	foot	minor
	11/20/94	Chippewa	toe	major
	11/19/94	Waupaca	foot	minor
	11/26/93	Dunn	toe	minor
	11/21/93	Dunn	toe	major
	11/20/93	Marinette	foot	major
	11/20/93	Columbia	ankle	major
	11/20/93	Washburn	toe	minor
	11/29/92	Door	foot	major
	11/25/92	Portage	toe	major
	11/23/92	Adams	foot	major
	11/22/92	Langlade	toe	major
	11/22/92	Calumet	foot	major
	11/21/92	Shawano	toe	minor
	12/5/91	Barron	foot	major
	11/30/91	Jackson	foot	major

Record: 1 of 46

The answer contains 46 injuries.

2. Do a query to show all self-inflicted accidents caused by shooters older than 50.

TYPE	CAUSE	SAGE
DEER	DEER	DEER
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
"si"		>49

Select Query: Query1							
	CASE	DATE	COUNTY	WOUND	INJURY	TYPE	
▶	51	11/24/91	Jackson	abdomn	major	si	ca
	11	11/17/90	Grant	toe	major	si	los
	35	11/24/90	Oneida	arm	major	si	los
	20	11/19/89	Chippewa	face	minor	si	ca
	8	11/21/88	Adams	thigh	major	si	ca
	8	11/21/87	Columbia	abdomn	fatal	si	vic
	17	11/22/87	Door	hand	major	si	un
	28	11/24/87	Adams	leg	major	si	fire
	19	11/22/94	Dane	ankle	major	si	fire
	*						

Record: 1 of 9

3. How old was the oldest victim in Jackson County? What wound did they have? How old was the youngest? What wound?

Field:	VAGE	COUNTY	WOUND
Sort:	Descending		
Show:	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		Jackson	
or:			

Select Query: Query1						
	CAS	DATE	COUNTY	WOUND	INJURY	VAGE
▶	43	11/27/87	Jackson	thigh	minor	62
	51	11/24/91	Jackson	abdomn	major	55
	57	11/25/91	Jackson	chest	major	42
	17	12/22/94	Jackson	thigh	minor	25
	27	11/26/94	Jackson	leg	major	20
	34	11/23/90	Jackson	arm	major	20
	67	11/30/91	Jackson	foot	major	15
	19	11/26/88	Jackson	foot	major	14
	*					

Record: 1 of 8

Oldest: 62, thigh Youngest: 14, foot

4. How many accidents occurred overall on the first day of deer hunting season last year (November 19, 1994)?

Field:	DATE	COUNTY	WOUND	INJURY
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	11/19/94			
or:				

(Note: Access will put hashmarks around a date. That's good: #11/19/94#)

DATE	COUNTY	WOUND	INJURY
#11/19/94	Chippewa	neck	fatal
11/19/94	Buffalo	shuldr	major
11/19/94	Outagamie	thigh	minor
11/19/94	Marathon	leg	major
11/19/94	Calumet	abdomn	major
11/19/94	Green Lake	chest	fatal
11/19/94	Waupaca	foot	minor
11/19/94	Washburn	abdomn	fatal
11/19/94	Iowa	arm	major

9 injuries

Calculations

Just as in spreadsheet, sometimes you want to do calculations between columns or “fields.” Because a database has no cell addresses, you have to use the names of fields. In Access you type a calculation right into the field name line in a fresh column.

Go to a new column on the query form. Let’s say you wanted to find cases where the shooter is older than the victim and compute their age difference. Calling the new calculated field AgeDiff, your would type this in the field row:

AgeDiff: = sage-vage

Then in the criteria row, you'd add:

>0 (greater than zero)

VAGE	DiffAge: [SAGE]-[VAGE]
DEER	
Descending	
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	>0

(Access will put the little brackets around the field names for you.)

Then click the Run button.

Data Groupies

The fourth step of any query is the most powerful: **GROUP BY** lets you summarize your entire database. For example, in our deer database, we might want to know what the most common injury is. Before you do the query, think about what your result might look like. (I always think about what the chart I might print in the newspaper would look like.) In this case I would have two columns: the wounds and the number of times each wound occurred. My headline would be: What Gets Wounded in All Hunting Accidents. Let's step through the four query steps:

1. Pick the fields: Select wound as a field in the first two columns. You need to pick it twice: once for the names of the wounds and another column to count the wounds. Because we're summarizing the data, we need to click the summarize button:



Access will add a second line to the query form. With the word "total" along the left. Access also will put **GROUP BY** in each column. If you click on **GROUP BY**, Access will bring up a list of summary functions:

Field:	WOUND	
Total:	Group By	
Sort:		Group By
Show:	<input checked="" type="checkbox"/>	Sum
Criteria:		Avg
or:		Min
		Max
		Count
		StDev
		Var

Choose **COUNT** in the second column:

Field:	WOUND	WOUND
Total:	Group By	Count
Sort:		
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		
or:		


2. Sort: If we want to know the most common wound, we'd probably want the highest count at the top. Click the **Sort** check box below **COUNT** and **WOUND**. Select descending.

