# Order related

# CONTENTS

Ordered set 2 Ordered loop Iteration function Positioning calculation 5

Sorting and alignment



Sequence	Requirement	Expression	Result
	Extract the third member	A1(3)	5
	Extract the first and fourth members	A1([1,4])	[2,10]
	Extract the first three members	A1(to(3))	[2,3,5]
[2,3,5,10,15,22,4,56]	Extract the second member	A1.m(2)	3
	Extract the second to fourth members	A1.m(2:4)	[3,5,10]
	Extract the last member	A1.m(-1)	56
	Extract the first two members, the fifth member, and the 7 <sup>th</sup> to last members	A1.m(:2,5,7:)	[2,3,15,4,56]

# Sequence tables are structured sequences

#### Ordered set

10000

quence table data       FEB EID NAME GENDER SALARY         1       1       1       Rebecca       F       7000         2       2       Ashley       F       1000       9000         3       3       Rachel       F       9000         4       4       Emily       F       7000         5       5       Ashley       F       10000         6       Mathew       M       11000         7       7       Alexis       F       9000         8       8       Megan       F       11000         9       Victoria       F       3000       13000	3     Ra       dex     EID       1     -       2     -       dex     EID       1     -       2     -       3     -       EID     -       4     -       2     -       3     -       EID     -       2     -       3     -       EID     -       2     -       3     -       EID     -       2     -       As     -	1 Re 4 En 2 As 3 Ra NAME	NAME ebecca mily NAME ebecca shley achel E	GENDER <u>E</u> <u>E</u> <u>E</u> <u>E</u> <u>E</u> <u>E</u> <u>E</u> <u>E</u>	SALAF SALAF SALAF 7 SALAF 11 9 SALARY
quence table data $1$ 11Rebeca $E$ 7000 $1$ 22Ashley $E$ 11000 $3$ 3Rachel $E$ 9000 $4$ 4Emily $E$ 7000 $6$ 6MatthewM11000 $6$ 6MatthewM11000 $7$ 7Alexis $E$ 9000 $9$ 9Victoria $E$ 3000 $10$ 10RyanM13000Astract the third memberA1(3)Extract the first and fourth membersA1([1,4])Extract the first three membersA1(to(3))Extract the second memberA1.m(2)	dex EID 1 2 dex EID 1 2 3 EID 2 As dex EID	1 Re 4 En 1 Re 2 As 3 Ra NAME	NAME ebecca mily NAME ebecca shley achel E	GENDER <u>F</u> <u>F</u> <u>F</u> <u>F</u> <u>F</u> <u>F</u> <u>F</u> <u>F</u>	SALAR 7( 7( 3ALAR 7( 110 90 SALARY
Image: Non-open constraints       Image: Non-open constraints       Image: Non-open constraints         Image: Non-open constraints       Image: Non-open constraints       Image: Non-open constraints         Image: Non-open constraints       Image: Non-open constraints       Image: Non-open constraints         Image: Non-open constraints       Image: Non-open constraints       Image: Non-open constraints         Image: Non-open constraints       Image: Non-open constraints       Image: Non-open constraints         Image: Non-open constraints       Image: Non-open constraints       Image: Non-open constraints         Image: Non-open constraints       Image: Non-open constraints       Image: Non-open constraints         Image: Non-open constraints       Image: Non-open constraints       Image: Non-open constraints         Image: Non-open constraints       Image: Non-open constraints       Image: Non-open constraints         Image: Non-open constraints       Image: Non-open constraints       Image: Non-open constraints         Image: Non-open constraints       Image: Non-open constraints       Image: Non-open constraints         Image: Non-open constraints       Image: Non-open constraints       Image: Non-open constraints         Image: Non-open constraints       Image: Non-open constraints       Image: Non-open constraints         Image: Non-open constraint       Image: Non-open constraints       Image:	dex EID 1 2 dex EID 1 2 3 EID 2 As dex EID	1 Re 4 En 2 As 3 Ra NAME hley	NAME ebecca mily NAME ebecca shley achel E E	GENDER <u>F</u> <u>F</u> GENDER <u>F</u> <u>F</u> <u>F</u> GENDER	SALAR' 70 SALAR' 70 110 90 SALARY
1       1	1 2 dex EID 1 2 3 EID 2 As exercises 2 As exercises EID EID EID EID EID EID EID EID EID	1 Re 4 En 1 Re 2 As 3 Ra NAME	ebecca mily NAME ebecca shley achel E E <u>F</u>	E           E           GENDER           E           E           E           GENDER	70 70 SALAR) 70 110 90 SALARY
0       0	2 dex EID 1 2 3 EID 2 As dex EID	4 En 1 Re 2 As 3 Ra NAME hley	NAME ebecca shley achel E E E	<u>F</u> <u>GENDER</u> <u>F</u> <u>F</u> <u>GENDER</u>	70 SALARY 70 110 90 SALARY
S       Ashley       F       16000         6       6       6       6       1000         7       7       7       7       1000         8       8       Megan       F       11000         9       9       9       9       10000         10       10       Ryan       13000         Requirement       Expression         Extract the third member       A1(3)         Extract the first and fourth members       A1([1,4])         Extract the first three members       A1(to(3))         Extract the second member       A1.m(2)	dex EID 1 2 3 EID 2 As block A	1 Re 2 As 3 Ra NAME	NAME ebecca shley achel E E <u>F</u>	GENDER <u>F</u> <u>F</u> <u>F</u> GENDER	SALARY
6       6       Matthew       M       11000         7       7       Alexis       F       9000         8       8       Megan       F       11000         9       9       Victoria       F       3000         10       10       Ryan       M       13000         Requirement       Expression         Extract the third member       A1(3)         Extract the first and fourth members       A1([1,4])         Extract the first three members       A1(to(3))         Extract the second member       A1.m(2)	1 2 3 EID 2 As dex EID	1 Re 2 As 3 Ra NAME	ebecca shley achel E <u>E</u>	E       E       E       GENDER	70 110 90 SALARY
7       7       A lexis       F       9000         8       8       Megan       F       11000         9       9       Victoria       F       3000         10       10       Ryan       M       13000	2 3 EID 2 As dex EID	2 As 3 Ra NAME	achel E E	GENDER	110 90 SALARY
8       8       Megan       F       11000         9       9       9       Victoria       F       3000         10       10       Ryan       M       13000         Requirement       Expression         Extract the third member       A1(3)         Extract the first and fourth members       A1([1,4])         Extract the first three members       A1(to(3))         Extract the second member       A1.m(2)	BID 2 As dex EID	3 Ra NAME	achel E <u>E</u>	GENDER	90 SALARY
9       9       Victoria       E       3000         10       10       Ryan       13000       13000         Requirement       Expression         Extract the third member       A1(3)         Extract the first and fourth members       A1([1,4])         Extract the first three members       A1(to(3))         Extract the second member       A1.m(2)	EID 2 As dex EID	NAME hley	E <u>F</u>	GENDER	SALARY
10       10       Requirement       Expression         Extract the third member       A1(3)         Extract the first and fourth members       A1([1,4])         Extract the first three members       A1(to(3))         Extract the second member       A1.m(2)	EID 2 As dex EID	NAME hley	E <u>F</u>	GENDER	SALARY
RequirementExpressionExtract the third memberA1(3)Extract the first and fourth membersA1([1,4])Extract the first three membersA1(to(3))Extract the second memberA1.m(2)	2 <u>As</u> dex EID	hley	<u>F</u>		44004
RequirementExpressionExtract the third memberA1(3)Extract the first and fourth membersA1([1,4])Extract the first three membersA1(to(3))Extract the second memberA1.m(2)	dex EID				11000
Requirement       Expression         Extract the third member       A1(3)         Extract the first and fourth members       A1([1,4])         Extract the first three members       A1(to(3))         Extract the second member       A1.m(2)			NAME	GENDER	SALARY
Extract the third memberA1(3)Extract the first and fourth membersA1([1,4])Extract the first three membersA1(to(3))Extract the second memberA1.m(2)	1	2 <u>As</u>	shley	<u>F</u>	110
Extract the first and fourth members       A1([1,4])         Extract the first three members       A1(to(3))         Extract the second member       A1.m(2)	2	3 <u>R</u> a	achel	<u>F</u>	90
Extract the first and fourth members       A1([1,4])         Extract the first three members       A1(to(3))         Extract the second member       A1.m(2)	3	4 <u>Er</u>	mily	<u>F</u>	70
Extract the first three members A1(to(3)) Extract the second member A1.m(2)	EID	NAME	E	GENDER	SALARY
Extract the first three members A1(to(3)) Extract the second member A1.m(2)	500 Jo	seph	М		12000
Extract the second member A1.m(2)				051050	
Extract the second member A1.m(2)	dex EID	4	NAME	GENDER	SALARY
	1	1 1 1 1 1 1 1	ebecca	<u>F</u>	/0
	2	2 <u>As</u>	sniey	<u>F</u>	110
Extract the second to fourth members A1.m(2:4)	3	5 AS	shley	<u>F</u>	160
Extract the last member $A1 m(-1)$	5	7 <u>Ale</u>	exis	<u> </u>	90
	0	0 1/6	ictoria	<u>_</u>	20
Extract the first two members, the fifth member, and	and it	10 P	van	<u>_</u>	120
the 7 <sup>th</sup> to last members	7		yall	M	130
	7	11 10	acab		120

