SDTL Merge Gallery

Version 1.0

The table below shows the locations of examples of the SDTL MergeDatasets command in various languages.

	Example source scripts:				
Sheet name	**SPSS**	**Stata**	**SAS**	**R**	**Python**
source tables -- dataframes used in examples					
Sequential	X	X			
Sequential with Replace			X		
Sequential with Update					X
Sequential with Duplicate Column name					X
1:1 merge	X	X	X		
1:1 newRow	X				
ful_join 1				X	X
ful_join 2				X	X
full_join 3		X		X	X
left_join 1				X	X
left_join 2				X	X
right_join 1				X	X
right_join 2				X	X
inner join 1				X	X
inner join 2				X	X
semi_join				X	
semi_join 2				X	

SDTL Merge Gallery

Version 1.0

The table below shows the locations of examples of the SDTL MergeDatasets command in various languages.

	Example source scripts:				
Sheet name	**SPSS**	**Stata**	**SAS**	**R**	**Python**
unmatched				X	
m:1		X			
Update 1		X	X		
Update 2		X	X		
SAS Update 1			X		
SAS Update 2			X		
SAS Update Reverse 1			X		
SAS Update Reverse 2			X		
SAS Modify Reverse 1			X		
SAS Modify Reverse 2			X		
Replace 1			X		
Replace 2					X
m:1 no newRow	X				
MergeFlagVariable	X				
MergeFlagVariable 2					X
MergeFlagVariable 3		X			
MergeFlagVariable 4		X			
FirstVariable	X				

SDTL Merge Gallery

Version 1.0

The table below shows the locations of examples of the SDTL MergeDatasets command in various languages.

	Example source scripts:				
Sheet name	**SPSS**	**Stata**	**SAS**	**R**	**Python**
Different Keys					X
Different Keys 2					X
Index Merge					X

DSmerge1

caseID	firstName	income	famID
1	Abe	15000	11
2	Barry		11
3	Chuck	17000	12
4	Dave	25000	12
5	Ed	21000	14

DSmerge2

rowID	lastName	income	famID2
1	Lincoln	31000	11
2	Madison	32000	12
3	Nelson	33000	13
6	Quincy	34000	14

DSmerge2a

caseID	lastName	income	famID2
1	Lincoln	31000	11
2	Madison	32000	12
3	Nelson	33000	13
6	Quincy	34000	14

DSmerge2b

rowID	lastName	income	famID
1	Lincoln	31000	11
2	Madison	32000	12
3	Nelson	33000	13
6	Quincy	34000	14

DSmerge3

caseID	lastName	income	famID
1	Lincoln	_	.
2	Madison	32000	.a
3	Nelson	.	13
4	Polk	5000	15
6	Quincy	34000	.
7	Roosevelt	.a	16

in DSmerge3, _ and .a are missing values that have been defined in advance by the user (this table was created in SAS).

caseID	firstName	income	famID	rowID	lastName	famID2
1	Abe	15000	11	1	Lincoln	11
2	Barry		11	2	Madison	12
3	Chuck	17000	12	3	Nelson	13
4	Dave	25000	12	6	Quincy	14
5	Ed	21000	14			

SPSS	MATCH FILES /FILE=='DSmerge1.sav' /FILE='DSmerge2.sav'					
Stata	use "DSmerge1.dta" merge 1:1 _n using "DSmerge2.dta"					
SDTL						
MergeDatasets						
	MergeByVariables					
	FirstVariable					
	LastVariable					

	MergeFileDescription		MergeFileDescription			
	FileName	**DSmerge1.sav**	FileName	**DSmerge2.sav**		
	MergeType	**Sequential**	MergeType	**Sequential**		
	MergeFlagVariable		MergeFlagVariable			
	RenameVariables		RenameVariables			
	update	**Master**	update	**FillNew**		
	NewRow	**TRUE**	NewRow	**TRUE**		
	KeepVariables		KeepVariables			
	DropVariables		DropVariables			
	MergeByNames		MergeByNames			

caseID	firstName	income	famID	rowID	lastName	famID2
1	Abe	31000	11	1	Lincoln	11
2	Barry	32000	11	2	Madison	12
3	Chuck	33000	12	3	Nelson	13
4	Dave	34000	12	6	Quincy	14
5	Ed	21000	14			

SAS	merge DSmerge1 DSmerge2;