Note: You can use this same technique for computing average, minimum, maximum and sum.

3. Pick the records: Think of this step as the headline on your chart in the paper. In this case our headline is for ALL hunting accidents, not just self-inflicted, not just old hunters, but everyone. So here, we have no selection criteria, because we are not narrowing our population.

4. Group by: Keep one thing in mind: *Access isn't as smart as you are.* As Access counts all the wounds, it doesn't know to put all the butt wounds in one pile and all the knee wounds in another pile. We have to help. The same technique would apply if a giant truck filled with vegetables backed into the classroom right now and the driver asked us to count up the different types of vegetables. What would be the first thing you'd do? OK, OK, leave the room. Well, I probably would too. But we also might put all the carrots in one pile, all the onions in another, and all the potatoes in another so we could go back and count them. This is why WOUND must appear twice in the query form:

Field:	WOUND	WOUND	
Total:	Group By	Count	
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Criteria:			
or:			

If everything looks okay, click the  button to run the query.

WOUND	CountOfWOUND
leg	41
arm	29
chest	26
thigh	25
foot	25
head	23
toe	19
hand	18
abdomn	9
shuldr	7
back	6
neck	5
face	3
butt	3
knee	3
hip	3
ankle	2
finger	2
ear	1
eye	1
scrotm	1

Record: 11 of 24

Leg injuries are most common, with 41 shot up.

Now, let's make a minor change to the query. Hit **VIEW/QUERY DESIGN** to throw away your answer. Say, we didn't care about all wounds, let's change that headline to: **Most Common Types of Self-Inflicted Wounds**. This would require changing just one thing in the four query steps. Often, we try to make this harder than it should be. Remember, if you're just changing the headline, all you need to do is change your criteria: **Step 3**. Add **TYPE** as a field and type **si** in the criteria box. Change the Total line to "Where" (all the way at the bottom of the drop-down list).

*Note: when you use a criteria for a summary query, the **GROUP BY** line should be changed to **WHERE**. This isn't intuitive, but Access has to do something with that **TOTAL** line.*

Field:	WOUND	WOUND	TYPE
Total:	Group By	Count	Group By
Sort:		Descending	Max
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Count
Criteria:			StDev
or:			Var
			First
			Last
			Expression
			Where

Answer: 19 foot injuries

Practice Exercises

1. What was the most common cause of hunting accidents? How many?
2. Which brand of gun is involved in the most hunting accidents? How many?
3. Which county had the most accidents? How many?
4. What is the average age for the shooters in butt-injury accidents?

Answers

1. What was the most common cause of hunting accidents? How many?

Field:	CAUSE	CAUSE
Total:	Group By	Count
Sort:		Descending
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		
or:		

Answer: victim in line of fire, with 48

2. Which brand of gun is involved in most hunting accidents? How many?

Field:	GUNBRND	GUNBRND	
Total:	Group By	Count	
Sort:		Descending	
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Criteria:			
or:			

Answer: Remington, with 79

Note: What additional reporting would you have to do to use this answer?

3. Which county had the most accidents? How many?

Field:	COUNTY	COUNTY
Total:	Group By	Count
Sort:		Descending
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		
or:		

Answer: Marathon, with 9

4. What is the average age for the shooters in butt-injury accidents?
(Hint: you don't want blanks or 0s)

Field:	SAGE	SAGE	WOUND	WOUND
Table:	DEER	DEER	DEER	DEER
Total:	\where	Avg	\where	Group By
Sort:		Descending		
Show:	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	>0		"butt"	

Answer: 16.5

Multi-Groupies

Another approach to the counting up self-inflicted wounds might be to group by both WOUND and TYPE.

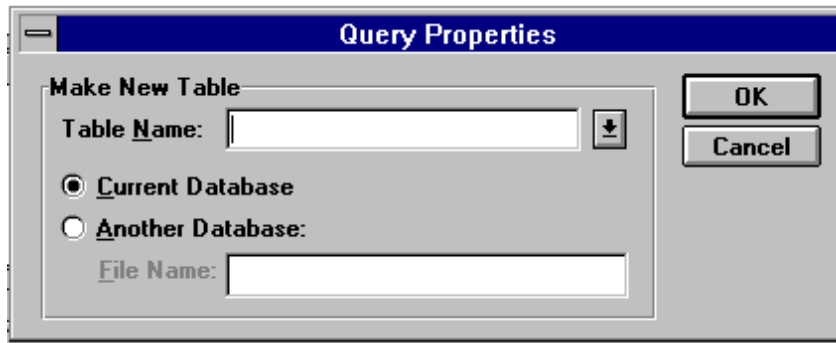
The steps:

1. Select WOUND, TYPE and COUNT of ??????. It can be difficult to decide what to count, but here's a hint: count will total the different combinations of everything you group by. But it will only count records that are filled in for the field you select. In our database, WOUND is always filled out. So let's just ask for COUNT in a new WOUND column.
2. Order by: Do it so the biggest count is on top.
3. Selection criteria: I know, I know, we just pulled out self-inflicted. But now we want both self-inflicted, plus the other types. So remove the selection criteria.
4. Group by: group by both WOUND and TYPE.

Run the query. Your answer will show all the possible combinations of TYPE and WOUND and count them. Answer: 30 leg injuries by second person

Field:	TYPE	WOUND	WOUND	
Table:	DEER	DEER	DEER	
Total:	Group By	Group By	Count	
Sort:			Descending	
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Criteria:				

A note about output: On the main query form, the output is usually set to browse. This means that your answer pops up in a window. You can change this output to save your answer to a separate file. To do this go to QUERY on the main menu and select MAKE TABLE, Access will then prompt you for the name of the table. Then click OK.




Additional Exercises

1. Do accidents of certain causes of accidents tend to have higher or lower rates of orange-wearing? (Hint: you'll need to create a new file: call it ORANGE) and do a second query on ORANGE.DBF)
2. Did more accidents occur in bad weather?
3. What was the most common cause for fatal accidents? Were most of those inflicted by someone else or by the shooter?

Answers

1. Do accidents of certain types tend to have higher or lower rates of orange-wearing? This is a two-part answer. What you really want in your end result is a list of pieces of orange per accident by cause.

Tell Access you want to do a MAKE TABLE query and call the table ORANGE.

Field:	CAUSE	CAUSE	VORANGE
Total:	Group By	Count	Sum 
Sort:		Descending	
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			
or:			

CAUSE	CountOfCAUSE	SumOfVORANGE
victim in line of fire	48	107
victim out of sight o	41	97
careless handling-tr	16	38
loaded firearm in ve	12	23
shooter stumbled ar	11	28
Loading firearm	9	20
ricochet-bullet thru	8	17
victim mistaken for	8	17
defective firearm	5	15
unloading firearm	5	11
ricochet	3	8
ricochet-tree	3	8
careless handling-d	2	6
victim mistaken for	2	4
careless handling-s	2	5
careless handling-tr	2	4
trigger caught on ol	2	4
victim in car-stray b	2	2
firearm fell-tree invo	2	3
firearm fell, insecure	2	5
careless handling a	2	5

Next, do another query on ORANGE to compute sum_vorang/cnt to get orange per accident type:

Avg: [sumofvorang]/[countofcause]	
Descending	<input checked="" type="checkbox"/>

Select Query: Query1	
CAUSE	Avg
careless handling-setting down	3
ricochet-corn	3
ricochet-corn field	3
ricochet	3
nonhunter watching, hit by slg	3
loading firearm-gloves	3
defective firearm-gun blew up	3
victim stumbled and fell	3
ricochet-scrub oak	3
careless handling-thick gloves	3
defective firearm	3
careless handling-raising gun	3

3. Did more accidents occur in bad weather? What other fields could you use for this query?

Field:	WEATHER	WEATHER
Total:	Group By	Count
Sort:		Descending
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		
or:		

WEATHER	CountOfWEATH
clear	125
cloudy	95
partly cldy	28
unknown	4
ptly cloudy	1
	0

Answer: no, most accidents occurred in clear weather.

4. What was the most common cause for fatal accidents? Were most of those inflicted by someone else or by the shooter?

Answer: “victim in line of fire” with 11 (most were inflicted by a second person)

Field:	CAUSE	CAUSE	TYPE	INJURY	↑
Total:	Group By	Count	Group By	Where	
Sort:		Descending			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Criteria:				fatal	
or:					↓
	←				→

CAUSE	CountOfCAUSE	TYPE
victim in line of fire	10	sp
victim out of sight of shooter	3	sp
victim mistaken for deer	2	
victim out of sight of shooter	1	
victim in line of fire-dark	1	sp
victim in line of fire	1	
victim in car-stray bullet	1	
victim fell from tree	1	si
shooter stumbled and fell	1	sp
firearm fell-tree involved	1	si
careless handling-tree involv	1	si
careless handling-tree involv	1	si
careless handling-finger on tr	1	sp
careless handling-dropped gu	1	si
bullet went thru home	1	

Good job! You're done hunting for trends in data.