13 Daniel

М

10

	Α	В
1	=file("E:/txt/students_score.txt").import@t()	
2	=A1.(English).sort()	/Sort by English score
3	=A2.len()+1	/Number of set members
4	=A2([A3\2,(A3+1)\2]).avg()	/If even number, take the mean value of the middle two; If odd number, take the middle number.

$\Delta 1 \sim \Delta / \Gamma \Delta C \Pi f C'$			14
	A1	$\sim A4$	results:

Index	Name	Math	Chinese	English
1	Natalie	84	90	84
2	Jessica	87	88	78
3	Brianna	89	90	75
4	Emma	88	84	94
5	Zachary	75	81	85
6	Sophia	74	86	93
7	Hannah	90	76	95
8	Christopher	71	81	86
9	Sean	98	86	81
10	Tyler	87	78	93

Index	Member	
1	75	
2	78	
3	81	
4	84	
5	85	
6	86	
7	93	
8	93	
9	94	
10	95	

	Value	
11		85.5

Value

A.pos(x), where x is a sequence, get the position of x members in A



A.pos(x), equivalent to the inverse operation of A(p)=x.

	Α	B
1	=file("E:/txt/students_score.txt").import@t()	
2	=A1.sort(-(Chinese+Math+English))	/Sort according to the total score from high to low
3	=A2.(Name).pos(["Tyler","Sean"])	/Query the rankings of "Tyler" and "Sean"

### A1~A3 results:

Index	Name	Math	Chinese	English
1	Natalie	84	90	84
2	Jessica	87	88	78
3	Brianna	89	90	75
4	Emma	88	84	94
5	Zachary	75	81	85
6	Sophia	74	86	93
7	Hannah	90	76	95
8	Christopher	71	81	86
9	Sean	98	86	81
10	Tyler	87	78	93

Index	Name	Math	Chinese	English
1	Emma	88	84	94
2	Sean	98	86	81
3	Hannah	90	76	95
4	Natalie	84	90	84
5	Tyler	87	78	93
6	Brianna	89	90	75
7	Jessica	87	88	78
8	Sophia	74	86	93
9	Zachary	75	81	85
10	Christopher	71	81	86

nber	Index
5	1
2	2

# Filter for records containing null values

Result

Index	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
1	1.0	Rebecca	Moore	<u>F</u>	California	1974-11-20	2005-03-11	R&D	7000.0
2	2.0	Ashley	Wilson	<u>F</u>	New York	1980-07-19	2008-03-16	Finance	11000.0
3	3.0	Rachel	Johnson	<u>F</u>	New Mexico	1970-12-17	2010-12-01	Sales	9000.0
4	4.0	Emily	Smith	<u>F</u>	Texas	1985-03-07	2006-08-15	HR	7000.0
5	5.0	Ashley	Smith	<u>F</u>	Texas	1975-05-13	2004-07-30	R&D	16000.0
6	6.0	Matthew	Johnson	M	California	1984-07-07	2005-07-07	Sales	11000.0
7	7.0	Alexis	Smith	<u>F</u>	Illinois	1972-08-16	2002-08-16	Sales	9000.0
8	8.0	Megan	Wilson	<u>F</u>	California	1979-04-19	1984-04-19	Marketing	11000.0
9	9.0	Victoria	Davis	<u>F</u>	Texas	1983-12-07	2009-12-07	HR	3000.0
10	10.0	Ryan	Johnson	M	Pennsylva	1976-03-12	2006-03-12	R&D	13000.0

# A1=file("E:/txt/EMPLOYEE\_nan.txt").import@t()2=A1.select(~.array().pos(null)>0)

	Index	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
	1	16.0	Christopher	(null)	M	Florida	1979-06-27	2007-06-27	Production	9000.0
	2	17.0	Hannah	Johnson	F	Texas	(null)	2006-07-19	Marketing	4000.0
N	3	(null)	Joseph	(null)	M	California	1983-08-27	2003-08-27	Finance	6000.0
	4	27.0	Alexis	Jones	F	California	1983-12-27	(null)	Marketing	10000.0
	5	(null)	Olivia	Harris	F	California	1979-08-27	2009-08-27	Sales	8000.0
	6	35.0	Justin	Smith	M	Texas	1978-08-20	(null)	R&D	7000.0
	7	41.0	Emily	Davis	F	Illinois	1984-07-07	(null)	Production	9000.0
	8	43.0	Joshua	Williams	M	California	1980-11-25	2000-11-25	(null)	12000.0
	9	55.0	Olivia	Anderson	F	(null)	1971-08-27	2001-08-27	Technology	13000.0
	10	65.0	Michael	Smith	(null)	Connecticut	1971-03-03	2004-03-01	Sales	8000.0

# Set union

## **Part of Employee Data Table**

Index	EID	NAME	GENDER	DEPT	SALARY
1	1	Rebecca	<u>F</u>	R&D	7000
2	2	Ashley	<u>F</u>	Finance	11000
3	3	Rachel	<u>F</u>	Sales	9000
4	4	Emily	<u>F</u>	HR	7000
5	5	Ashley	<u>F</u>	R&D	16000
6	6	Matthew	M	Sales	11000
7	7	Alexis	<u>F</u>	Sales	9000
8	8	Megan	<u>F</u>	Marketing	11000
9	9	Victoria	<u>F</u>	HR	3000
10	10	Ryan	M	R&D	13000

