

DATA ANALYSIS THE DATA.TABLE WAY

The official Cheat Sheet for the <u>DataCamp</u> course

General form: $DT[i, j, by] \rightarrow \Psi$ \rightarrow "Take DT, subset rows using i, then calculate j grouped by by"

	CREATE A DATA TABLE				
Create a	library(data.table)	> DT			
data.table	set.seed(45L)	V1	V2	V3	V4
and call it DT.	<pre>DT <- data.table(V1=c(1L,2L),</pre>	1: 1	А	-1.1727	1
	V2=LETTERS[1:3],	2: 2	В	-0.3825	2
	V3=round(rnorm(4),4),	3: 1	С	-1.0604	3
	V4 =1:12)	4: 2	А	0.6651	4
		5: 1	В	-1.1727	5
		6: 2	С	-0.3825	6
		7: 1	А	-1.0604	7
		8: 2	В	0.6651	8
		9: 1	С	-1.1727	9
		10: 2	А	-0.3825	10
		11: 1	В	-1.0604	11
		12: 2	С	0.6651	12

	SUBSETTING ROWS USING 1					
What?	Example	Notes	Output			
Subsetting rows by numbers.	DT[3:5,] #or DT[3:5]	Selects third to fifth row.	V1 V2 V3 V4 1: 1 C -1.0604 3 2: 2 A 0.6651 4 3: 1 B -1.1727 5			
Use column names to select rows in i based on a condition using fast automatic indexing. Or for selecting on multiple values: DT[column %in% c("value1", "value2")], which selects all rows that have value1 or	DT[V2 == "A"]	Selects all rows that have value ${f A}$ in column V2.	V1 V2 V3 V4 1: 1 A -1.1727 1 2: 2 A 0.6651 4 3: 1 A -1.0604 7 4: 2 A -0.3825 10			
value2 in column.	DT[V2 %in% c("A","C")]	Select all rows that have the value ${\bf A}$ or ${\bf C}$ in column ${\bf V2}.$	V1 V2 V3 V4 1: 1 A -1.1727 1 2: 1 C -1.0604 3 7: 2 A -0.3825 10 8: 2 C 0.6651 12			

	What?		MANIPULAT Example	ING ON C	OLUMNS IN J Notes		Output
Note of a constraint of a con	Select 1 column in j.	DT[,	· .	Colum			[1] "A" "B" "C" ".
<form> App of the set of the s</form>	delect several columns in j.	DT[,	.(V2,V3)]				V2 1: A -1. 2: B -0.
	() is an alias to list(). If .()	is used	, the returned value is a data	.table.If .() is not used, the result is a ve	ector.	
And the state of the state	all functions in j.						
Note of the section	omputing on several columns.	DT[,	.(sum(V1),sd(V3))]	elemer	its of column $\mathbf{V1}$ and the stand	ard	
Note				The sa	me as above, but with new nar		
		DT[,	.(V1, Sd.V3 = sd(V3))]				1: 1 0.7634 2: 2 0.7634
Image Dec. Dec. Dec. Dec. Dec. initial is transmission 27, -78,5-27 - ar (270) - ar (270,270) The mean state is difference provide of the state is an initial is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is a state is difference provide of the state is difference		DT[,	plot(V3)	Print	column $\mathbf{V2}$ and plot $\mathbf{V3}$.		12: 2 0.7634 [1] "A" "B" "C" ". "B" "C"
				G _{J BY} G			
		DT[,.		1]	Calculates the sum of V4 , f	or every grou	p in V1 V4.
