润乾
RAQSOFT

结构化文本计算

润乾软件 - 集算器

www.raqsoft.com.cn

结构化理解

结构化文本，即行文本，每行对应一条记录，各行的字段数一样，相当于数据库中的一张二维表。下面是几个常见的结构化文本。

txt格式，”\t”分割，有标题

CLASS	NAME	English	Chinese	Math
1	Adams Brooke	63	31	69
1	Adams Hannah	89	85	79
1	Adams Jonathan	88	87	91
1	Allen Ashley	98	97	97
1	Allen Brandon	93	76	78
1	Baker Danielle	83	40	95
1	Brown Amanda	94	59	81

csv格式，”，”分割，无标题

```
498765,5431438,2019-03-12,2019-04-12,138.5903
34524,5443211,2019-03-15,2019-04-15,208.0805
821741,5461707,2019-03-22,2019-04-22,421.2097
263534,5472320,2019-03-26,2019-04-26,212.6537
238853,5459750,2019-03-21,2019-04-21,817.4593
21071,5393299,2019-03-01,2019-04-01,112.0961
415214,5342607,2019-02-16,2019-03-16,947.1053
224972,5472584,2019-03-26,2019-04-26,133.766
```

txt格式，”|”分割，有标题

```
EID|NAME|SURNAME|GENDER|STATE|BIRTHDAY|HIREDATE|DEPT|SALARY
1|Rebecca|Moore|F|California|1974-11-20|2005-03-11|R&D|7000
2|Ashley|Wilson|F|New York|1980-07-19|2008-03-16|Finance|11000
3|Rachel|Johnson|F|New Mexico|1970-12-17|2010-12-01|Sales|9000
4|Emily|Smith|F|Texas|1985-03-07|2006-08-15|HR|7000
5|Ashley|Smith|F|Texas|1975-05-13|2004-07-30|R&D|16000
6|Matthew|Johnson|M|California|1984-07-07|2005-07-07|Sales|11000
7|Alexis|Smith|F|Illinois|1972-08-16|2002-08-16|Sales|9000
8|Megan|Wilson|F|California|1979-04-19|1984-04-19|Marketing|11000
9|Victoria|Davis|F|Texas|1983-12-07|2009-12-07|HR|3000
```

目录

CONTENTS

01

单文件基本运算

- 过滤
- 聚合
- 计算列
- 文件读取

02

单文件高级运算

- 排序
- 分组汇总
- 去重
- 并行计算

03

关联计算

- 连接理解
- 文件关联
- 集合运算

04

SQL与命令行

- 单表SQL
- 连接与子查询
- 命令行

05

合并与拆分

- 合并
- 拆分

目录 CONTENTS

1. 单文件基本运算
2. 单文件高级运算
3. 关联计算
4. SQL与命令行
5. 合并与拆分

01

单文件基本计算

过滤

小文件过滤，筛选出10班的学生成绩

	A	B
1	=file("E:/txt/students_scores.txt").import@t()	/@t选项，把第一行读作标题，默认”\t”分割
2	=A1.select(CLASS==10)	/筛选出10班的学生成绩,立即计算

文本内容

CLASS	NAME	English	Chinese	Math
1	Adams Brooke	63	31	69
1	Adams Hannah	89	85	79
1	Adams Jonathan	88	87	91
1	Allen Ashley	98	97	97

A2结果

Index	CLASS	NAME	English	Chinese	Math
1	10	Adams Ashl...	89	49	91
2	10	Adams Kayla	85	74	45
3	10	Allen Danielle	62	77	88
4	10	Allen Samuel	85	51	57
5	10	Anderson D...	53	74	50

大文件过滤，筛选出10班的学生成绩

	A	B
1	=file("E:/txt/students_scores.txt").cursor@t()	/@t选项，把第一行读作标题
2	=A1.select(CLASS==10)	/筛选出10班的学生成绩，延迟计算
3	=A2.fetch()	/从游标中取数，同时执行A2中的附加计算

A1~A3结果

Value
com.raqsoft.dm.cursor.FileCursor@bcce68f

Value
com.raqsoft.dm.cursor.FileCursor@bcce68f

Index	CLASS	NAME	English	Chinese	Math
1	10	Adams Ashl...	89	49	91
2	10	Adams Kayla	85	74	45
3	10	Allen Danielle	62	77	88
4	10	Allen Samuel	85	51	57
5	10	Anderson D...	53	74	50

聚合

小文件聚合，计算语文的总分

	A	B
1	=file(“E:/txt/students_scores.csv”).import@t(;,”,”)	/SPL可以指定文件分割符，如这里的”,”
2	=A1.sum(Chinese)	/计算语文成绩总分

文本内容

CLASS,NAME,English,Chinese,Math
1,Adams Brooke,63,31,69
1,Adams Hannah,89,85,79
1,Adams Jonathan,88,87,91
1,Allen Ashley,98,97,97

A2结果

Value
181025

大文件聚合，计算语文的总分

	A	B
1	=file("E:/txt/students_scores.csv").cursor@tc()	/当分割符是”,”时，使用@c即可
2	=A1.total(sum(Chinese))	/计算语文成绩总分

A1、A2结果

Value
com.raqsoft.dm.cursor.FileCursor@56225d7

Value
181025

计算列

小文件计算列计算，计算学生的总分

	A	B
1	=file("E:/txt/students_scores_.txt").import@t(;,"|")	/文件以"|"分割，SPL可以指定分割符
2	=A1.derive(English+Chinese+Math:total_score)	/增加一列学生的总分

文本内容

```
CLASS|NAME|English|Chinese|Math
1|Adams Brooke|63|31|69
1|Adams Hannah|89|85|79
1|Adams Jonathan|88|87|91
1|Allen Ashley|98|97|97
```

A2的结果

Index	CLASS	NAME	English	Chinese	Math	total_score
1	1	Adams Bro...	63	31	69	163
2	1	Adams Han...	89	85	79	253
3	1	Adams Jon...	88	87	91	266
4	1	Allen Ashley	98	97	97	292
5	1	Allen Brand...	93	76	78	247

大文件计算列计算，计算学生的总分

	A	B
1	=file("E:/txt/students_scores_.txt").cursor@t(;,"|")	/文件以"|"分割，SPL可以指定分割符
2	=A1.derive(English+Chinese+Math:total_score)	/附加derive计算总分，返回游标
3	=A2.fetch@x(100)	/取数并执行计算，关闭游标

A3的结果

Index	CLASS	NAME	English	Chinese	Math	total_score
1	1	Adams Bro...	63	31	69	163
2	1	Adams Han...	89	85	79	253
3	1	Adams Jon...	88	87	91	266
4	1	Allen Ashley	98	97	97	292
5	1	Allen Brand...	93	76	78	247

综合计算

小文件综合计算，

计算10班学生的语文平均分和语文及格的学生的平均分

	A	B
1	=file("E:/txt/students_scores_.txt").import@t(CLASS,Chinese;,"|")	/文件以"|"分割,取Class和Chinese
2	=A1.select(CLASS==10)	/筛选出10班的成绩
3	=[A2.avg(Chinese),A2.avg(if(Chinese>=60,Chinese))]	/计算总平均和及格学生的平均

文本内容

```
CLASS|NAME|English|Chinese|Math
1|Adams Brooke|63|31|69
1|Adams Hannah|89|85|79
1|Adams Jonathan|88|87|91
1|Allen Ashley|98|97|97
```

A2、A3的结果

Index	CLASS	Chinese
1	10	49
2	10	74
3	10	77
4	10	51
5	10	74

Index	Member
1	62.666666666666664
2	78.70588235294117

大文件综合计算，

计算10班学生的语文平均分和语文及格的学生的平均分

	A	B
1	=file("E:/txt/students_scores_.txt").cursor@t(CLASS,Chinese;,"|")	/游标读取Class和Chinese
2	=A1.select(CLASS==10)	/附加select计算
3	=A2.total(avg(Chinese),avg(if(Chinese>=60,Chinese)))	/计算总平均和几个学生的平均

A3的结果

Index	Member
1	62.666666666666664
2	78.70588235294117

问题一：指定字段分隔符

文件内容

CLASS,NAME,English,Chinese,Math
1,Adams Brooke,63,31,69
1,Adams Hannah,89,85,79
1,Adams Jonathan,88,87,91
1,Allen Ashley,98,97,97

CLASS|NAME|English|Chinese|Math
1|Adams Brooke|63|31|69
1|Adams Hannah|89|85|79
1|Adams Jonathan|88|87|91
1|Allen Ashley|98|97|97