#### A.select(GENDER=="M")

Index	EID	NAME	GENDER	DEPT	SALARY
1	6	Matthew	M	Sales	11000
2	10	Ryan	M	R&D	13000
3	11	Jacob	M	Sales	12000
4	13	Daniel	M	Finance	10000
5	16	Christopher	M	Production	9000

#### A2&A3

Index	EID	NAME	GENDER	DEPT	SALARY
1	6	Matthew	M	Sales	11000
2	10	Ryan	M	R&D	13000
3	11	Jacob	M	Sales	12000
4	13	Daniel	M	Finance	10000
5	16	Christopher	M	Production	9000

#### A.select(DEPT=="Sales")

Index	EID	NAME	GENDER	DEPT	SALARY
1	3	Rachel	<u>F</u>	Sales	9000
2	6	Matthew	M	Sales	11000
3	7	Alexis	<u>F</u>	Sales	9000
4	11	Jacob	M	Sales	12000
5	12	Jessica	<u>F</u>	Sales	7000

#### A.select(GENDER==M||DEPT=="Sales")

Index	EID	NAME	GENDER	DEPT	SALARY
1	3	Rachel	<u>F</u>	Sales	9000
2	6	Matthew	M	Sales	11000
3	7	Alexis	<u>F</u>	Sales	9000
4	10	Ryan	M	R&D	13000
5	11	Jacob	M	Sales	12000
4	10 11	Ryan Jacob	M	R&D Sales	1

**(B)** 

A.eq

true

Sequence	Calculation	Description	Result
	A1==A2	Determine whether the sequence is equal	false
A1=[1,2,3]	A1^A3	Intersection of A1 and A3	[1,3]
A2=[1,3,2]	A3^A1	Intersection of A3 and A1	[3,1]
A3=[3,1,4,5]	A1&A3	Union of A1 and A3	[1,2,3,4,5]
	A3&A1	Union of A3 and A1	[3,1,4,5,2]

#### Determine whether members are the same

Sequence	Calculation	Description	Result
A1=[1,2,3]	A1.eq(A2)	Dotormino whothor mombors	true
A2=[1,3,2] A3=[3,1,4,5]	A1.eq(A3)	are the same	false

Table

A2&A3

EID	NAME	GENDER	DEPT	SALARY
2	Matthew	M	Sales	11000
6	Ryan	M	R&D	13000
7	Jacob	M	Sales	12000
3	Alexis	<u>F</u>	Sales	9000

A select(	GENDER==M	/IIDEPT=="Sal	, '
7.301001			3

EID	NAME	GENDER	DEPT	SALARY
2	Matthew	M	Sales	11000
3	Alexis	<u>F</u>	Sales	9000
6	Ryan	M	R&D	13000
7	Jacob	M	Sales	12000



# Employee data

Index	EID	NAME	GENDER	SALARY
1	1	Rebecca	<u>F</u>	7000
2	2	Ashley	<u>F</u>	11000
3	3	Rachel	<u>F</u>	9000
4	4	Emily	<u>F</u>	7000
5	5	Ashley	<u>F</u>	16000
6	6	Matthew	M	11000
7	7	Alexis	<u>F</u>	9000
8	8	Megan	<u>F</u>	11000
9	9	Victoria	<u>F</u>	3000
10	10	Ryan	M	13000
11	11	Jacob	M	12000
12	12	Jessica	F	7000

#### Records of even-numbered positions (A.select(#%2==0))

Index	EID	NAME	GENDER	SALARY
1	2	Ashley	<u>F</u>	11000
2	4	Emily	<u>F</u>	7000
3	6	Matthew	M	11000
4	8	Megan	<u>F</u>	11000
5	10	Ryan	M	13000
6	12	Jessica	<u>F</u>	7000
7	14	Alyssa	<u>F</u>	4000
8	16	Christopher	M	9000
9	18	Jonathan	M	7000
10	20	Alexis	<u>F</u>	16000
11	22	Jacob	M	16000
12	24	Chloe	<u>F</u>	10000

### Ordered loop - Cross-row reference

# A.(x),A.(x[])

Sequence	Function	Expression	Result	Description
	A.(~)	~	[2,3,5,6]	Return the original sequence
[2,3,5,6]	A.(~[-1])	~[-1]	[null,2,3,5]	Return a member whose position from the current member is - 1
	A.(~[2])	~[2]	[5,6,null,null]	Return a member whose position from the current member is 2

			Fund	ction			Exp	oress	ion				De	scrip	tion				
			A.(p	rod)				prod					F	Prod fie	eld				
			A.(pro	od[-1])		-	р	rod[-1	]	Me	embers	with a posi	ition of -	1 from	the memb	oers o	of the pr	od field	
			A.(pro	od[:])			F	orod[:	]		A seq	uence cons	sisting of	a sequ	uence of a	ll pro	d memb	ers	
			A.(pro	od[:0])			р	rod[:(	)]	A se	quence	e of membe	rs conta	ining p	rod from s	start to	o curren	t positior	า
						7									+				
S	· · · · ·	Index	Member		Index	Member		Index	Member		Index	Member		Index	Member				
018.2		1	1		1	(null)		1	[1,2,3,]		1	1		1	[1]	1			
814.8		2	2		2	1		2	[1,2,3,]		2	2		2	[1,2]	1			
400.0		3	3		3	2		3	[1.2.3]		3	3		3	[1,2,3]				
0.0		4	4		4	3		4	[1.2.3]		4	4		4	[1,2,3,]				
0.0		5	5		5	4		5	[1.2.3]		5	5		5	[1,2,3,]			日告	
0.0		6	6		6	5		6	[1,2,3,]		6	6		6	[1,2,3,]		1	JAN MA	1
0.0		7	7		7	6		7	[1,2,3,]		7	7		7	[1,2,3,]		2		2
0.0		8	8		8	7		8	[1,2,3,]		8	8		8	[1,2,3,]		3		- 3
897.6		9	9		9	8		9	[1,2,3,]		9	9		9	[1,2,3,]	1			
0.0		10	10		10	9		10	[1,2,3,]		10	10		10	[1,2,3,]	1			

# Data

Index	prod	month	sales
1	1	1	2018.2
2	2	1	1814.8
3	3	1	400.0
4	4	1	0.0
5	5	1	0.0
6	6	1	0.0
7	7	1	0.0
8	8	1	0.0
9	9	1	1897.6
10	10	1	0.0

	Α
1	=file("E:/txt/stock1001_price.txt").import@t()
2	=a=0,A1.max(a=if(CL>CL[-1],a+1,0))

A1, A2 results

Index	stockid	DT	CL
1	1001	2009-01-01	4.0
2	1001	2009-01-02	3.64
3	1001	2009-01-05	3.95
4	1001	2009-01-06	3.68
5	1001	2009-01-07	3.53
6	1001	2009-01-08	3.59
7	1001	2009-01-09	3.9
8	1001	2009-01-12	3.56
9	1001	2009-01-13	3.22
10	1001	2009-01-14	3.17