		DT[,.	(V4.Sum = sum(V4)),by=.	(V1,V2)]	The same as above, but for	every group i	1:1 A 2:2 B
Note of the set of th	all functions in by .	DT[,.	(V4.Sum = sum(V4)), by=s	ign (V1-1)]	Calculates the sum of V4 f	or every grou	4: 2 A 5: 1 B 6: 2 C
Name of the stand of the s				5	sign (V1-1).		1: 0 2: 1
NameNo. dots when a product on contract on the product on		DT[,.		1))]			1: 0
Auto Autom Auto Autom Auto Autom Auto Autom Auto Autom Auto Autom Note Same Auto auto Auto Auto Auto Auto Auto Auto Auto A		DT [1:	5,.(V4.Sum = sum(V4)),b	y=V1]			s. 1:1
	mber of observations of each	DT[,.	N,by=V1]			for every grou	1:
	oup.	ADD	ING/UPDATING COLU	IMNS BY R	EFERENCE IN J USING	:=	2:
International Section 1 International Section 2 International			Example		Notes		
Entropy of a proper by a proper	ference using : = in one line. atch out: extra assignment	.,		, , , , , , , , , , , , , , , , , , ,		Column V 2 to [1	1 went from: [1] 1
Control (Control (Contro))))))	lding/updating several	(roun	d(exp(V1),2), LETTERS		. and $\mathbf{V2}$ are updated by what :	is Returns t Column V Column V	71 changed as above. 72 went from: [1] "A
	sing functional :=.	DT[,	':=' (V1 =	Another wa	ay to write the same line as	"D" "E"	"F" "D" "E" "F" .
mass starting in production of the starting in		1	round(exp(V1),2),	above this comments	one, but easier to write side-by-side. Also, when [] is	one, but t screen bee	he result is printed to cause of the [] at the
Process (Process		DT[,	V1 := NULL]			Returns t	he result invisibly.
mark and a structure is structure is structure is structure is structure is a structure i	emove several columns	DT[,	c("V1","V2") := NULL]	Removes co	blumns $V1$ and $V2$.	Returns t	he result invisibly. Co
Number of the source	rap the name of a variable						
<text> Note Control Description of a read biology of a read bioread biology of a read biology of a read biology of a re</text>	renthesis to pass the contents	DT[,	Cols.chosen := NULL]			Column w	with name Cols.chos
NNC Origin (1999) Origin (1999) <thorigin (1999)<="" th=""> Origin (1999)</thorigin>		DT[,	(Cols.chosen) := NULL]			Returns t	he result invisibly.
recently in the result of t			INDE>	ING AND	KEYS		
a hay binary anymeny anymeny any any any any any any any any any a	e setkey() to set a key on a D		•	A key is set			Returns results
ender sees. P(1 = (1, 1, 1, 1)) P(1 = (1, 1, 1)) P(1 = (1, 1, 1)) P(1 = (1, 1, 1)) P(1 = (1, 1, 1)) Barraw all de wow where whe wohen NVP, here here wohen NVP, here here wohen NVP, here here wohen NVP, here here here here here wohen NVP, here here here here here here here her	ecified by reference.						
$ \frac{2}{2} \left[\frac{1}{2} $		ames	DT["A"]				1: 1 A -1.1727 2: 2 A 0.6651
$ \begin{array}{c} $			DT[c("A","C")]			n (V2) has the	V1 V2 V3 1: 1 A -1.1727 2: 2 A 0.6651
$ \frac{1}{2} (1, 2^{n}, m) t = \frac{1}{2} (1, 2^{$	ne mult argument is used to con	ntrol	<pre>DT["A", mult ="first"]</pre>			e value ${f A}$ in	8: 2 C 0.6651 V1 V2 V3
$\frac{1}{2} = \frac{1}{2} = \frac{1}$			<pre>DT["A", mult = "last"]</pre>			e value A in	
$ \begin{array}{c} 1 \\ \mbox{rescale} (1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$	ntrol what happens when a valu ecified in i has no match in the	ue rows	DT[c("A","D")]	Returns all value A or D	the rows where the key column		v1 v2 v3 1: 1 A -1.1727 2: 2 A 0.6651
$ \frac{1}{2} = 0 $ $ \frac{1}{2} = 0$	anged to 0.	be					4:2 A -0.3825
$ \frac{1}{2} \sum_{i=1}^{2} \sum_{j=1}^{n} \sum_{i=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{i=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{i=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{i=1}^{n} \sum_{i=1}$	turned for that non-matched rov	w of i .		value A or E because of tl	D. Value D is not found and not ne		1: 1 A -1.1727 2: 2 A 0.6651 3: 1 A -1.0604
$\frac{1}{2} \text{ for the set of use y_{2} = 25211 \frac{1}{2} \text{ cm}(\sqrt{3}, \frac{1}{2}y_{2}, u_{2}(2, 0) \frac{1}{2} \text{ cm}(\sqrt{3}, \frac{1}{2}y_{2}, u_{2}(2, 0)) \frac{1}{2} \text{ cm}(\sqrt{3}, u_{2}(2, 0$						e rows of the	
y aunder of columns can be set at y using active (1). This way reves he selected on 2 keys which is on ujoin. DT[. (2, *C*)] DT[. (2, *C*)]			DT[c("A","C"),	Returns one V2 that hav another sum	sum of column V4 for the row e value A , and		1: <i>I</i>
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	y using setkey (). This way ro	ws		Sorts by colu group of colu	umn V1 .		invisibly.