SPL代码

“,” 分割

	A
1	=file(path).import@t(;,“,”)
2	=file(path).import@tc()

“|” 分割

	A
1	=file(path).import@t(;,"|")

SPL输出

Index	CLASS	NAME	English	Chinese	Math
1	1	Adams Bro...	63	31	69
2	1	Adams Ha...	89	85	79
3	1	Adams Jon...	88	87	91
4	1	Allen Ashley	98	97	97

Index	CLASS	NAME	English	Chinese	Math
1	1	Adams Bro...	63	31	69
2	1	Adams Ha...	89	85	79
3	1	Adams Jon...	88	87	91
4	1	Allen Ashley	98	97	97

问题二：第一行就是内容，没有标题

文件内容

```
1	Adams Brooke	63	31	69
1	Adams Hannah	89	85	79
1	Adams Jonathan	88	87	91
1	Allen Ashley	98	97	97
1	Allen Brandon	93	76	78
1	Baker Danielle	83	40	95
```

SPL代码

没有标题

	A
1	=file(path).import()

SPL输出

Index	_1	_2	_3	_4	_5
1	1	Adams Bro...	63	31	69
2	1	Adams Ha...	89	85	79
3	1	Adams Jon...	88	87	91
4	1	Allen Ashley	98	97	97
5	1	Allen Bran...	93	76	78

问题三：自动识别的字段类型或日期格式不正确

文件内容

```
user_id,gender,age,insertdate
483833,M,19,2018/1/11
156772,M,31,2018/1/13
173388,M,34,2018/1/21
199107,F,25,2018/1/5
122560,M,23,2018/1/2
```

SPL代码

user_id应为字符串　日期格式：yyyy/MM/dd

	A 正常读取，run函数修改
1	=file(path).import@t()
2	=A2.run(user_id=string(user_id),insertdate=date(insertdate,"yyyy/MM/dd"))

	A 正确读取
1	=file(path).import@t(user_id:string,gender,age,insertdate:date:"yyyy/MM/dd")

SPL输出

Index	user_id	gender	age	insertdate
1	483833	M	19	2018/1/11
2	156772	M	31	2018/1/13
3	173388	M	34	2018/1/21
4	199107	F	25	2018/1/5
5	122560	M	23	2018/1/2

正常读取

Index	user_id	gender	age	insertdate
1	483833	M	19	2018-01-11
2	156772	M	31	2018-01-13
3	173388	M	34	2018-01-21
4	199107	F	25	2018-01-05
5	122560	M	23	2018-01-02

run函数修改

Index	user_id	gender	age	insertdate
1	483833	M	19	2018-01-11
2	156772	M	31	2018-01-13
3	173388	M	34	2018-01-21
4	199107	F	25	2018-01-05
5	122560	M	23	2018-01-02

正确读取

问题四：读取部分字段

文件内容

CLASS	NAME	English	Chinese	Math
1	Adams Brooke		63	31	69
1	Adams Hannah		89	85	79
1	Adams Jonathan		88	87	91
1	Allen Ashley		98	97	97
1	Allen Brandon		93	76	78

SPL代码

	A
1	=file(path).import@t(CLASS,Chinese)
2	=file(path).import@t(#1,#4)

SPL输出

Index	CLASS	Chinese
1	1	31
2	1	85
3	1	87
4	1	97
5	1	76

问题五：字符集

文件内容

```
user_id,reg_mon,gender,age,cell_province,id_province,id_city,insertdate
483833,2017-04,男,19,c29,c26,c26241,2018-12-11
156772,2016-05,男,31,c11,c11,c11159,2018-02-13
173388,2016-05,男,34,c02,c02,c02182,2018-08-21
199107,2016-07,女,25,c09,c09,c09046,2018-06-05
122560,2016-03,男,23,c05,c05,c05193,2018-04-02
```

SPL代码

	A
1	=file(path).import@tc()

正常读取

	A
1	=file(path:"utf-8").import@tc()

指定字符集读取

SPL输出

正常读取

Index	user_id	reg_mon	gender	age	cell_provin...	id_province	id_city	insertdate
1	483833	2017-04	鐢□	19	c29	c26	c26241	2018-12-11
2	156772	2016-05	鐢□	31	c11	c11	c11159	2018-02-13
3	173388	2016-05	鐢□	34	c02	c02	c02182	2018-08-21
4	199107	2016-07	瀛□	25	c09	c09	c09046	2018-06-05
5	122560	2016-03	鐢□	23	c05	c05	c05193	2018-04-02

指定字符集读取

Index	user_id	reg_mon	gender	age	cell_provin...	id_province	id_city	insertdate
1	483833	2017-04	男	19	c29	c26	c26241	2018-12-11
2	156772	2016-05	男	31	c11	c11	c11159	2018-02-13
3	173388	2016-05	男	34	c02	c02	c02182	2018-08-21
4	199107	2016-07	女	25	c09	c09	c09046	2018-06-05
5	122560	2016-03	男	23	c05	c05	c05193	2018-04-02

目录 CONTENTS

1. 单文件基本运算
2. 单文件高级运算
3. 关联计算
4. SQL与命令行
5. 合并与拆分

02

单文件高级计算

排序

小文件排序一：将学生成绩按照语文升序排序

	A	B
1	=file("E:/txt/students_score.txt").import@t()	/读取文件
2	=A1.sort(Chinese)	/升序排序

A1、A2的结果

Index	Name	Math	Chinese	English
1	Natalie	84	90	84
2	Jessica	87	88	78
3	Brianna	89	90	75

Index	Name	Math	Chinese	English
1	Hannah	90	76	95
2	Tyler	87	78	93
3	Zachary	75	81	85

大文件排序一：将学生成绩按照语文升序排序

	A	B
1	=file("E:/txt/students_score.txt").cursor@t()	/创建游标
2	=A1.sortx(Chinese)	/升序排序，返回游标
3	=A2.fetch@x(100)	/取数

A3的结果

Index	CLASS	NAME	English	Chinese	Math	total_score
1	1	Adams Bro...	63	31	69	163
2	1	Adams Han...	89	85	79	253
3	1	Adams Jon...	88	87	91	266
4	1	Allen Ashley	98	97	97	292
5	1	Allen Brand...	93	76	78	247

排序

小文件排序二：将学生成绩按照总分降序排序

	A	B
1	=file("E:/txt/students_score.txt").import@t()	/读取文件
2	=A1.sort@z(Math+English+Chinese)	/计算列降序排序

A1、A2的结果

Index	Name	Math	Chinese	English
1	Natalie	84	90	84
2	Jessica	87	88	78
3	Brianna	89	90	75

Index	Name	Math	Chinese	English
1	Emma	88	84	94
2	Sean	98	86	81
3	Hannah	90	76	95

大文件排序二：将学生成绩按照总分降序排序

	A	B
1	=file("E:/txt/students_score.txt").cursor@t()	/创建游标
2	=A1.sortx(-(Math+English+Chinese))	/计算列降序排序，返回游标
3	=A2.fetch@x(100)	/取数

A3的结果

Index	Name	Math	Chinese	English
1	Emma	88	84	94
2	Sean	98	86	81
3	Hannah	90	76	95

排序

小文件排序三：将学生按照班级升序，总分降序排序

	A	B
1	=file("E:/txt/students_scores.txt").import@t()	/读取文件
2	=A1.sort(CLASS,-(English+Chinese+Math))	/按照班级升序，总分降序

A1、A2的结果

Index	CLASS	NAME	English	Chinese	Math
1	1	Adams Bro...	63	31	69
2	1	Adams Ha...	89	85	79
3	1	Adams Jon...	88	87	91

Index	CLASS	NAME	English	Chinese	Math
1	1	Allen Ashley	98	97	97
2	1	Lewis Anto...	93	92	94
3	1	Adams Jon...	88	87	91

大文件排序三：将学生按照班级升序，总分降序排序

	A	B
1	=file("E:/txt/students_scores.txt").cursor@t()	/创建游标
2	=A1.sortx(CLASS,-(English+Chinese+Math))	/按需求排序，返回游标
3	=A2.fetch@x(100)	/取数

A3的结果

Index	CLASS	NAME	English	Chinese	Math
1	1	Allen Ashley	98	97	97
2	1	Lewis Anto...	93	92	94
3	1	Adams Jon...	88	87	91