	Α
1	=file("E:/txt/stock1001_price.txt").import@t()
2	=A1.derive(CL[-5:0].avg():5_avg)

A1,A2 results
---------------

Index	stockid	DT	CL
1	1001	2009-01-01	4.0
2	1001	2009-01-02	3.64
3	1001	2009-01-05	3.95
4	1001	2009-01-06	3.68
5	1001	2009-01-07	3.53
6	1001	2009-01-08	3.59
7	1001	2009-01-09	3.9
8	1001	2009-01-12	3.56
9	1001	2009-01-13	3.22
10	1001	2009-01-14	3.17

Index	stockid	DT	CL	5_avg
1	1001	2009-01-01	4.0	4.0
2	1001	2009-01-02	3.64	3.8200000
3	1001	2009-01-05	3.95	3.8633333
4	1001	2009-01-06	3.68	3.8175
5	1001	2009-01-07	3.53	3.7600000
6	1001	2009-01-08	3.59	3.7316666
7	1001	2009-01-09	3.9	3.715
8	1001	2009-01-12	3.56	3.7016666
9	1001	2009-01-13	3.22	3.58
10	1001	2009-01-14	3.17	3.4949999

	Α
1	=file("E:/txt/prod1.txt").import@t()
2	=A1.derive(sum(sales[:0]):add_up)

A1,A2 results

sales	month	prod	Index
2018.2	1	1	1
0.0	2	1	2
672.0	3	1	3
2657.8	4	1	4
1344.0	5	1	5
838.45	6	1	6
1140.0	7	1	7
450.0	8	1	8
0.0	9	1	9
630.0	10	1	10
1850.0	11	1	11
0.0	12	1	12

Index	prod	month	sales	add_up
1	1	1	2018.2	2018.2
2	1	2	0.0	2018.2
3	1	3	672.0	2690.2
4	1	4	2657.8	5348.0
5	1	5	1344.0	6692.0
6	1	6	838.45	7530.45
7	1	7	1140.0	8670.45
8	1	8	450.0	9120.45
9	1	9	0.0	9120.45
10	1	10	630.0	9750.45
11	1	11	1850.0	11600.45
12	1	12	0.0	11600.45

	Α	В
1	=file("E:/txt/prod.txt").import@t()	
2	=A1.sort(prod,month)	/Sort by prod and month
3	=A2.select(if(prod==prod[-1],sales/sales[-1]>1.1))	/Find the result

A1~A3	
results	

Inde

¢	prod	month	sales	Index	prod	month	sales	Index	prod	month	sales
1	1	1	2018.2	1	1	1	2018.2	1	1	3	672.0
2	2	1	1814.8	2	1	2	0.0	2	2 1	4	2657.8
3	3	1	400.0	3	1	3	672.0	3	1	7	1140.0
1	4	1	0.0	4	1	4	2657.8	4	1	10	630.0
5	5	1	0.0	5	1	5	1344.0	Ę	i 1	11	1850.0
5	6	1	0.0	6	1	6	838.45	6	2	3	1125.5
7	7	1	0.0	7	1	7	1140.0	7	2	4	1388.5
3	8	1	0.0	8	1	8	450.0	8	2	6	0.0
9	9	1	1897.6	9	1	9	0.0	9	2	7	0.0
	10	1	0.0	10	1	10	630.0	10	2	8	605.0

()rdarad la	n

	Α
1	=file("E:/txt/prod.txt").import@t()
2	=A1.sort(prod,month)
3	=A2.derive(if(prod==prod[-1]&∏==prod[1],sales[-1:1].avg()):moving_avg)

	Index	prod	month	sales	Index	prod	month	sales	I	ndex	prod	month	sales	moving_avg
AI~A3	1	1	1	2018.2	1	1	1	2018.2		1	1	1	2018.2	(null)
results	2	2	1	1814.8	2	1	2	0.0		2	1	2	0.0	896.733333
roomo	3	3	1	400.0	3	1	3	672.0		3	1	3	672.0	1109.93333
	4	4	1	0.0	4	1	4	2657.8		4	1	4	2657.8	1557.93333
	5	5	1	0.0	5	1	5	1344.0		5	1	5	1344.0	1613.41666
	6	6	1	0.0	6	1	6	838.45		6	1	6	838.45	1107.48333
	7	7	1	0.0	7	1	7	1140.0		7	1	7	1140.0	809.483333
	8	8	1	0.0	8	1	8	450.0		8	1	8	450.0	530.0
	9	9	1	1897.6	9	1	9	0.0		9	1	9	0.0	360.0
	10	10	1	0.0	10	1	10	630.0		10	1	10	630.0	826.666666

	Α
1	=file("E:/txt/orders.txt").import@t()
2	=A1.sort(prod,month)
3	=A2.derive((x=if(prod!=prod[-1],#,x),sum(sales[x-#:0])):add_up)

# A1~A3 results

index	prod	monui	Sales
1	1	1	2018.2
2	2	1	1814.8
3	3	1	400.0
4	4	1	0.0
5	5	1	0.0
6	6	1	0.0
7	7	1	0.0
8	8	1	0.0
9	9	1	1897.6
10	10	1	0.0
11	11	1	1194.0
12	12	1	0.0
13	13	1	0.0
14	14	1	319.2
15	15	1	0.0
16	16	1	1005.9
17	17	1	102.4
18	18	1	0.0

Index pred menth color

Index	prou	monut	Sales
1	1	1	2018.2
2	1	2	0.0
3	1	3	672.0
4	1	4	2657.8
5	1	5	1344.0
6	1	6	838.45
7	1	7	1140.0
8	1	8	450.0
9	1	9	0.0
10	1	10	630.0
11	1	11	1850.0
12	1	12	0.0
13	2	1	1814.8
14	2	2	631.6
15	2	3	1125.5
16	2	4	1388.5
17	2	5	0.0
18	2	6	0.0

Index pred menth color

Index	prod	month	sales	add_up
1	1	1	2018.2	2018.2
2	1	2	0.0	2018.2
3	1	3	672.0	2690.2
4	1	4	2657.8	5348.0
5	1	5	1344.0	6692.0
6	1	6	838.45	7530.45
7	1	7	1140.0	8670.45
8	1	8	450.0	9120.45
9	1	9	0.0	9120.45
10	1	10	630.0	9750.45
11	1	11	1850.0	11600.45
12	1	12	0.0	11600.45
13	2	1	1814.8	1814.8
14	2	2	631.6	2446.4
15	2	3	1125.5	3571.9
16	2	4	1388.5	4960.4
17	2	5	0.0	4960.4
18	2	6	0.0	4960.4



# Iteration function

**SUM:** Set an initial value of 0, then traverse each member of the set, adding the member value to the initial value each time until the members are traversed.

MAX: Set the initial value to be infinitesimal, traverse the members of the set, and replace the initial value for each member larger than the initial value until the traversal is completed.

MIN: Like MAX, the initial value and the direction of comparison are reversed.

Iteration function

a=initial value,  $\sim$ = current member,  $\sim$ ~= current initial value In this way, these calculations can be completed.