$ \left(\left(\frac{1}{2} + \frac{1}{2}$				(column V1) V2).	and the value ${\boldsymbol C}$ for the second	l key (column	1: 2 C -0.3825 2: 2 C 0.6651
ADVANCED DATA TABLE OPERATIONS Whet? Description Notes Output teontains the number of rows or the at rw. Usable in 1: D7[.N] Returns the pendlimate row of the data.table. VI V2 V3 1: 1 n -1,0604 1 is an alias to 1 is t (1) and means esame. The .() notation is not edued when there is only one item in 'or j. Usable in (: D7[., V2, V3]) for DF[.list(V2, V3]] Returns the number of rows. [1] Usable in (: D7[., 0] Usable in (: D7[., (V2, V3])] for DF[.list(V2, V3]] Returns the result of j, grouped by all possible combinitions of groups specified in by. V1 V2 2: n -0.3 3: c -1.0 10 is a data.table and holds all the laws of all columns, except the one effold in by. D7[, print(.SD], by=V2] To look at what .SD contains. 441 of .S0 (output too long to display here) D7[, print(.SD], by=V2] Selects the first and last row grouped by column V2. V2 VI V3 1: k 1 -1.127 2: k 2 -0.3825 1: a 1 -1.0407 2: k 2 -0.3825 1: a 2 -0.3825 1: a 1 -0.0407 1: k 1 -1.027 2: k 2 -0.3825 1: a				(column $V1$)	and within those rows the val		1: 2 A 0.6651 2: 2 A -0.3825 3: 2 C -0.3825
contains the number of rows or the traw. Usable in :: DT [. +1] Returns the penultimate row of the data.table. VI V2 V3 is an allas to list () and means is same. The . () notation is not odd when there is only one item in or j. Usable in j: DT [. , N] Returns the number of rows. [1] 12 Usable in by: DT [. , N] Usable in j: DT [. , V2, V3] *CC Columns V2 and V3 are returned as a DT [.list (V2, V3]] VI V2 odd when there is only one item in or j. Usable in by: DT [. mean (V3), by=V2] Returns the result of j. grouped by all possible combinations of groups specified in by. VI V2 bis a data.table and holds all the loss of all columns, except the one celled in by. If reduces gramming time but keeps dabilitysb is only accessible in j. DT [. print (.SD), by=V2] To look at what .SD column V2. V2 VI V3 Drs is used together with .SD, to clip (1, lapp1y (.SD, sum), by=V2] Calculates the sum of all columns in .SD col j: a us base of the columns of .SD to clip a subset of the columns of .SD to clip a subset of the columns of .SD to clip a subset of the columns of .SD to clip a subset of the columns of .SD to clip a subset of the columns of .SD to clip a subset of the columns of .SD to clip a subset of (V1, SD, sum), by=V2, SD col s = (.CV3^m, V4^1)] Same as ubove, but only for columns V3 and V4 of .SD. SI to clip a sub (V1, V4, Sum = sum (V4)), by=V2, SD col s = pasted (VT, J, J44) CHAINING HELPS TACK EXPRESSIONS TOGETHER AND AVOID (UNNECESSARY) INTERMEDIATE ASSIGNMENTS V2 V1 V4 Var				TA TABLE			
Number of the construction Usable in j:DT[., N] Returns the number of rows. [1] 12 1 is an alias to 11st () and means esame. The . () notation is not edded when there is only one item in or j. Usable in j:DT[., (V2, V3)] Columns V2 and V3 are returned as a data.table. 1: A -1.1 (2: S -0.3) (2: C -1.0) (2:	contains the number of rows or	r the	· · ·		eturns the penultimate row of t	che	v1 v2 v3