小文件分组聚合

例：统计各省的用户登录次数

	A	B
1	=file("E:/txt/user_info_reg.csv").import@tc()	/读取文件
2	=A1.groups(id_province;count(~):cnt)	/分组后count

A1、A2的结果

Index	user_id	reg_mon	age	cell_provin...	id_province	id_city	insertdate	reg_time
1	483833	2017-04	19	c29	c26	c26241	2018-12-11	56558
2	156772	2016-05	31	c11	c11	c11159	2018-02-13	81617
3	173388	2016-05	34	c02	c02	c02182	2018-08-21	729
4	199107	2016-07	25	c09	c09	c09046	2018-06-05	86299
5	122560	2016-03	23	c05	c05	c05193	2018-04-02	2657

Index	id_province	cnt
1	c01	27202
2	c02	61735
3	c03	14433
4	c04	100639
5	c05	48326

大文件分组聚合（小结果集）

例：统计各省的用户登录次数

	A	B
1	=file("E:/txt/user_info_reg.csv").cursor@tc()	/创建游标
2	=A1.groups(id_province;count(~):cnt)	/分组后count

A1、A2的结果

Value
com.raqsoft.dm.cursor.FileCursor@5ae3860d

Index	id_province	cnt
1	c01	27202
2	c02	61735
3	c03	14433
4	c04	100639
5	c05	48326

大文件分组聚合（大结果集）

例：统计每个用户的登录总时长

	A	B
1	=file("E:/txt/user_info_reg.csv").cursor@tc()	/创建游标
2	=A1.groupx(user_id;sum(reg_time):total_reg)	/分组后sum，返回游标
3	=A2.fetch(1000)	

A1~A3的结果

Value
com.raqsoft.dm.cursor.FileCursor@3a544c70

Value
com.raqsoft.dm.cursor.MemoryCursor@3bd310fd

Index	user_id	total_reg
1	1	2345
2	2	74990
3	3	53724
4	4	47153
5	5	23507

小文件分组后过滤

例：找出登录总时长低于1000分钟的用户

	A	B
1	=file("E:/txt/user_info_reg.csv").import@tc()	/读取文件
2	=A1.groups(user_id;sum(reg_time):total_reg)	/分组后sum
3	=A2.select(total_reg<1000)	/过滤

A1~A3的结果

Index	user_id	reg_mon	age	cell_province	id_province	id_city	insertdate	reg_time
1	483833	2017-04	19	c29	c26	c26241	2018-12-11	56558
2	156772	2016-05	31	c11	c11	c11159	2018-02-13	81617
3	173388	2016-05	34	c02	c02	c02182	2018-08-21	729
4	199107	2016-07	25	c09	c09	c09046	2018-06-05	86299
5	122560	2016-03	23	c05	c05	c05193	2018-04-02	2657

Index	user_id	total_reg
1	1	2345
2	2	74990
3	3	53724
4	4	47153
5	5	23507

Index	user_id	total_reg
1	41	512
2	68	130
3	90	486
4	203	865
5	519	556

大文件分组后过滤

例：找出登录总时长低于1000分钟的用户

	A	B
1	=file("E:/txt/user_info_reg.csv").cursor@tc()	/创建游标
2	=A1.groupx(user_id;sum(reg_time):total_reg)	/分组后sum，返回游标
3	=A2.select(total_reg<1000).fetch()	/过滤，取数

A1~A3的结果

Index	user_id	reg_mon	age	cell_province	id_province	id_city	insertdate	reg_time
1	483833	2017-04	19	c29	c26	c26241	2018-12-11	56558
2	156772	2016-05	31	c11	c11	c11159	2018-02-13	81617
3	173388	2016-05	34	c02	c02	c02182	2018-08-21	729
4	199107	2016-07	25	c09	c09	c09046	2018-06-05	86299
5	122560	2016-03	23	c05	c05	c05193	2018-04-02	2657

Index	user_id	total_reg
1	1	2345
2	2	74990
3	3	53724
4	4	47153
5	5	23507

Index	user_id	total_reg
1	41	512
2	68	130
3	90	486
4	203	865
5	519	556

去重

小文件去重，查看所有的用户id

	A	B
1	=file("E:/txt/user_info_reg.csv").import@tc()	/读取指定字段
2	=A1.id(user_id)	/去重，查看用户id

A1~A2的结果

Index	user_id	reg_mon	age	cell_provin...	id_province	id_city	insertdate	reg_time
954205	363648	2017-01	23	c08	c19	c19303	2018-01-17	51365
954206	292977	2016-10	22	c06	c27	c27051	2018-06-13	84647
954207	644550	2017-09	25	c04	c04	c04348	2018-04-10	18970
954208	608246	2017-08	35	c04	c04	c04319	2018-12-06	54282
954209	834041	2018-04	24	c25	c09	c09294	2018-09-11	33953

Index	Member
928191	928191
928192	928192
928193	928193
928194	928194
928195	928195

大文件去重，查看所有的用户id

	A	B
1	=file("E:/txt/user_info_reg.csv").cursor@tc()	/创建游标
2	=A1.id(user_id)	/去重，查看用户id

A2的结果

Index	Member
928191	928191
928192	928192
928193	928193
928194	928194
928195	928195

去重计数

小文件去重计数，

将数据按日期、产品去除重复，再统计记录条数

	A	B
1	=file("E:/txt/PRODUCT_SALE.txt").import@t(DATE,PID)	/读取指定字段
2	=A1.groups(date(DATE),PID)	/去重
3	=A2.len()	/计算非重复记录数

A1~A3的结果

Index	DATE	PID
9999998	2018-12-31	10014923
9999999	2018-12-31	10040866
10000000	2018-12-31	10057996

Index	DATE	PID
9849395	2018-12-31	10099926
9849396	2018-12-31	10099939
9849397	2018-12-31	10099955

Value
9849397

大文件去重计数，

将数据按日期、产品去除重复，再统计记录条数

	A	B
1	=file("E:/txt/PRODUCT_SALE.txt").cursor@t(DATE,PID)	/读取指定字段
2	=A1.groupx(date(DATE),PID)	/去重
3	=A2.skip()	/计算非重复记录数

A3结果

Value
9849397

分组去重计数

小文件分组去重计数，统计每个产品有销售记录的天数

	A	B
1	=file("E:/txt/PRODUCT_SALE.txt").import@t(DATE,PID)	/读取指定字段
2	=A1.groups(PID,date(DATE))	/去重
3	=A2.groups(PID;count(1):no_sdate)	/分组，统计有销售记录的天数

A1~A3的结果

Index	DATE	PID
9999998	2018-12-31	10014923
9999999	2018-12-31	10040866
10000000	2018-12-31	10057996

Index	PID	date(DATE)
9849395	10100001	2018-11-26
9849396	10100001	2018-11-28
9849397	10100001	2018-12-10

Index	PID	no_sdate
99998	10099999	93
99999	10100000	100
100000	10100001	109

大文件分组去重计数，统计每个产品有销售记录的天数

	A	B
1	=file("E:/txt/PRODUCT_SALE.txt").cursor@t(DATE,PID)	/创建游标
2	=A1.groupx(date(DATE),PID)	/去重
3	=A2.groups(PID;count(1):no_sdate)	/分组，统计有销售记录的天数

A3结果

Index	PID	no_sdate
99998	10099999	93
99999	10100000	100
100000	10100001	109

并行过滤，筛选出10月份的产品销售记录（多路游标）

	A
1	=now()
2	=file("E:/txt/PRODUCT_SALE.txt").cursor@t()
3	=A2.select(month(DATE)==10)
4	=A3.fetch(100000)
5	=interval@ms(A1,now())

单游标过滤

A4、A5的结果

Index	ID	PID	DATE	QUANTITY	SID
99999	2051417	10051515	2011-10-02	37	10075
100000	2051976	10056022	2011-10-02	67	10143

Value
1525

	A
1	=now()
2	=file("E:/txt/PRODUCT_SALE.txt").cursor@mt()
2	=file(“E:/txt/PRODUCT_SALE.txt”).cursor@t().mcursor()
3	=A2.select(month(DATE)==10)
4	=A3.fetch(100000)
5	=interval@ms(A1,now())