# The iteration process of sum ():

3		~	~~		
5	Initialization		0		
6	Step 1	3	3		
2	Step 2	5	8		
1	Step 3	6	14		
9	Step i	A(I)	~~+A(i)		
8	Step 7	8	34	Result: 3	84

	Α	B
1	[2,4,6,11]	
2	=A1.sum()	=A1.iterate(~~=~~+~,0)
3	=A1.min()	=A1.iterate(if(~<~~,~,~~),inf())
4	=A1.max()	=A1.iterate(if(~>~~,~,~~),-inf())
5	=demo.query("select * from EMPLOYEE")	
6	=A5.maxp@a(EID)	=A5.iterate(if(!~  ~.EID>~~.EID,~,if(~.EID==~~.EID,~~ ~,~~)),null)
7	=A5.maxp@a(SALARY)	=A5.iterate(if(!~  ~.SALARY>~~.SALARY,~,if(~.SALARY==~~.SALARY,~~ ~,~~)),null)

### Same row results are the same, with 2,3,4,6,7 rows as follows

11

Value

23

Value

Value

In	dex	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
	1	500	Joseph	Smith	M	Pennsylva	1972-06-04	2003-12-01	Production	12000

Index	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
1	5	Ashley	Smith	<u>F</u>	Texas	1975-05-13	2004-07-30	R&D	16000
2	20	Alexis	Allen	<u>F</u>	Florida	1977-08-07	2007-08-07	Administrat	16000
3	22	Jacob	Davis	M	Texas	1985-05-07	2001-05-07	R&D	16000
4	47	Elizabeth	Brown	F	Pennsylva	1971-08-27	2001-08-27	Marketing	16000

	Α
1	=10.iterate([~~(2),(~~(1)+~~(2))],[1,1])



#### Approximate solution of Inx+2x-6=0 by method of bisection

	Α	В	
1	func	return In(A1)+2*A1-6	
	>y=1	1e-10	
2	=1000.iterate((x=(~~(1)+~~(2))/2,y=func(A1,x),if(y>0,[~~(1),x],[x,~~(2)])),[0,3],abs(y) <b2)< th=""></b2)<>		
3	=X		
4	=func(A1,x)		

The Iterative process of A2 :

	~~	x=(~~(1)+~~2)/2	y=ln(x)+2*x-6	abs(y)
Initialization	[0,3]			
Step 1	[0,3]	1.5	-2.5945	2.5945
Step 2	[1.5,3]	2.25	-0.6891	0.6891
Step 3	[2.25,3]	2.625	0.2151	0.2151
Step 4	[2.25,2.625]	2.4375	-0.234	0.234
Step i	~~	(~~(1)+~~(2))/2	ln(x)+2*x-6	abs(y)
Step n	~~	(~~(1)+~~(2))/2	ln(x)+2*x-6	if abs(y)<1e-10 break

Value 2.534919132012874

Value -2.6577851031106547E-11

	Α	В		
1	=demo.query("select ORDERID,SELLERID,AMOUNT,ORDERDATE from SALES ")			
2	=A1.select(SELLERID==10).sort(ORDERDATE)	/Select salesman No. 10 and sort by ORDERDATE		
3	=A2.derive(iterate(~~+AMOUNT,0):cum_sum)	/Using iterate function to calculate cumulative value		

# **A1**



	Index	ORDERID	SELLERID	AMOUNT	ORDERDATE
Results	1	1	17	392.0	2012-11-02
	2	2	6	4802.0	2012-11-09
	3	3	16	13500.0	2012-11-05
	4	4	9	26100.0	2012-11-08
	5	5	11	4410.0	2012-11-12
	6	6	18	6174.0	2012-11-07
	7	7	2	17800.0	2012-11-06
	8	8	7	2156.0	2012-11-09
	9	9	14	17400.0	2012-11-12
	10	10	10	19200.0	2012-11-12

Index	ORDERID	SELLERID	AMOUNT	ORDERDA
1	33	10	21800.0	2012-12-03
2	29	10	29500.0	2012-12-05
3	76	10	20700.0	2013-01-20
4	99	10	11800.0	2013-02-11
5	110	10	11000.0	2013-02-24
6	142	10	8232.0	2013-03-28
7	156	10	7448.0	2013-04-08
8	182	10	23700.0	2013-05-05
9	259	10	25600.0	2013-07-18
10	271	10	4900.0	2013-07-29



Index	ORDERID	SELLERID	AMOUNT	ORDERDA	cum_sum
1	33	10	21800.0	2012-12-0	21800.0
2	29	10	29500.0	2012-12-0	51300.0
3	76	10	20700.0	2013-01-2	72000.0
4	99	10	11800.0	2013-02-1	83800.0
5	110	10	11000.0	2013-02-2	94800.0
6	142	10	8232.0	2013-03-2	103032.0
7	156	10	7448.0	2013-04-0	110480.0
8	182	10	23700.0	2013-05-0	134180.0
9	259	10	25600.0	2013-07-1	159780.0
10	271	10	4900.0	2013-07-2	164680.0

114144.0

127744.0 153844.0

167844.0

	Α	В			
1	=demo.query("select CLIENT,SELLERID,AMOUNT,ORDERDATE from SALES ")				
2	2 =A1.sort(SELLERID,ORDERDATE) /Sort by SELLERID and ORDERDATE				
3	=A2.derive(iterate(~~=~~+AMOUNT,0;SELLERID):cum_sum)				
4	=A2.derive(cum(AMOUNT;SELLERID):cum_sum)				
	/A3 Using iterate function to calculate cumulative value; A4: cum(x;G) When G changes, x starts to calculate again.				

R	esults		<b>A1</b>				A2			ŀ	13,	<b>A4</b>	I		
Index	CLIENT	SELLERID	AMOUNT	ORDERDATE	Inde	X CLIENT	SELLERID	AMOUNT	ORDERDATE	Inde	CLIENT	SELLERID	AMOUNT	ORDERDA	cum_sum
1	UJRNP	17	392.0	2012-11-02 1		1 JAYB	1	7644.0	2012-11-16 1		1 JAYB	1	7644.0	2012-11-1	7644.
2	SJCH	6	4802.0	2012-11-09 1		2 HANAR	1	13200.0	2013-01-17 1		2 HANAR	1	13200.0	2013-01-1	20844.
3	UJRNP	16	13500.0	2012-11-05 1		3 <u>YZ</u>	1	11600.0	2013-01-20 1		3 <u>YZ</u>	1	11600.0	2013-01-2	32444.
4	PWQ	9	26100.0	2012-11-08 1		4 AVU	1	21800.0	2013-02-05 1		4 <u>AVU</u>	1	21800.0	2013-02-0	54244.
5	PWQ	11	4410.0	2012-11-12 1		5 <u>HL</u>	1	26400.0	2013-02-18 1		5 <u>HL</u>	1	26400.0	2013-02-1	80644.
6	HANAR	18	6174.0	2012-11-07 1		6 PWQ	1	17500.0	2013-02-21 1		6 PWQ	1	17500.0	2013-02-2	98144.
7	EGU	2	17800.0	2012-11-06 1		7 FHYBR	1	16000.0	2013-03-03 1		7 FHYBR	1	16000.0	2013-03-0	114144.
8	VILJX	7	2156.0	2012-11-09 1		8 <u>HP</u>	1	13600.0	2013-03-15 1		8 <u>HP</u>	1	13600.0	2013-03-1	127744.
9	JAYB	14	17400.0	2012-11-12 1		9 DNEDL	1	26100.0	2013-05-13 1		9 DNEDL	1	26100.0	2013-05-1	153844.
10	JAXE	19	19200.0	2012-11-12 1	1	0 EGU	1	14000.0	2013-05-20 1	1	0 EGU	1	14000.0	2013-05-2	167844.