$\begin{array}{c} \text{ added when there is only one item in \\ \text{or j.} \\ \text{ Usable in by: DT[, mean(V3), \\ by=.(V1,V2)] \\ \text{ usable in by: DT[, mean(V3), \\ by=.(V1,V2)] \\ \text{ by=.(V1,V2)] } \\ \text{ add there is only one item in \\ by=.(V1,V2)] \\ \text{ by=.(V1,V2)] \\ \text{ specified in by. } \\ \text{ Returns the result of j, grouped by all } \\ \text{ possible combinations of groups } \\ \text{ specified in by. } \\ \text{ specified in by. } \\ \text{ add there is only one item in } \\ \text{ by=.(V1,V2)] \\ \text{ specified in by. } \\ \text{ by=.(V1,V2)] \\ \text{ by=.(V1,V2)] } \\ \text{ by=.(V1,V2)] \\ \text{ specified in by. } \\ \text{ by=.(V1,V2)] \\ \text{ specified in by. } \\ \text{ by=.(V1,V2)] \\ \text{ by=.(V1,V2)] } \\ \text{ base of all columns, except the one } \\ \text{ contains. } \\ \text{ box of all columns, except the one } \\ \text{ contains. } \\ \text{ box of all columns, except the one } \\ \text{ contains. } \\ \text{ box of all columns, except the one } \\ \text{ contains. } \\ \text{ box of all columns, except the one } \\ \text{ column V2. } \\ \text{ column V2. } \\ \text{ selects the first and last row grouped by } \\ \text{ v2 V1 } \\ \text{ v3 } \\ \text{ is a 1 - 1,127 } \\ \text{ is a 1 - 1,127 } \\ \text{ is a 2 - 0,3825 } \\ \text{ is B 2 - 0,3825 } \\ \text{ is B 1 - 1,0604 } \\ \text{ is C 1 - 1,0604 } \\ \text{ is C 1 - 1,0604 } \\ \text{ is C 2 - 0,3825 } \\ \text{ is B 1 - 1,0604 } \\ \text{ is C 1 - 1,0604 } \\ \text{ is C 2 - 0,3825 } \\ \text{ is B 2 - 0,3825 } \\ \text{ is C 1 - 1,0604 } \\ is $) is an alias to list () and me		Usable in j: DT[,.(V2,V3)]	Re #or Co	eturns the number of rows. Jumns ${f V2}$ and ${f V3}$ are returned] 12 V2
by =. (V1, V2)] by =. (V1, V2)] column V2. by =. (V1, V2) column V2. by =. (V2, V1, V3) column V2. column V2.	eded when there is only one ite					hy all	2: B -0.3 3: C -1.0
D is a data.table and holds all the lues of all columns, except the one ecified in by. It reduces ogramming time but keeps adabilitySD is only accessible in j. DT[,.SD[c(1,.N)], by=V2] adabilitySD is only accessible in j. DT[, lapply(.SD, sum), by=V2] DT[, lapply(.SD, sum), by=V2] DCls is used together with .SD, to DT[, lapply(.SD, sum), by=V2, used in j. DCols = paste0("V", 3:4)] DT[, lapply(.SD, sum), by=V2, Addition and V4 of .SD. DCls = paste0("V", 3:4)] DT[, lapply(.SD, sum), by=V2, Addition and V4 of .SD. DCls = paste0("V", 3:4)] DT[, lapply(.SD, sum), by=V2, Addition and V4 of .SD. DCls = paste0("V", 3:4)] DT[, lapply(.SD, sum), by=V2, Addition and V4 of .SD. DCls = paste0("V", 3:4)] DT[, lapply(.SD, sum), by=V2, Addition and DT[, lapply(.SD, sum), by=V2, Addition and DT[, lapply(.SD, sum), by=V2, Addition call. DT[, lapply(.SD, sum), by=V2, Addition and DT[, lapply(.SD, sum), by=V2, Addition call. DT[, lapply(.SD, sum), by=V2, Addition and DT[, la				ро	ssible combinations of groups	- , un	1: 1 A -1.11 2: 2 B 0.14 3: 1 C -1.11 4: 2 A 0.14 5: 1 B -1.11