多路游标过滤，方法一：读取数据和过滤都是并行

多路游标过滤，方法二：只有过滤并行

A4、A5的结果

Index	ID	PID	DATE	QUANTITY	SID
99999	3573602	10028733	2016-10-09	103	10689
100000	3579513	10010320	2016-10-09	35	10074

Value
1088

Index	ID	PID	DATE	QUANTITY	SID
99999	6110232	10037262	2010-10-16	36	10413
100000	6113472	10011967	2010-10-16	54	10663

Value
1261

注意：多路游标除了fetch全部时，返回的结果集可能改变原来的数据次序

并行分组聚合，计算各个产品的销售总量（多路游标）

	A
1	=now()
2	=file("E:/txt/PRODUCT_SALE.txt").cursor@t()
3	=A2.groups(PID;sum(QUANTITY):total_num)
4	=interval@ms(A1,now())

单游标过滤

A4、A5的结果

Index	PID	total_num
1	10000002	5799
2	10000003	6554

Value
10043

	A
1	=now()
2	=file("E:/txt/PRODUCT_SALE.txt").cursor@mt()
2	=file(“E:/txt/PRODUCT_SALE.txt”).cursor@t().mcursor()
3	=A2.groups(PID;sum(QUANTITY):total_num)
4	=interval@ms(A1,now())

多路游标过滤，方法一：读取数据和过滤都是并行

多路游标过滤，方法二：只有过滤并行

A4、A5的结果

Index	PID	total_num
1	10000002	5799
2	10000003	6554

Value
4054

Index	PID	total_num
1	10000002	5799
2	10000003	6554

Value
7192

目录 CONTENTS

1. 单文件基本运算
2. 单文件高级运算
3. 关联计算
4. SQL与命令行
5. 合并与拆分

03

文件关联

SQL连接

1. 笛卡尔积

Employee

ID	NAME	DEPT
1	David	1
2	Daniel	2
3	Andrew	1

×

Department

ID	NAME
1	Sales
2	R&D

=

ID	NAME	DEPT	ID	NAME
1	David	1	1	Sales
1	David	1	2	R&D
2	Daniel	2	1	Sales
2	Daniel	2	2	R&D
3	Andrew	1	1	Sales
3	Andrew	1	2	R&D

2. 条件过滤

ID	NAME	DEPT	ID	NAME
1	David	1	1	Sales
~~1~~	~~David~~	~~1~~	~~2~~	~~R&D~~
~~2~~	~~Daniel~~	~~2~~	~~1~~	~~Sales~~
2	Daniel	2	2	R&D
3	Andrew	1	1	Sales
~~3~~	~~Andrew~~	~~1~~	~~2~~	~~R&D~~

Employee.DEPT = Department.ID →

ID	NAME	DEPT	ID	NAME
1	David	1	1	Sales
2	Daniel	2	2	R&D
3	Andrew	1	1	Sales

SPL连接

Employee

ID	NAME	DEPT
1	David	1
2	Daniel	2
3	Andrew	1

JOIN/SWITHCH

Department

ID	NAME
1	Sales
2	R&D

Employee	Department
[1, David, 1]	[1, Sales]
[2, Daniel, 2]	[2, R&D]
[3, Andrew, 1]	[1, Sales]

SPL将两个或多个集合连接后，以集合成员构成的二元组为成员，而不是简单的展开所有集合的数据结构拼在一起。SPL的做法不仅更符合JOIN的概念和原意，表间关系更加清晰可见，语法也比SQL更加简洁。

文件关联

两个小文件外键关联一 例：找出员工中夫妻年龄和超过80的员工

	A	B
1	=file("E:\\txt\\Employees.txt").import@t().keys(ID)	/将ID设置为主键
2	=file("E:\\txt\\EmpRel.txt").import@t()	
3	=A2.select(Relationship=="Spouse")	/筛选出A2表中的夫妻关系
4	>A3.switch(Emp1,A1;Emp2,A1)	/将雇员关系表中的两个雇员字段都替换为相应的记录
5	=A3.select(age(Emp1.Birthday)+age(Emp2.Birthday)>80)	/筛选出年龄和大于80的记录
6	>A5.run(Emp1=Emp1.Name,Emp2=Emp2.Name)	/将记录改成记录的Name字段

A1

Index	ID	Name	Gender	Post	Birthday	AccountNo	BasePay
1	1	Mike	Female	Sale	1968-12-0...	536936891...	5600.0
2	2	Jake	Male	Vice Presid...	1962-02-1...	964107677...	2500.0
3	3	Lucy	Female	Sale	1973-08-3...	665248245...	10800.0
4	4	Andy	Male	Sales Man...	1968-09-1...	650028860...	7500.0
5	5	Jim	Male	Sales Man...	1965-03-0...	441380247...	4700.0

A2

Index	Emp1	Emp2	Relationship
1	21	22	Spouse
2	10	1	Spouse
3	5	19	Spouse
4	16	3	Spouse

A3执行后的A3

Index	Emp1	Emp2	Relationship
1	21	22	Spouse
2	10	1	Spouse
3	5	19	Spouse
4	16	3	Spouse

外键对象化

Index	Emp1	Emp2	Relationship
1	21	22	Spouse
2	10	1	Spouse
3	5	19	Spouse
4	16	3	Spouse

ID	Name	Gender	Post	Birthday	AccountNo	BasePay
22	Ken	Female	Sale	1982-07-1...	824387323...	3200.0

ID	Name	Gender	Post	Birthday	AccountNo	BasePay
21	Joe	Male	R&D Leader	1984-09-1...	528924335...	3500.0

A4执行后的A3

Index	Emp1	Emp2	Relationship
1	10	1	Spouse
2	5	19	Spouse
3	16	3	Spouse

A5执行后的A5

Index	Emp1	Emp2	Relationship
1	Tiger	Mike	Spouse
2	Jim	Howard	Spouse
3	Ed	Lucy	Spouse

A6执行后的A5

两个小文件外键关联二 例：找出部门经理最年轻的部门

	A	B
1	=file("E:/txt/EMPLOYEE.txt").import@t()	/读入员工信息
2	=file("E:/txt/DEPARTMENT.txt").import@t()	/读入部门信息
3	=A2.join(MANAGER,A1:EID,~:manager)	/员工信息外键对象化与部门信息关联
4	=A3.minp(manager.(age(BIRTHDAY))).manager.DEPT	/找到经理最年轻的部门

A1

Index	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
1	1	Rebecca	Moore	F	California	1974-11-20	2005-03-11	R&D	7000
2	2	Ashley	Wilson	F	New York	1980-07-19	2008-03-16	Finance	11000
3	3	Rachel	Johnson	F	New Mexico	1970-12-17	2010-12-01	Sales	9000
4	4	Emily	Smith	F	Texas	1985-03-07	2006-08-15	HR	7000
5	5	Ashley	Smith	F	Texas	1975-05-13	2004-07-30	R&D	16000

A2

Index	DEPT	MANAGER
1	Administrat...	20
2	Finance	2
3	HR	162
4	Marketing	47
5	Production	58
6	R&D	5
7	Sales	40
8	Technology	55

A3

Index	DEPT	MANAGER	manager
1	Administration	20	[20,Alexis,Alle...
2	Finance	2	[2,Ashley,Wils...
3	HR	162	[162,Gabriel,...
4	Marketing	47	[47,Elizabeth
5	Production	58	[5
6	R&D	5	[5
7	Sales	40	[40,Madeline,...
8	Technology	55	[55,Olivia,And...

外键对象化

EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
20	Alexis	Allen	F	Florida	1977-08-07	2007-08-07	Administrat...	16000

A4

Value
Finance

文件关联

两个小文件外键关联三

例：将用户信息表中的用户信息加入到用户借贷信息表中组成宽表

	A	B
1	=file("E:/txt/lending_info.csv").import@tc()	/读取借贷信息
2	=file("E:/txt/user_info.csv":"utf-8").import@tc()	/读取用户信息，字符集是“utf-8”
3	=A2.group@1s(user_id)	/user_id去重，取分组后的第一条，保证主键唯一
4	=A3.fname().m(2:)	/列出除user_id以为的其他用户信息
5	=A1.join(user_id,A3:user_id,${A4.concat@c()})	/两表关联，组成宽表

A1

Index	user_id	listing_id	auditing_d...	due_date	due_amt
129998	233442	5319333	2019-02-10	2019-03-10	168.3364
129999	20165	5336095	2019-02-15	2019-03-15	350.2759
130000	265473	5460170	2019-03-21	2019-04-21	293.8277

A2

Index	user_id	reg_mon	gender	age	cell_province	id_province	id_city	insertdate
954207	644550	2017-09	男	25	c04	c04	c04348	2018-04-10
954208	608246	2017-08	男	35	c04	c04	c04319	2018-12-06
954209	834041	2018-04	男	24	c25	c09	c09294	2018-09-11