	Α	В			
1	=file("E:/txt/students_c.txt").import@t()				
2	=A1.select(CLASS==1).sort(-SCORE)	/Sort in reverse order according to SCORE			
3	=A2.derive(iterate((x=x+1,if(SCORE==SCORE[-1],~~,x)),(x=0)):RANK)				
	/Initialize x=0, iterate x+1, when SCORE is the same, ranking is unchanged; If SCORE is different, ranking is x.				



A	1			
Index	CLASS		NAME	SCORE
1		1	Adams Bro	607
2		1	Adams Ha	620
3		1	Adams Jon	611
4		1	Allen Ashley	609
5		1	Allen Bran	612
6		1	Baker Dani	620
7		1	Brown Am	616
8		1	Brown Lau	618
9		1	Carter Mich	621
10		1	Clark Justin	615



Index	CLASS	NAME	SCORE
1	1	Smith Willi	629
2	1	Garcia Bryan	628
3	1	Jones Justin	628
4	1	Lee Rachel	628
5	1	Moore Mich	628
6	1	Smith Reb	628
7	1	Lewis Gabr	627
8	1	Martin Jose	627
9	1	Moore Jon	627
10	1	Williams N	627

# **A3**

Index	CLASS	NAME	SCORE	RANK
1	1	Smith Willi	629	1
2	1	Garcia Bryan	628	2
3	1	Jones Justin	628	2
4	1	Lee Rachel	628	2
5	1	Moore Mich	628	2
6	1	Smith Reb	628	2
7	1	Lewis Gab	627	7
8	1	Martin Jos	627	7
9	1	Moore Jon	627	7
10	1	Williams N	627	7

	Α	В			
1	=file("E:/txt/students_c.txt").import@t()				
2	2 =A1.sort(CLASS,-SCORE) /Sort in reverse order according to CLASS and SCOR				
3	=A2.derive(iterate((x=x+1,if(SCORE==SCORE[-1],~~,x)),(x=0);CLASS):RANK)				
4	=A2.derive(rank(SCORE;CLASS):RANK)				
	/The results for A3 and A4 are the same, A3:Initialize x=0, iterate x+1, when SCORE is the same, ranking is unchanged; If SCORE is different, ranking is x. When CLASS changes, x is reinitialized. A4:when SCORE is the same, ranking is unchanged; If SCORE is different, Ranking equals the current ranking plus the number of previous rankings. When				

CLASS changes, rank from 1.



Δ	1	
	- No.	

Index	CLASS	NAME	SCORE
1	1	Adams Bro	607
2	1	Adams Ha	620
3	1	Adams Jon	611
4	1	Allen Ashley	609
5	1	Allen Bran	612
6	1	Baker Dani	620
7	1	Brown Am	616
8	1	Brown Lau	618
9	1	Carter Mich	621
10	1	Clark Justin	615



Index	CLASS	NAME	SCORE
1	1	Smith Willi	629
2	1	Garcia Bryan	628
3	1	Jones Justin	628
4	1	Lee Rachel	628
5	1	Moore Mich	628
6	1	Smith Reb	628
7	1	Lewis Gab	627
8	1	Martin Jos	627
9	1	Moore Jon	627
10	1	Williams N	627

|--|

Index	CLASS	NAME	SCORE	RANK
1	1	Smith Willi	629	1
2	1	Garcia Bryan	628	2
3	1	Jones Justin	628	2
4	1	Lee Rachel	628	2
5	1	Moore Mich	628	2
6	1	Smith Reb	628	2
7	1	Lewis Gab	627	7
8	1	Martin Jos	627	7
9	1	Moore Jon	627	7
10	1	Williams N	627	7



# **Positioning Calculation**

No.	Sequence	Expression	Judgement	Option	Result
1	7		False		
2	15		True	Null	2
3	8		False		
4	9	~>10	False		
5	23		True	@a	[2 5 7]
6	5		Falese	eu	[2,0,7]
7	22		True		

A.pselec(GENDER:"M",DEPT:"Sales")

```
=A.pselect(GENDER=="M"&&DEPT=="Sales")
```

## Partial functions that are position related

Function

#### Positioning calculation



Index	EID	NAME	GENDER	SALARY
1	1	Rebecca	<u>F</u>	7000
2	2	Ashley	<u>F</u>	11000
3	3	Rachel	<u>F</u>	9000
4	4	Emily	<u>F</u>	7000
5	5	Ashley	<u>F</u>	16000
6	6	Matthew	M	11000
7	7	Alexis	<u>F</u>	9000
8	8	Megan	<u>F</u>	11000
9	9	Victoria	<u>F</u>	3000
10	10	Ryan	M	13000



			-		
Find out positions where EID is greater than 10 and less than or equal to 20			3	45	
Find out position of maximum EID					
Find out position of minimum salary	Index 1	EID 9	NAME Victoria	GENDER F	SALARY 3000
Get the records corresponding to the mittingen	2	39	Andrew	M	3000
value	3	45	Kayla	F	3000
	<ul> <li>Find out positions where EID is greater than 10 and less than or equal to 20</li> <li>Find out position of maximum EID</li> <li>Find out position of minimum salary</li> <li>Get the records corresponding to the minimum value</li> </ul>	Find out positions where EID is greater than 10 and less than or equal to 20         Find out position of maximum EID         Find out position of minimum salary         Get the records corresponding to the minimum value	Find out positions where EID is greater than 10 and less than or equal to 20         Find out position of maximum EID         Find out position of minimum salary         Get the records corresponding to the minimum value	Find out positions where EID is greater than 10 and less than or equal to 20       3         Find out position of maximum EID       Index EID NAME         Find out position of minimum salary       1       9       Victoria         Get the records corresponding to the minimum value       3       45       Kayla	Find out positions where EID is greater than 10 and less than or equal to 20       3       45         Find out position of maximum EID       Index EID NAME GENDER       1       9       Victoria E       1       9       Victoria E       1       9       Victoria E       2       39       Andrew M       3       45       45

	Α	B
1	=file("E:/txt/stock_price.txt").import@t()	
2	=A1.sort(stockid)	/Sort by stockid
3	=A2.pselect@a(CL>100)	/Find out record position where the stock price is above 100
4	=A2.calc(A3,if(day(DT)==1,~.CL,~.CL-~[-1].CL))	/Calculate increase rate using position

#### A1~A4 results:

Index	stockid	DT	CL
1	1001	2009-01-01	4.0
2	1026	2009-01-01	2.13
3	1028	2009-01-01	20.09
4	1070	2009-01-01	14.95
5	1107	2009-01-01	3.74
6	1134	2009-01-01	10.68
7	1137	2009-01-01	42.31
8	1147	2009-01-01	19.61
9	1206	2009-01-01	14.01
10	1213	2009-01-01	40.94

