[before] d_tip.top(I):

(index)	id	date	q…	total	tip	type
0	"03"	"2011-11-14T16:28:54Z"	1	300	200	"visa"
1	"12"	"2011-11-14T17:29:52Z"	1	200	100	"visa"
2	"02"	"2011-11-14T16:20:19Z"	2	190	100	"tab"
3	"01"	"2011-11-14T16:17:54Z"	2	190	100	"tab"
4	"11"	"2011-11-14T17:25:45Z"	2	200	0	"cash"
5	"10"	"2011-11-14T17:22:59Z"	2	90	0	"tab"
6	"09"	"2011-11-14T17:07:21Z"	2	90	0	"tab"
7	"08"	"2011-11-14T16:58:03Z"	2	90	0	"tab"
8	"07"	"2011-11-14T16:54:06Z"	1	100	0	"cash"
9	"06"	"2011-11-14T16:53:41Z"	2	90	0	"tab"
10	"05"	"2011-11-14T16:48:46Z"	2	90	0	"tab"
11	"04"	"2011-11-14T16:30:43Z"	2	90	0	"tab"

[before] d_type.top(I):

(index)	id	date	q…	total	tip	type
0	"12"	"2011-11-14T17:29:52Z"	1	200	100	"visa"
1	"03"	"2011-11-14T16:28:54Z"	1	300	200	"visa"
2	"10"	"2011-11-14T17:22:59Z"	2	90	0	"tab"
3	"09"	"2011-11-14T17:07:21Z"	2	90	0	"tab"
4	"08"	"2011-11-14T16:58:03Z"	2	90	0	"tab"
5	"06"	"2011-11-14T16:53:41Z"	2	90	0	"tab"
6	"05"	"2011-11-14T16:48:46Z"	2	90	0	"tab"
7	"04"	"2011-11-14T16:30:43Z"	2	90	0	"tab"
8	"02"	"2011-11-14T16:20:19Z"	2	190	100	"tab"
9	"01"	"2011-11-14T16:17:54Z"	2	190	100	"tab"
10	"11"	"2011-11-14T17:25:45Z"	2	200	0	"cash"
11	"07"	"2011-11-14T16:54:06Z"	1	100	0	"cash"

[before] g_tip.top(I):

(index)	key	value
0	0	8
1	100	3
2	200	1

[before] g_tip.all():

(index)	key	value
0	0	8
1	100	3
2	200	1

[before] g_type.top(I):

(index)	key	value
0	"tab"	8
1	"visa"	2
2	"cash"	2

[before] g_type.all():

(index)	key	value
0	"cash"	2
1	"tab"	8
2	"visa"	2

[before] facts.groupAll().value(): 12
[before] d_tip.groupAll().value(): 12
[before] d_type.groupAll().value(): 12

--- the next two statements are equivalent ---
[before] facts.groupAll().value(): 12
[before] facts.groupAll().reduceCount().value(): 12

--- the next two statements are equivalent ---
[before] d_tip.groupAll().value(): 12
[before] d_tip.groupAll().reduceCount().value(): 12

--- the next two statements are equivalent ---
[before] d_type.groupAll().value(): 12
[before] d_type.groupAll().reduceCount().value(): 12

--- applying reduceSum ---
[before] facts.groupAll().reduceSum(d=>.d.total).value(): 1720
[before] d_tip.groupAll().reduceSum(d=>.d.total).value(): 1720
[before] d_type.groupAll().reduceSum(d=>.d.total).value(): 1720

d_type.filter("tab") <<<<<<<

d_tip.top(I):

(ind…	id	date	qu…	total	tip	type
0	"02"	"2011-11-14T16:20:19Z"	2	190	100	"tab"
1	"01"	"2011-11-14T16:17:54Z"	2	190	100	"tab"
2	"10"	"2011-11-14T17:22:59Z"	2	90	0	"tab"
3	"09"	"2011-11-14T17:07:21Z"	2	90	0	"tab"
4	"08"	"2011-11-14T16:58:03Z"	2	90	0	"tab"
5	"06"	"2011-11-14T16:53:41Z"	2	90	0	"tab"
6	"05"	"2011-11-14T16:48:46Z"	2	90	0	"tab"
7	"04"	"2011-11-14T16:30:43Z"	2	90	0	"tab"

d_type.top(I):

(inde…	id	date	q…	total	tip	type
0	"10"	"2011-11-14T17:22:59Z"	2	90	0	"tab"
1	"09"	"2011-11-14T17:07:21Z"	2	90	0	"tab"
2	"08"	"2011-11-14T16:58:03Z"	2	90	0	"tab"
3	"06"	"2011-11-14T16:53:41Z"	2	90	0	"tab"
4	"05"	"2011-11-14T16:48:46Z"	2	90	0	"tab"
5	"04"	"2011-11-14T16:30:43Z"	2	90	0	"tab"
6	"02"	"2011-11-14T16:20:19Z"	2	190	100	"tab"
7	"01"	"2011-11-14T16:17:54Z"	2	190	100	"tab"

g_tip.top(I):

(index)	key	value
0	0	6
1	100	2
2	200	0

8 to 6 b/c id=[11,07] is excluded
3 to 2 b/c id=12 is excluded
1 to 0 b/c id=03 is excluded

g_tip.all():

(index)	key	value
0	0	6
1	100	2
2	200	0

8 to 6 b/c id=[11,07] is excluded
3 to 2 b/c id=12 is excluded
1 to 0 b/c id=03 is excluded

g_type.top(I):

(index)	key	value
0	"tab"	8
1	"visa"	2
2	"cash"	2

g_type.all():

(index)	key	value
0	"cash"	2
1	"tab"	8
2	"visa"	2

facts.groupAll().value(): 8
d_tip.groupAll().value(): 8
d_type.groupAll().value(): 12

Note: a grouping intersects the crossfilter's current filters, **except for the associated dimension's filter**. Thus, group methods consider only records that satisfy every filter except this dimension's filter. So, if the crossfilter of payments is filtered by type and total, then group by total only observes the filter by type.

--- the next two statements are equivalent ---
facts.groupAll().value(): 8
facts.groupAll().reduceCount().value(): 8

--- the next two statements are equivalent ---
d_tip.groupAll().value(): 8
d_tip.groupAll().reduceCount().value(): 8

--- the next two statements are equivalent ---
d_type.groupAll().value(): 12
d_type.groupAll().reduceCount().value(): 12

--- applying reduceSum ---
facts.groupAll().reduceSum(d=>.d.total).value(): 920
d_tip.groupAll().reduceSum(d=>.d.total).value(): 920
d_type.groupAll().reduceSum(d=>.d.total).value(): 1720