A3

Index	user_id	reg_mon	gender	age	cell_provin...	id_province	id_city	insertdate
928193	928193	2019-03	男	23	c07	c07	c07297	2019-03-29
928194	928194	2019-03	男	28	c20	c26	c26243	2019-03-29
928195	928195	2019-03	男	23	c29	c29	c29063	2019-03-30

外键关联要保证主键必须唯一

也就是A2中user_id必须唯一

A5

Index	user_id	listing_id	auditing_d...	due_date	due_amt	reg_mon	gender	age	cell_provin...	id_province	id_city	insertdate
129998	233442	5319333	2019-02-10	2019-03-10	168.3364	2016-08	女	30	c20	c05	c05103	2019-02-09
129999	20165	5336095	2019-02-15	2019-03-15	350.2759	2015-03	男	34	c06	c06	c06195	2018-11-30
130000	265473	5460170	2019-03-21	2019-04-21	293.8277	2016-09	男	29	c13	c13	c13003	2019-03-20

文件关联

一个大文件一个小文件关联一

例：物品信息和销售信息存储在两张表中，请计算销售数量小于10的产品的总销售额

	A	B
1	=file("E:/txt/Products.txt").import@t().primary@i(ID)	/读入商品列表并建立索引，内存表
2	=file("E:/txt/Sales.txt").cursor@t()	/单游标和多路游标都可以
3	=A2.select(quantity<=10)	/游标过滤
方法一 4	=A3.switch(productid,A1:ID)	/swich对游标附加外键对象化
5	=A4.groups(;sum(quantity*productid.Price):total)	/求和汇总
方法二 4	=A3.join(productid,A1:ID,~:products)	/join对游标附加外键对象化
5	=A4.groups(;sum(quantity*products.Price):total)	/求和汇总
方法三 4	=A3.join(productid,A1:ID,Price)	/join拼接Price字段
5	=A4.groups(;sum(quantity*Price):total)	/求和汇总

A1、A5结果

Index	ID	Name	Category	Price
1	1	Apple juice	Low-end	18.0
2	2	Mile	Low-end	19.0
3	3	Tomato sa...	Low-end	10.0
4	4	Salt	Low-end	22.0
5	5	Sesame oil	Low-end	21.35

Index	total
1	142740.18000000008

文件关联

一个大文件一个小文件关联二 例：将用户信息表中的用户信息加入到用户借贷信息表中组成宽表

	A	B
1	=file("E:/txt/lending_info.csv").cursor@tc()	/创建游标
2	=file("E:/txt/user_info.csv":"utf-8").import@tc()	/读取用户信息，字符集是“utf-8”
3	=A2.group@1s(user_id)	/user_id去重，取分组后的第一条
4	=A3.fname().m(2:)	/列出除user_id以为的其他用户信息
5	=A1.join(user_id,A3:user_id,${A4.concat@c()})	/游标外键式连接小表，返回游标
6	=A5.fetch@x(100)	/取100行，关闭游标

A1

Value
com.raqsoft.dm.cursor.FileCursor@158cf9ff

A2

Index	user_id	reg_mon	gender	age	cell_provin...	id_province	id_city	insertdate
1	483833	2017-04	男	19	c29	c26	c26241	2018-12-11
2	156772	2016-05	男	31	c11	c11	c11159	2018-02-13
3	173388	2016-05	男	34	c02	c02	c02182	2018-08-21
4	199107	2016-07	女	25	c09	c09	c09046	2018-06-05
5	122560	2016-03	男	23	c05	c05	c05193	2018-04-02

A6

Index	user_id	listing_id	auditing_d...	due_date	due_amt	reg_mon	gender	age	cell_provin...	id_province	id_city	insertdate
1	498765	5431438	2019-03-12	2019-04-12	138.5903	2017-05	男	37	c11	c11	c11245	2019-03-11
2	34524	5443211	2019-03-15	2019-04-15	208.0805	2015-07	男	26	c25	c25	c25074	2019-03-14
3	821741	5461707	2019-03-22	2019-04-22	421.2097	2018-03	女	25	c22	c22	c22308	2019-03-21
4	263534	5472320	2019-03-26	2019-04-26	212.6537	2016-09	女	42	c17	c17	c17290	2019-03-25
5	238853	5459750	2019-03-21	2019-04-21	817.4593	2016-08	男	44	c10	c26	c26057	2018-12-21

文件关联

两个大文件关联 例：订单表、订单明细表分别存储在两个文件中，计算每个客户的总消费额

	A	B
1	=file(“E:/txt/Orders.txt”).cursor@t().sortx(orderid)	/如果已知数据按照orderid有序则不需要sortx
2	=file("E:/txt/OrderDetails.txt").cursor@t().sortx(orderid)	
3	=joinx(A1:order,orderid;A2: detail,orderid)	/使用joinx实现，两个游标关联
4	=A3.groups(order.clientid:clientid;sum(detail.price):amount)	/计算得到每个客户的消费额

订单表

orderid	clientid	date
10012	100658	2019-02-13
10023	103478	2019-01-12
10040	108013	2019-01-04
10045	100373	2019-01-20
10054	102525	2019-03-07
10057	102740	2019-03-21
10068	107448	2019-03-18
10095	107735	2019-03-27
10108	106552	2019-03-28
10114	108699	2019-01-10
10120	101530	2019-02-15
10134	101134	2019-02-01

订单明细表

orderid	no	productid	price
10012	1	3018	428.5
10012	2	3019	349.2
10023	1	3019	349.2
10040	1	3093	139.5
10040	2	3070	137.9
10040	3	3050	210.6
10045	1	3012	21.8
10054	1	3064	462.5
10057	1	3049	123.5
10057	2	3059	186.1
10068	1	3077	145.8
10068	2	3070	137.9

A4结果

Index	clientid	amount
1	100008	12350.0
2	100011	53400.00000000006
3	100015	13789.999999999976
4	100037	44200.0
5	100042	48380.00000000006
6	100075	27290.000000000044
7	100077	109920.00000000055
8	100083	12479.999999999984
9	100087	48040.00000000009
10	100088	59529.99999999963

小文件的集合运算一　例：根据需求查找社区俱乐部人员

	A	B
1	=file(“E:/txt/running.txt”).import@t().(NAME,SURNAME)	/跑步俱乐部的人
2	=file("E:/txt/ball.txt").import@t().([NAME,SURNAME])	/球类俱乐部的人
3	=A1|A2	/和集，两个俱乐部的人数之和
4	=A1&A2	/并集，至少报名一个俱乐部的人
5	=A1^A2	/交集，两个俱乐部都报名的人
6	=A1\A2	/差集，只报名跑步俱乐部的人

A1

Index	Member
28	[Jacob,Moore]
29	[Jacob,Wilson]
30	[Jonathan,Miller]

A2

Index	Member
34	[Daniel,Smith]
35	[Alyssa,Smith]
36	[Cameron,Johnson]

A3

Index	Member
64	[Daniel,Smith]
65	[Alyssa,Smith]
66	[Cameron,Johnson]

A4

Index	Member
57	[Daniel,Smith]
58	[Alyssa,Smith]
59	[Cameron,Johnson]

A5

Index	Member
5	[Jacob,Moore]
6	[Jacob,Wilson]
7	[Jonathan,Miller]

A6

Index	Member
21	[Abigail,Smith]
22	[Nathan,Johnson]
23	[Joshua,King]

文件关联

小文件的集合运算二 例：用户登录信息分月存储在不同的文件中，根据不同需求查询用户登录信息

	A	B
1	=file("E:/txt/user_login_info_1.txt").import@t().group@1(userid)	/1月份用户的第一次的登录信息
2	=file("E:/txt/user_login_info_2.txt").import@t().group@1(userid)	/2月份用户的第一次的登录信息
3	=file("E:/txt/user_login_info_3.txt").import@t().group@1(userid)	/3月份用户的第一次的登录信息
4	=[A1,A2,A3].merge(userid)	/按照userid有序归并用户各月份的第一次的登录信息
5	=[A1,A2,A3].merge@u(userid)	/并集，3个月内至少登录一次的用户
6	=[A1,A2,A3].merge@i(userid)	/交集，3个月中每个月都登录过的用户
7	=[A1,A2,A3].merge@d(userid)	/差集，只在1月份登陆过的用户

A1

Index	userid	login
93675	699997	2019-01-2
93676	699998	2019-01-1
93677	700000	2019-01-13 14:57:35