Index	stockid	DT	CL
1	1001	2009-01-01	4.0
2	1001	2009-01-02	3.64
3	1001	2009-01-05	3.95
4	1001	2009-01-06	3.68
5	1001	2009-01-07	3.53
6	1001	2009-01-08	3.59
7	1001	2009-01-09	3.9
8	1001	2009-01-12	3.56
9	1001	2009-01-13	3.22
10	1001	2009-01-14	3.17

Index	Member
1	243
2	244
3	245
4	246
5	247
6	248
7	257
8	260
9	353
10	354

Index	Member
1	134.61
2	-10.62000000000019
3	-3.890000000000000
4	-12.00999999999999
5	4.079999999999998
6	-9.60000000000000
7	9.409999999999999
8	4.67000000000002
9	231.91
10	-10.719999999999999

(Keep the stock price at 1<sup>st</sup> of the month)

	Α	В
1	=file("E:/txt/stock1001_price.txt").import@t()	
2	=A1.pmax(CL)	/Find the position of highest price
3	=A1.calc(A2,if(day(DT)==1,A1.CL,A1.CL/A1.CL[-1]-1))	/Calculate the growth rate of the day

A1~A3	results:
	results.

Index	stockid	DT	CL
1	1001	2009-01-01	4.0
2	1001	2009-01-02	3.64
3	1001	2009-01-05	3.95
4	1001	2009-01-06	3.68
5	1001	2009-01-07	3.53
6	1001	2009-01-08	3.59
7	1001	2009-01-09	3.9
8	1001	2009-01-12	3.56
9	1001	2009-01-13	3.22
10	1001	2009-01-14	3.17







# Sorting and alignment

BasePay

7500.0

6800.0

4300.0

8900.0

5700.0

11000.0

BasePay

5600.0

2500.0

10800.0

4700.0

9300.0

2900.0

3500.0

7200.0

7600.0

7500.0

		Α								В					
1	=file	=file("E:/txt/Employees.txt").import@t()													
2	=A1	l.sort(r	and())	(to(A1.I	en()*0.	3))			/Dist	urb the	order o	of data, t	ake 30%	)	
3	=A1	1\A2							/Difference set						
									Index	ID	Name	Gender	Post	Birthday	AccountN
									1	4	Andy	Male	Sales Man	1968-09-1	65002886
									2	23	Larry	Female	R&D Leader	1980-07-2	84193330
									3	10	Tiger	Male	General M	1970-10-1	25549086
dex	ID	Name	Gender	Post	Birthday	AccountNo	BasePay		4	17	Frank	Male	Engineer	1983-10-2	40000407
1	1	MIKe		Sale	1908-12-0	030930891	0.000		5	40	-				12828497
0		laka	Mala	Vice Dreed	1060.00.1	064407677	0500.0		5	10	Ed	Male	R&D Leader	1983-02-1	12828497 65254835
2	2	Jake	Male	Vice Presid	1962-02-1	964107677	2500.0		6	10	Charlie	Male Female	R&D Leader Engineer	1983-02-1 1979-01-0	12828497 65254835 39427476
2 3 4	2	Jake Lucy	Male Female	Vice Presid Sale	1962-02-1 1973-08-3	964107677 665248245	2500.0 10800.0 7500.0		6	10	Ed Charlie	Male Female	R&D Leader Engineer	1983-02-1 1979-01-0	12828497 65254835 39427476
2 3 4	2 3 4 5	Jake Lucy Andy	Male Female Male Male	Vice Presid Sale Sales Man	1962-02-1 1973-08-3 1968-09-1 1965-03-0	964107677 665248245 650028860 441380247	2500.0 10800.0 7500.0 4700.0		6	10	Ed Charlie	Male Female	R&D Leader Engineer	1983-02-1 1979-01-0	12828497 65254835 39427476
2 3 4 5 6	2 3 4 5 6	Jake Lucy Andy Jim David	Male Female Male Male Male	Vice Presid Sale Sales Man Sales Man Sale	1962-02-1 1973-08-3 1968-09-1 1965-03-0 1967-07-0	964107677 665248245 650028860 441380247 860916757	2500.0 10800.0 7500.0 4700.0 9300.0		6 Index	10 14	Ed Charlie Name	Male Female Gender	R&D Leader Engineer Post	1983-02-1 1979-01-0 Birthday	AccountN
2 3 4 5 6 7	2 3 4 5 6 7	Lucy Andy Jim David Jessica	Male Female Male Male Male Male	Vice Presid Sale Sales Man Sales Man Sale Sale	1962-02-1 1973-08-3 1968-09-1 1965-03-0 1967-07-0 1960-05-2	964107677 665248245 650028860 441380247 860916757 568899664	2500.0 10800.0 7500.0 4700.0 9300.0 2900.0		6 Index	ID 10 10	Ed Charlie Name Mike	Male Female Gender Female	R&D Leader Engineer Post Sale	1983-02-1 1979-01-0 Birthday 1968-12-0	Accountly 53693689
2 3 4 5 6 7 8	2 3 4 5 6 7 8	Andy Jim David Jessica	Male Female Male Male Male Male Female	Vice Presid Sale Sales Man Sales Man Sale Sale Inside Sale	1962-02-1 1973-08-3 1968-09-1 1965-03-0 1967-07-0 1960-05-2 1969-01-0	964107677 665248245 650028860 441380247 860916757 568899664 428844667	2500.0 10800.0 7500.0 4700.0 9300.0 2900.0 3500.0		6 Index 1 2	ID 12	Ed Charlie Name Mike Jake	Male Female Gender Female Male	R&D Leader Engineer Post Sale Vice Presid	1983-02-1 1979-01-0 Birthday 1968-12-0 1962-02-1	AccountN 53693689 96410767
2 3 4 5 6 7 8 9	2 3 4 5 6 7 8 9	Andy Andy David David Jessica Lily Mary	Male Female Male Male Male Male Female Female	Vice Presid Sale Sales Man Sales Man Sale Sale Inside Sale Sale	1962-02-1 1973-08-3 1968-09-1 1965-03-0 1967-07-0 1960-05-2 1969-01-0 1969-07-0	964107677 665248245 650028860 441380247 860916757 568899664 428844667 962447951	2500.0 10800.0 7500.0 4700.0 9300.0 2900.0 3500.0 7200.0		6 Index 1 2 3	ID 12 3	Ed Charlie Name Mike Jake Lucy	Male Female Gender Female Male Female	R&D Leader Engineer Post Sale Vice Presid Sale	1983-02-1 1979-01-0 Birthday 1968-12-0 1962-02-1 1973-08-3	AccountN 53693689 96410767 66524824
2 3 4 5 6 7 8 9 10	2 3 4 5 6 7 8 9 9	Andy Andy Jim David Jessica Lily Mary Tiger	Male       Male       Male       Male       Male       Male       Male       Female       Female       Female       Male	Vice Presid Sale Sales Man Sales Man Sale Sale Inside Sale Sale General M	1962-02-1 1973-08-3 1968-09-1 1965-03-0 1967-07-0 1960-05-2 1969-01-0 1969-07-0 1970-10-1	964107677 665248245 650028860 441380247 860916757 568899664 428844667 962447951 255490864	2500.0 10800.0 7500.0 4700.0 9300.0 2900.0 3500.0 7200.0 4300.0		5 6 1 2 3 4	ID 11 12 3 5	Ed Charlie Name Mike Jake Lucy Jim	Male Female Gender Female Male Female Male	R&D Leader Engineer Post Sale Vice Presid Sale Sales Man	1983-02-1 1979-01-0 Birthday 1968-12-0 1962-02-1 1973-08-3 1965-03-0	12828497 65254835 39427476 AccountN 53693689 96410767 66524824 44138024

6 7

8

9

10

8 Lily

9 Mary

11 Kate

12 AI

Female

Female

Female

Male

Sale

Sale

Inside Sale... 1969-01-0... 428844667.