Inters of all columns. here) here) Inters of all columns. here) ogramming time but keeps DT[,.SD[c(1,.N)], by=V2] Selects the first and last row grouped by V2 V1 V3 adabilitySD is only accessible in j. DT[,.SD[c(1,.N)], by=V2] Selects the first and last row grouped by V2 V1 V3 adabilitySD is only accessible in j. DT[, lapply(.SD, sum), by=V2] Calculates the sum of all columns in .SD V2 V1 V3 DT[, lapply(.SD, sum), by=V2] Calculates the sum of all columns in .SD V2 V1 V3 grouped by V2. 1: A 6 -1.9505 2: B 6 -1.9505 2: B 6 -1.9505 DCols is used together with .SD, to DT[, lapply(.SD, sum), by=V2, .Same as above, but only for columns V3 V2 V3 used in j. .SDcols = c("V3", "V4")] and V4 of .SD. V2 V3 DCls can be the result of a netion call. DT[, lapply(.SD, sum), by=V2, .Same result as the line above. 2: B -1.9505 .SDcols = paste0("V", 3:4)] .Stocls = paste0("V", 3:4)] .CHAINING HELPS TACK EXPRESSIONS TOGETHER AND AVODD (UNNECESSARY) INTERMEDIATE ASSIGNMENTS V1 v4 Scols of statements DT<-DT[, .(V4.Sum = sum(V4)), by=V1]	.SD is a data.table and holds all the values of all columns, except the one specified in by . It reduces programming time but keeps readabilitySD is only accessible in j .		<pre>DT[, print(.SD), by=V2</pre>				ll of .SD (output
4: B 1 -1.0604 5: C 1 -1.0604 6: C 2 0.6651DT[, lapply(.SD, sum), by=V2] grouped by V2.Calculates the sum of all columns in .SD grouped by V2.V2 V1 V3 1: A 6 -1.9505 2: B 6 -1.9505 2: C 6 -1.9505 2: C 6 -1.9505 2: C 6 -1.9505 2: C 6 -1.9505 2: Dcols is used together with .SD, to ecify a subset of the columns of .SD to used in j.DT[, lapply(.SD, sum), by=V2, .SDcols = c("V3", "V4")] and V4 of .SD.Same as above, but only for columns V3 and V4 of .SD.V2 V3 1: A -1.9505 2: E -1.9505 2: E -1.9505 3: C 1 -1.9505Dcols can be the result of a netion call.DT[, lapply(.SD, sum), by=V2, .SDcols = paste0("V", 3:4)]Same result as the line above. 3: C -1.9505CHAINING HELPS TACK EXPRESSIONS TOGETHER AND AVOID (UNNECESSARY) INTERMEDIATE ASSIGNMENTSV2 V1 V2What?ExampleNotesVhat?ExampleNotesOutput p2 (or more) sets of statements once by chaining them in oneDT<[, .(V4.Sum = sum(V4)), by=V1] Ho chainingFirst calculates sum of V4, grouped by V1. Then selects that group of which the sum is > 40			DT[,.SD[c(1,.N)], by=V	2] Se	lects the first and last row gro	he	re) V2 V1 V3 1: A 1 -1.1727 2: A 2 -0.3825
grouped by V2. 1: A 6 -1.9505 grouped by V2. 1: A 6 -1.9505 grouped by V2. 1: A 6 -1.9505 2: B 6 -1.9505 3: C 6 -1.9505 3: C 6 -1.9505 3: C 6 -1.9505 DECols is used together with .SD, to DT[, lapply(.SD, sum), by=V2, Same as above, but only for columns V3 and V4 of .SD. used in j. .SDcols = c("V3", "V4")] and V4 of .SD. DECols can be the result of a net of a motion and the line above. DT[, lapply(.SD, sum), by=V2, Same result as the line above. Scools = paste0("V", 3:4)] 3: C -1.9505 CHAINING HELPS TACK EXPRESSIONS TOGETHER AND AVOID (UNNECESSARY) INTERMEDIATE ASSIGNMENTS 0.1.9505 What? Example Notes 0.2 (or more) sets of statements DT .(V4.Sum = sum(V4)), by=V1] 0.2 (or more) sets of statements DT .(V4.Sum = sum(V4)), by=V1] 0.2 (or more) sets of statements DT .(V1.V4.Sum > 40] #no chaining selects that group of which the sum is > 40 1: 1			DT lapple of	h17-1701	laulates the states of the	a in the	4: B 1 -1.0604 5: C 1 -1.0604 6: C 2 0.6651