A2

Index	userid	login
96997	699998	2019-02-03
96998	699999	2019-02-04
96999	700000	2019-02-12 08:46:00

A3

Index	userid	login
97586	699998	2019-03-2
97587	699999	2019-03-1
97588	700000	2019-03-01 00:52:27

A4

Index	userid	login
288262	700000	2019-03-0
288263	700000	2019-02-12
288264	700000	2019-01-13 14:57:35

A5

Index	userid	login
99996	699998	2019-01-1
99997	699999	2019-02-0
99998	700000	2019-01-13 14:57:35

A6

Index	userid	login
88669	699997	2019-01-2
88670	699998	2019-01-1
88671	700000	2019-01-13 14:57:35

A7

Index	userid	login
64	698338	2019-01-24
65	699398	2019-01-10
66	699763	2019-01-13 14:27:32

文件关联

大文件的集合运算　例：用户登录信息分月存储在不同的文件中，根据不同需求查询用户登录信息

	A	B
1	=file("E:/txt/user_login_info_1.txt").cursor@t().sortx(userid).group@1(userid)	/1,2,3月份用户的第一次的登录信息。 如果已知数据有序，则不需要sortx
2	=file("E:/txt/user_login_info_2.txt").cursor@t().sortx(userid).group@1(userid)	
3	=file("E:/txt/user_login_info_3.txt").cursor@t().sortx(userid).group@1(userid)	
4	=[A1,A2,A3].mergex(userid).fetch()	/按照userid有序归并用户各月份的第一次的登录信息
4	=[A1,A2,A3].mergex@u(userid).fetch()	/并集，3个月内至少登录一次的用户
4	=[A1,A2,A3].mergex@i(userid).fetch()	/交集，3个月中每个月都登录过的用户
4	=[A1,A2,A3].mergex@d(userid).fetch()	/差集，一月份登录，二三月份没登录过的用户

Index	userid	login
288262	700000	2019-03-01 00:52:27
288263	700000	2019-02-12 08:46:00
288264	700000	2019-01-13 14:57:35

Index	userid	login
99996	699998	2019-01-17 18:13:56
99997	699999	2019-02-04 02:58:13
99998	700000	2019-01-13 14:57:35

Index	userid	login
88669	699997	2019-01-27 06:37:22
88670	699998	2019-01-17 18:13:56
88671	700000	2019-01-13 14:57:35

Index	userid	login
64	698338	2019-01-24 09:27:19
65	699398	2019-01-10 02:16:04
66	699763	2019-01-13 14:27:32

目录 CONTENTS

1. 单文件基本运算
2. 单文件高级运算
3. 关联计算
4. SQL与命令行
5. 合并与拆分

04

SQL与命令行

SQL计算结构化文本数据一（过滤）

例：找出10班学生的成绩

	A	B
1	$select * from E:/txt/Students_scores.txt where CLASS=10	/SQL过滤

A1结果

Index	CLASS	NAME	English	Chinese	Math
1	10	Adams Ashley	89	49	91
2	10	Adams Kayla	85	74	45
3	10	Allen Danielle	62	77	88
4	10	Allen Samuel	85	51	57
5	10	Anderson Des...	53	74	50

SQL计算结构化文本数据二（聚合）

例：计算全体学生的语文平均分

	A	B
1	$select avg(Chinese) from E:/txt/Students_scores.txt	/SQL聚合

A1结果

Index	_1
1	62.16517857142857

SQL计算结构化文本数据三（计算列）

例：增加一列学生成绩总分列

	A	B
1	$select *,English+Chinese+Math as total_score from E:/txt/students_scores.txt	/SQL增加计算列

A1结果

Index	CLASS	NAME	English	Chinese	Math	total_score
1	1	Adams Bro...	63	31	69	163
2	1	Adams Han...	89	85	79	253
3	1	Adams Jon...	88	87	91	266
4	1	Allen Ashley	98	97	97	292
5	1	Allen Brand...	93	76	78	247

SQL计算结构化文本数据四（case...when...）

例：增加一列如果英语成绩高于60为及格，其他为不及格

	A	B
1	$select *, case when English>=60 then 'Pass' else 'Fail' end as English_evaluation from E:/txt/students_scores.txt	/SQL增加英语是否及格

A1的结果

序号	CLASS	NAME	English	Chinese	Math	English_evaluation
1	1	Adams Bro...	63	31	69	Pass
2	1	Adams Ha...	89	85	79	Pass
3	1	Adams Jon...	88	87	91	Pass
4	1	Allen Ashley	98	97	97	Pass
5	1	Allen Bran...	93	76	78	Pass

SQL计算结构化文本数据五（排序）

例：按照各班班级升序，总分降序排序

	A	B
1	$select * from E:/txt/students_scores.txt order by CLASS,English+Chinese+Math desc	/按照条件排序

A1的结果

Index	CLASS	NAME	English	Chinese	Math
1	1	Allen Ashley	98	97	97
2	1	Lewis Anto...	93	92	94
3	1	Adams Jon...	88	87	91
4	1	Walker Ja...	84	83	89
5	1	Adams Ha...	89	85	79

SQL计算结构化文本数据六（分组聚合）

例：查询各班的数学平均值

	A	B
1	$select CLASS,avg(English) as avg_En from E:/txt/students_scores.txt group by CLASS	/分组聚合

A1的结果

Index	CLASS	avg_En
1	1	74.43103448275862
2	2	77.34375
3	3	72.72857142857143
4	4	69.6046511627907
5	5	70.34615384615384

SQL计算结构化文本数据七（分组过滤）

例：查询英语平均分低于70的班级

	A	B
1	$select CLASS,avg(English) as avg_En from E:/txt/students_scores.txt group by CLASS having avg(English)<70	/分组过滤

A1的结果

Index	CLASS	avg_En
1	4	69.6046511627907
2	7	69.86

SQL计算结构化文本数据八（去重）

例：查看班级id

	A	B
1	$select distinct(CLASS) from E:/txt/students_scores.txt	/distinct去重

A1的结果

Index	_1
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8

SQL计算结构化文本数据八（去重计数）

例：统计产品的数量

	A	B
1	$select count(distinct PID) from E:/txt/PRODUCT_SALE.txt	/count(distinct)，去重计数

A1的结果

Index	_1
1	100000

SQL计算结构化文本数据九（分组去重计数）

例：统计每个产品有销售记录的天数

	A	B
1	$select PID,count(*) as no_sdate from (select PID from E:/txt/PRODUCT_SALE.txt group by PID,DATE) group by PID	/group+group，分组去重计数
2	$select PID,count(*) as no_sdate from (select distinct PID,DATE from E:/txt/PRODUCT_SALE.txt) group by PID	/distinct+group，分组去重计数

A1的结果

Index	PID	no_sdate
99998	10099999	93
99999	10100000	100
100000	10100001	109

SQL计算结构化文本数据十（文件关联）

例：物品信息和销售信息存储在两张表中，请计算销售数量小于10的产品的总销售额

	A	B
1	$select sum(S.quantity*P.Price) as total from E:/txt/Sales.txt as S join E:/txt/Products.txt as P on S.productid=P.ID where S.quantity<=10	/join，过滤，聚合

A1的结果

Index	total
1	142740.18000000008

SQL计算结构化文本数据十一（文件关联）

例：查询California州的HR部门的员工

	A	B
1	$select e.NAME as NAME from E:/txt/EMPLOYEE_J.txt as e join E:/txt/DEPARTMENT.txt as d on e.DEPTID=d.DEPTID join E:/txt/STATE.txt as s on e.STATEID=s.STATEID where d.NAME='HR' and s.NAME='California'	/单层多个外键join

A1的结果

Index	NAME
1	Gabriel
2	Megan

SQL计算结构化文本数据十二（文件关联）

例：查找经理是California州的New York州员工

	A	B
1	$select e.NAME as ENAME from E:/txt/EMPLOYEE.txt as e join E:/txt/DEPARTMENT.txt as d on e.DEPT=d.NAME join E:/txt/EMPLOYEE.txt as emp on d.MANAGER=emp.EID where e.STATE='New York' and emp.STATE='California'	/多层外键join

A1的结果

Index	ENAME
1	Jessica
2	Alexis
3	Cameron
4	Ashley
5	Brandon
6	Grace
7	Jacob
8	William
9	Matthew
10	Emily