Human Re... 1980-04-0... 683984106..

1969-07-0... 962447951...

1980-03-0... 739029848..

# sort vs psort



#### Sequence A

Sort is to sort sequence members and return sequence members;

Psort is to sort sequence members and return sequence of member positions;

So A(C)=B

	Α	В
1	=file("E:/txt/stock1001_price.txt").import@t()	
2	=A1.psort@z(CL)	/Return the sequence of No.s after reverse sorting
3	=A2(to(3))	/Take the first three
4	=A1.calc(A3,if(day(DT)==1,CL,CL/CL[-1]-1))	/Calculate the increase rate

#### A1~A4 results:

Index	stockid	DT	CL
1	1001	2009-01-01	4.0
2	1001	2009-01-02	3.64
3	1001	2009-01-05	3.95
4	1001	2009-01-06	3.68
5	1001	2009-01-07	3.53
6	1001	2009-01-08	3.59
7	1001	2009-01-09	3.9
8	1001	2009-01-12	3.56
9	1001	2009-01-13	3.22
10	1001	2009-01-14	3.17

Index	Member
1	18
2	22
3	21
4	19
5	17
6	20
7	16
8	1
9	3
10	14

Index	Member		
1		18	
2		22	
3		21	

Index	Member
1	0.09691629955947145
2	0.061833688699360234
3	0.08816705336426933

# psort vs ranks



The results of sorting can be reused by psort function, and sorting and ranking can be obtained by one sorting.

				Α			В						
1	[7,15,8,9	),23,5	,22,5,5,5]										
2	=A1.pso	rt()			/Ret	/Return the sequence of No.s after reverse sorting							
3	=to(A1.len())												
4	>A2.run	(if(A1	(~)!=A1(~[	-1]),A3(	/lf d	/If duplicated, the ranking remains unchanged							
5	=A1.rank	<s()< th=""><th></th><th></th><th></th><th></th><th></th><th>/Cal</th><th colspan="5">/Calculate ranks</th></s()<>						/Cal	/Calculate ranks				
		A1		A2	2		Ļ			A	5		
		Index	Member	Index	Member	Index	Member	Inde	Member	Index	Member		
		2	15	2	8	2	2		2 8	2			
		3	8	3	9	3	3		3 6	3	6		
		4	9	4	10	4	4		4 7	4	7		
		5	23	5	1	5	5		5 10	5	10		
		7	22	7	4	7	7		7 9	7	9		
		8	5	8	2	8	8		3 1	8	1		
		9	5	9	7	9	9	9	) 1	9	1		
		10	5	10	5	10	10	1	) 1	10	1		

#### A3 when A3 executed A3 after A4 executed

Use the psort function to create an index. Check the information that Name is "Andy".



1   2 = 3 =	[beijing,tian an,chongqii xi,zhejiang,	jin,he ng,gu	ilongjian	a.jilin.liaoni											,
2 : 3 :		1 [beijing,tianjin,heilongjiang,jilin,liaoning,neimenggu,xinjiang,ningxia,gansu,qinghai,shaanxi,xizang,sichu an,chongqing,guizhou,yunnan,shanxi,hebei,shandong,henan,anhui,jiangsu,shanghai,hubei,hunan,jiang xi,zhejiang,fujian,taiwan,guangxi,guangdong,hainan,xianggang,aomen]										g,sichu /Pro an,jiang	<pre>     /Provinces in geographical order g </pre>		
3 =	=file("E:/txt/population.txt").import@t()										/Po	/Population ranking table of provinces			
	=file("E:/txt/	GDP.	.txt").impo	ort@t()								/GE	OP ranking ta	ble	of provinces
4 :	=A1.(A2.select@1(province==A1.~)) /Filter population table in a specified order														
5 ;	=A1.(A3.se	lect@	1(provine	ce==A1.~))	)							/Fil	ter GDP tabl	e in	a specified order
6 :	6 =A2.align(A1,province)														
7:	=A3.align(A	1.pro	vince)												
A1~A	A7 results	8:						The sa	ime as A4			The sar	ne as A5	pr beiji	ovince 2018_GDP
Index N	Member	Index	province	Population	Index	province	2018_GDP	Index	province	Population	Index	Member	Ir	dex	Member
1 beijin	ng	1	guangdong	10430.03	1	guangdong	9.73	1	beijing	1961.2	1	[beijing,3.03]		28	[fujian,3.58]
2 <u>tianjir</u>	in	2	shandong	9579.31	2	jiangsu	9.26	2	tianjin	1293.82	2	[tianjin,1.88]		29	(null)
3 heilor	ongjiang	3	henan	9402.36	3	shandong	7.63	3	heilongjiang	3831.22	3	[heilongjiang,1.67]		30	guangxi,2.04]
4 jilin		4	sichuan	8041.82	4	zhejiang	5.54	4	Jilin	2746.22	4	[jilin,1.6]		31	guangdong,9.73]
5 liaoni		5	jiangsu babai	7865.99	5	nenan	4.81	5	neimenaau	4374.03	5	[liaoning,2.53]		32	hainan,0.48]
7 viniio	ang	0	hunon	6560.27	0	hubei	4.07	7	vinijana	2470.05	6	[neimenggu,1.73]		33	(null)

	Α	В
1	[beijing,tianjin,heilongjiang,jilin,liaoning,neimenggu,xinjiang,ningxia,gansu,qinghai,shaanxi,xizan g,sichuan,chongqing,guizhou,yunnan,shanxi,hebei,shandong,henan,anhui,jiangsu,shanghai,hub ei,hunan,jiangxi,zhejiang,fujian,taiwan,guangxi,guangdong,hainan,xianggang,aomen]	/Provinces in geographical order
2	=file("E:/txt/population.txt").import@t()	/Population ranking table of provinces
3	=file("E:/txt/GDP.txt").import@t()	/GDP ranking table of provinces
4	=A2.align(A1,province)	/Filter population table in a specified order
5	=A3.align(A1,province)	/Filter GDP table in a specified order
6	=A1.new(~:province,A5(#).#2/A4(#).#2*10000:Per_capita_GPD)	/The result is obtained by alignment calculation.



A3 lacks GDP data for Hong Kong, Macao and Taiwan, so null is included in the results.

Index	province	Per_capita_GPD
28	fujian	10.07882882882883
29	taiwan	(null)
30	guangxi	4.432219629518583
31	guangdong	9.328832227711713
32	hainan	5.535374502681196
33	xianggang	(null)
34	aomen	(null)