ecify a subset of the columns of .SD to .SDcols = c("V3", "V4")] and V4 of .SD. V2 V3 used in j.			שדן, tapply(.SD, sum),			sın .SD	1: A 6 -1.9505 2: B 6 -1.9505
used in j. 2: B -1.9505 Dcols can be the result of a notion call. DT[, lapply(.SD, sum), by=V2, Same result as the line above. SDcols = paste0("V", 3:4)] 3: C -1.9505 CHAINING HELPS TACK EXPRESSIONS TOGETHER AND AVOID (UNNECESSARY) INTERMEDIATE ASSIGNMENTS 3: C -1.9505 What? Example Notes Of (units) once by chaining them in one DT (V4. Sum = sum (V4)), by=V1] First calculates sum of V4, grouped by V1. Then once by chaining them in one DT V1 V4.	ecify a subset of the columns of					mns V3	
CHAINING HELPS TACK EXPRESSIONS TOGETHER AND AVOID (UNNECESSARY) INTERMEDIATE ASSIGNMENTS What? Example Notes Output 0.2 (or more) sets of statements once by chaining them in one DT<-DT[, .(V4.Sum = sum(V4)), by=V1]	Dcols can be the result of a			-	me result as the line above.		2: B -1.9505
What?ExampleNotesOutput0 2 (or more) sets of statements once by chaining them in oneDT<-DT[, . (V4.Sum = sum(V4)), by=V1]							
once by chaining them in one DT[V4.Sum > 40] #no chaining selects that group of which the sum is > 40 1:1			Example		Notes		
rresponds to <i>having</i> in SQL.				ng s	elects that group of which the		1: 1

	DT[, .(V4.Sum = sum(V4)), by=V1][V4.Sum > 40]	Same as above, but with chaining.	V1 V4.Sum 1: 2 42
Order the results by chaining.	<pre>DT[, .(V4.Sum = sum(V4)),</pre>	Calculates sum of $V4$, grouped by $V1$, and then orders the result on $V1$.	V1 V4.Sum 1: 2 42 2: 1 36

	USING TI	HE set()-FAMILY			
What?	Example	Notes	Output		
<pre>set() is used to repeatedly update rows and columns by reference. Set() is a loopable low overhead version of :=. Watch out: It can not handle grouping operations.</pre>	<pre>Syntax of set(): for (i in from: rows = list(3:4,5:6) cols = 1:2 for (i in seq_along(rows)) { set(DT, i=rows[[i]], j = cols[i], value = NA) }</pre>	to) set(DT, row, column, new val Sequence along the values of rows, and for the values of cols, set the values of those elements equal to NA.	Lue). Returns the result invisibly. > DT V1 V2 V3 V4 1: 1 A -1.1727 1 2: 2 B -0.3825 2 3: NA C -1.0604 3 4: NA A 0.6651 4 5: 1 NA -1.1727 5 6: 2 NA -0.3825 6 7: 1 A -1.0604 7 8: 2 B 0.6651 8		
setnames () is used to create or update column names by	<pre>Syntax of setnames(): setnames(DT,"old","new")[]</pre>	Changes (set) the name of column old end of any set () function the result is			
reference.	<pre>setnames(DT,"V2","Rating") setnames(DT,c("V2","V3"), c("V2.rating","V3.DataCamp"))</pre>	Sets the name of column V2 to Rating . Changes two column names.	Returns the result invisibly. Returns the result invisibly.		
setcolorder () is used to reorder columns by reference.	<pre>setcolorder(DT, "neworder")</pre>	neworder is a character vector of the new column name ordering.			
	<pre>setcolorder(DT,</pre>	Changes the column ordering to the contents of the vector.	Returns the result invisibly. The new column order is now [1] "V2" "V1" "V4" "V3"		