SQL计算结构化文本数据十三（子查询）

例：找出部门经理最年轻的部门

	A
1	$select DEPT from (select emp.BIRTHDAY as BIRTHDAY,emp.DEPT as DEPT from E:/txt/DEPARTMENT.txt as dept left join E:/txt/EMPLOYEE.txt emp on dept.MANAGER=emp.EID) where BIRTHDAY=(select max(BIRTHDAY) from (select emp1.BIRTHDAY as BIRTHDAY from E:/txt/DEPARTMENT.txt as dept1 left join E:/txt/EMPLOYEE.txt as emp1 on dept1.MANAGER=emp1.EID))

A1的结果

Index	DEPT
1	Finance

命令行执行简单SQL（绝对路径）

命令行cd到esProc/bin的目录下(有esprocx.exe)，使用下面的格式执行网格脚本：.\esprocx+空格+”-r”+空格+”SQL”。

例：计算各部门的平均工资

命令行内容

```
.\esprocx -r  “select DEPT,avg(SALARY) from E:/txt/EMPLOYEE.txt group by DEPT"
```

```
PS E:\esproc\esProc\bin> .\esprocx -r "select DEPT,avg(SALARY) from EMPLOYEE.txt group by DEPT"
```

运行结果

```
Administration  10000.0
Finance 7395.833333333333
HR      7263.1578947368425
Marketing       7409.090909090909
Production      7285.714285714285
R&D     8241.379310344828
Sales   7286.096256684492
Technology      7319.148936170212
```

命令行执行简单SQL（相对路径）

例：计算各部门的平均工资

命令行内容

```
.\esprocx -r "select DEPT,avg(SALARY) from EMPLOYEE.txt group by DEPT"
```

```
PS E:\esproc\esProc\bin> .\esprocx -r "select DEPT,avg(SALARY) from EMPLOYEE.txt group by DEPT"
```

注意：处理的文件可以是绝对路径，也可以位于主目录或者搜索目录下

运行结果

```
Administration  10000.0
Finance 7395.833333333333
HR      7263.1578947368425
Marketing       7409.090909090909
Production      7285.714285714285
R&D     8241.379310344828
Sales   7286.096256684492
Technology      7319.148936170212
```

主目录和搜索目录可以在Program菜单Options选项中的Environment选项卡进行设置，如图：

目录 CONTENTS

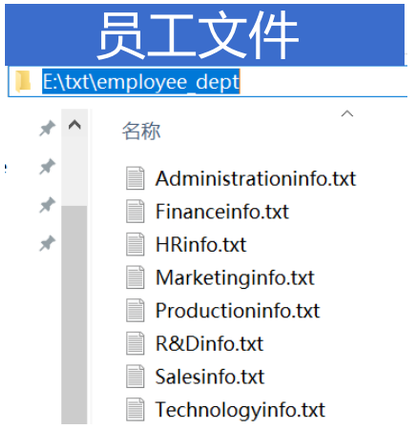
1. 单文件基本运算
2. 单文件高级运算
3. 关联计算
4. SQL与命令行
5. 合并与拆分

05

合并与拆分

多文件合并一

例：各部门员工数据存储在同一目录下的不同文件中，请将员工数据合并并按照员工编号排序，然后导出。



	A	B
1	=directory@p("E:/txt/employee_dept")	/列出文件目录下带有完整路径名的文件
2	=A1.(file(~).import@t())	/读取各部门的员工数据
3	=A2.conj().sort(EID)	/合并并排序
4	=file(“E:/txt/EMPLOYEE.txt”).export@t(A3)	/写出文件

A1

Index	Member
1	E:\txt\employee_dept\Administrationinfo.txt
2	E:\txt\employee_dept\Financeinfo.txt
3	E:\txt\employee_dept\HRinfo.txt
4	E:\txt\employee_dept\Marketinginfo.txt
5	E:\txt\employee_dept\Productioninfo.txt
6	E:\txt\employee_dept\R&Dinfo.txt
7	E:\txt\employee_dept\Salesinfo.txt
8	E:\txt\employee_dept\Technologyinfo.txt

A2

Index	Member
1	[[18,Jonathan,Moore, ...],[20,Alexis,Alle
2	[[2,Ashley,Wilson, ...],[13,Daniel,Davis,
3	Emily,Smith, ...],[9,Victoria,Davis, ...]
4	Megan,Wilson, ...],[17,Hannah,Johr
5	6,Christopher,He
6	[[1,Rebecca,Moore,
7	[[3,Rachel,Johnson
8	[[55,Olivia,Anderso

A3

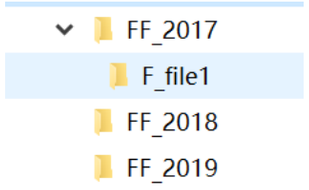
Index	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
1	1	Rebecca	Moore	F	California	1974-11-20	2005-03-11	R&D	7000
2	2	Ashley	Wilson	F	New York	1980-07-19	2008-03-16	Finance	11000
3	3	Rachel	Johnson	F	New Mexico	1970-12-17	2010-12-01	Sales	9000
4	4	Emily	Smith	F	Texas	1985-03-07	2006-08-15	HR	7000
5	5	Ashley	Smith	F	Texas	1975-05-13	2004-07-30	R&D	16000
6	6	Matthew	Johnson	M	California	1984-07-07	2005-07-07	Sales	11000
7	7	Alexis	Smith	F	Illinois	1972-08-16	2002-08-16	Sales	9000
8	8	Megan	Wilson	F	California	1979-04-19	1984-04-19	Marketing	11000

Index	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
1	18	Jonathan	Moore	M	Florida	1971-03-07	2000-03-07	Administrat...	7000
2	20	Alexis	Allen	F	Florida	1977-08-07	2007-08-07	Administrat...	16000
3	26	Timothy	Miller	M	Florida	1977-12-24	2007-12-24	Administrat...	5000
4	42	Michael	Jones	M	Pennsylva...	1978-08-20	2008-08-20	Administrat...	12000

多文件合并二

例：递归读取多级目录，将目录下的文件合并

文件目录



文件内容

FF_file1	1
FF_file1	2
FF_file1	3

	A	B
1	=directory@p(path)	/列出目录下文件名的全目录
2	=A1.(file(~).import())	/导入根目录文件
3	=A2.conj()	/合并结果
4	=file("d:\\result.txt").export@a(A3)	/追加的形式写出
5	=directory@dp(path)	/列出目录下的目录
6	>A5.(call("E:/esproc_test/readfiles.dfx",~))	/递归调用本脚本

合并结果

FF_file1	1
FF_file1	2
FF_file1	3
FF_file2	4
FF_file2	5
FF_file2	6
FF_file3	7
FF_file3	8
FF_file3	9
FF_file1/F_file1	1
FF_file1/F_file1	2
FF_file1/F_file1	3
FF_file1/F_file2	4

A1

Index	Member
1	D:\file\FF_file1.txt
2	D:\file\FF_file2.txt
3	D:\file\FF_file3.txt

A2

Index	Member
1	[[FF_file1,1],[FF_file1,2],[FF_file1,3]]
2	[[FF_file2,4],[FF_file2,5],[FF_file2,6]]
3	[[FF_file3,7],[FF_file3,8],[FF_file3,9]]

A3

Index	_1	_2
1	FF_file1	1
2	FF_file1	2
3	FF_file1	3
4	FF_file2	4
5	FF_file2	5
6	FF_file2	6
7	FF_file3	7
8	FF_file3	8
9	FF_file3	9

A5

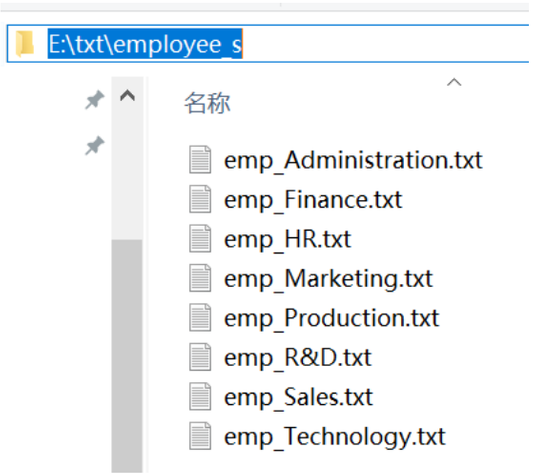
Index	Member
1	D:\file\FF_2017
2	D:\file\FF_2018
3	D:\file\FF_2019

小文件拆分一

例：将员工信息按部门写出到不同的文件。

	A	B
1	=file("E:/txt/EMPLOYEE.txt").import@t()	/读入员工信息
2	=A1.group(DEPT)	/按部门分组
3	=A2.(file("E:/txt/employee_s/emp_"+~.DEPT+".txt").export@t(~))	/命名文件，导出

文件目录



A1

Index	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
1	1	Rebecca	Moore	F	California	1974-11-20	2005-03-11	R&D	7000
2	2	Ashley	Wilson	F	New York	1980-07-19	2008-03-16	Finance	11000
3	3	Rachel	Johnson	F	New Mexico	1970-12-17	2010-12-01	Sales	9000
4	4	Emily	Smith	F	Texas	1985-03-07	2006-08-15	HR	7000
5	5	Ashley	Smith	F	Texas	1975-05-13	2004-07-30	R&D	16000
6	6	Matthew	Johnson	M	California	1984-07-07	2005-07-07	Sales	11000
7	7	Alexis	Smith	F	Illinois	1972-08-16	2002-08-16	Sales	9000
8	8	Megan	Wilson	F	California	1979-04-19	1984-04-19	Marketing	11000

A2

Index	Member
1	[[18,Jonathan,Moore, ...],[20,Alexis,Allen, ...],[26,...
2	[[2,Ashley,Wilson, ...],[13,Daniel,Davis, ...],[23,J...
3	[[4,Emily,Smith, ...],[9,Victoria,Davis, ...
4	[[8,Megan,Wilson, ...],[17,Hannah,Joh...
5	[[16,Christopher,Hernandez, ...],[19,Samantha,...
6	[[1,Rebecca,Moore, ...],[5,Ashley,Smith, ...],[10,...
7	[[3,Rachel,Johnson, ...],[6,Matthew,Johnson, ...]...
8	[[55,Olivia,Anderson, ...],[56,Jacob,Smith, ...],[8...

Index	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
1	18	Jonathan	Moore	M	Florida	1971-03-07	2000-03-07	Administrat...	7000
2	20	Alexis	Allen	F	Florida	1977-08-07	2007-08-07	Administrat...	16000
3	26	Timothy	Miller	M	Florida	1977-12-24	2007-12-24	Administrat...	5000
4	42	Michael	Jones	M	Pennsylva...	1978-08-20	2008-08-20	Administrat...	12000

小文件拆分二 例：含有缺失值和不含缺失值的数据拆分到两个文件中

	A	B
1	=file("E:/txt/EMPLOYEE_nan.txt").import@t()	/导入数据
2	=[true,false]	/确保分成两组
3	=A1.align@a(A2,~.array().pos(null)>0)	/将有无缺失值的数据分成两组
4	=A3.(file("E:/txt/employee_N_s/employee_"+["NA","NO_NA"](#)+".txt").export@t(~))	
	/分别导出两组数据	

A1

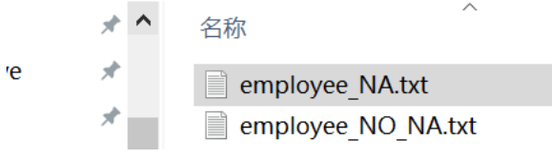
Index	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
175	175	Jasmine	Smith	F	Pennsylva...	1976-03-23	2005-07-01	(null)	7000.0
176	176	Joshua	Miller	M	Mississippi	1979-07-24	2004-05-01	HR	10000.0
177	177	Megan	Johnson	F	Missouri	1978-03-11	(null)	HR	5000.0

A3

Index	Member
1	[[16,Christopher,, ...],[17,Hannah,Johnson, ...],[23,Joseph,, ...], ...
2	[[1,Rebecca,Moore, ...],[2,Ashley,Wilson, ...],[3,Rachel,Johnso...

Index	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
1	16	Christopher	(null)	M	Florida	1979-06-27	2007-06-27	Production	9000.0
2	17	Hannah	Johnson	F	Texas	(null)	2006-07-19	Marketing	4000.0
3	23	Joseph	(null)	M	California	1983-08-27	2003-08-27	Finance	6000.0

文件目录



大文件拆分一　例：将员工信息按部门写出到不同的文件。

	A	B
1	=file("E:/txt/EMPLOYEE.txt").cursor@t()	
2	for A1,100	=A2.group(DEPT)
3		=B2.(file("E:/txt/EMPLOYEE/EMP_"+~.DEPT+".txt").export@at(~))
	/游标读取文件，循环取数，把每次取到的数据按照小文件的处理方式来处理，但导出时使用追加写入@a	

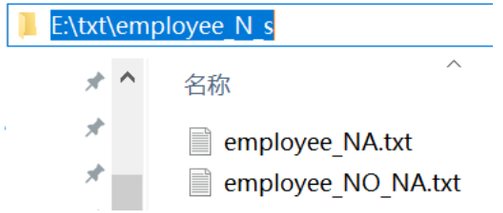
首次循环时的A2、B2

Index	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
1	1	Rebecca	Moore	F	California	1974-11-20	2005-03-11	R&D	7000
2	2	Ashley	Wilson	F	New York	1980-07-19	2008-03-16	Finance	11000
3	3	Rachel	Johnson	F	New Mexico	1970-12-17	2010-12-01	Sales	9000
4	4	Emily	Smith	F	Texas	1985-03-07	2006-08-15	HR	7000
5	5	Ashley	Smith	F	Texas	1975-05-13	2004-07-30	R&D	16000

Index	Member
1	[[18,Jonathan,Moore, ...],[20,Alexis,Allen, ...],[26,Timothy,Miller, ...], ...]
2	[[2,Ashley,Wilson, ...],[13,Daniel,Davis, ...],[23,Joseph,Turner, ...], ...]
3	[[4,Emily,Smith, ...],[9,Victoria,Davis, ...],[51,Madison,Willia
4	[[8,Megan,Wilson, ...],[17,Hannah,Johnson, ...],[21,Jacob,I
5	[[16,Christopher,Hernandez, ...],[19,Samantha,Williams, .

Index	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
1	18	Jonathan	Moore	M	Florida	1971-03-07	2000-03-07	Administrat...	7000
2	20	Alexis	Allen	F	Florida	1977-08-07	2007-08-07	Administrat...	16000

文件目录



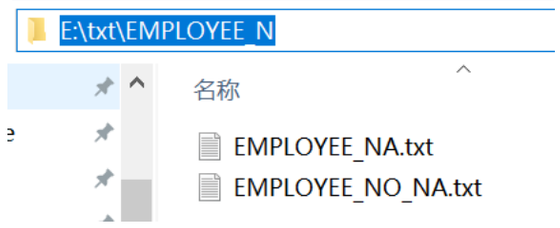
大文件拆分二 例：含有缺失值和不含缺失值的数据拆分到两个文件中

	A	B
1	=file("E:/txt/EMPLOYEE_nan.txt").cursor@t()	
2	=[true,false]	/确保每次分组都是两组
3	for A1,100	=A3.align@a(A2,~.array().pos(null)>0)
4		=B2.(file("E:/txt/EMPLOYEE_N/EMPLOYEE_"+["NA","NO_NA"](#)+".txt").export@at(~))
	/游标读取文件，循环取数，把每次取到的数据按照小文件的处理方式来处理，但导出时使用追加写入@a	

首次循环时的A2、B2

Index	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
1	1	Rebecca	Moore	F	California	1974-11-20	2005-03-11	R&D	7000.0
2	2	Ashley	Wilson	F	New York	1980-07-19	2008-03-16	Finance	11000.0
3	3	Rachel	Johnson	F	New Mexico	1970-12-17	2010-12-01	Sales	9000.0
4	4	Emily	Smith	F	Texas	1985-03-07	2006-08-15	HR	7000.0
5	5	Ashley	Smith	F	Texas	1975-05-13	2004-07-30	R&D	16000.0

文件目录



Index	Member
1	[[16,Christopher,, ...],[17,Hannah,Johnson, ...],[23,Joseph,, ...]
2	[[1,Rebecca,Moore, ...],[2,Ashley,Wilson, ...],[3,Rachel,Johnso...

Index	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
1	16	Christopher	(null)	M	Florida	1979-06-27	2007-06-27	Production	9000.0
2	17	Hannah	Johnson	F	Texas	(null)	2006-07-19	Marketing	4000.0
3	23	Joseph	(null)	M	California	1983-08-27	2003-08-27	Finance	6000.0
4	27	Alexis	Jones	F	California	1983-12-27	(null)	Marketing	10000.0

润乾
RAQSOFT

THANKS

创新技术 推动应用进步

www.raqsoft.com.cn