SDTL				
MergeDatasets				
	MergeByVariables			
	FirstVariable			
	LastVariable			
	MergeFileDescription		MergeFileDescription	
	FileName	**DSmerge1.sav**	FileName	**DSmerge2.sav**
	MergeType	**Sequential**	MergeType	**Sequential**
	MergeFlagVariable		MergeFlagVariable	
	RenameVariables		RenameVariables	
	update	**Master**	update	**Replace**
	NewRow	**TRUE**	NewRow	**TRUE**
	KeepVariables		KeepVariables	
	DropVariables		DropVariables	
	MergeByNames		MergeByNames	

caseID	firstName	income	famID	rowID	lastName	famID2
1	Abe	15000	11	1	Lincoln	11
2	Barry	32000	11	2	Madison	12
3	Chuck	17000	12	3	Nelson	13
4	Dave	25000	12	6	Quincy	14
5	Ed	21000	14	NaN	NaN	NaN

Python DSmerge1.combine_first(DSmerge2)

SDTL

MergeDatasets

MergeByVariables

FirstVariable

LastVariable

MergeFileDescription		MergeFileDescription	
FileName	**DSmerge1.sav**	FileName	**DSmerge2.sav**
MergeType	**Sequential**	MergeType	**Sequential**
MergeFlagVariable		MergeFlagVariable	
RenameVariables		RenameVariables	
update	**Master**	update	**UpdateMissing**
NewRow	**TRUE**	NewRow	**TRUE**
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	

Does not replace or update overlapping columns; instead both are included with no rename												
caseID	firstName	income	famID	rowID	lastName	income	famID2					
1	Abe	15000	11	1	Lincoln	31000	11					
2	Barry	NaN	11	2	Madison	32000	12					
3	Chuck	17000	12	3	Nelson	33000	13					
4	Dave	25000	12	6	Quincy	34000	14					
5	Ed	21000	14	NaN	NaN	NaN	NaN					
Python	pd.concat([DSmerge1, DSmerge2], axis=1)											
OUT OF SCOPE for SDTL												
This Python command creates a dataframe with a duplicate column name. This can be prevented by using the verify_integrity paramter. Since the output dataframe would be unusable, it is not in scope for SDTL.												

caseID	firstName	income	famID	lastName	famID2
1	Abe	15000	11	Lincoln	11
2	Barry		11	Madison	12
3	Chuck	17000	12	Nelson	13
4	Dave	25000	12		
5	Ed	21000	14		
6		34000		Quincy	14

SPSS

```
MATCH FILES /FILE=='DSmerge1.sav'
/FILE='DSmerge2.sav'
/RENAME rowID=caseID
/BY caseID
.
```

Stata

```
merge 1:1 caseid using "DSmerge2a.dta" , nogenerate
```

"nogenerate" suppresses the creation of a _merge variable. See MergeFlagVariable 3 and 4

SAS

```
update DSmerge2a DSmerge1 UPDATEMODE=nomissingcheck;
by caseID;
```

the contents are the same, but this results in the following column order:
caseID, lastName, income, famID2, firstName, famID

SDTL

MergeDatasets

MergeByVariable	caseID			
FirstVariable				
LastVariable				
MergeFileDescription		MergeFileDescription		
FileName	DSmerge1.sav	FileName	DSmerge2.sav	
MergeType	**OneToOne**	MergeType	**OneToOne**	
MergeFlagVariab		MergeFlagVariab		
RenameVariables		RenameVariables	oldVariable: "rowID" newVariable: "caseID"	
update	**Master**	update	**FillNew**	
NewRow	TRUE	NewRow	TRUE	
KeepVariables		KeepVariables		
DropVariables		DropVariables		

caseID	firstName	income	famID2	rowID	lastName
1	Abe	15000	11	1	Lincoln
2	Barry		11		
3	Chuck	17000	12	2	Madison
4	Dave	25000	12		
		33000	13	3	Nelson
5	Ed	21000	14	6	Quincy
SPSS	MATCH FILES /FILE=='DSmerge1.sav' /rename famID=famID2 /FILE='DSmerge2.sav' /by famID2				
SDTL					
MergeDatasets					
	MergeByVariable	famID2			
	FirstVariable				
	LastVariable				
	MergeFileDescription		MergeFileDescription		
	FileName	DSmerge1.sav	FileName	DSmerge2.sav	
	MergeType	OneToOne	MergeType	OneToOne	
	MergeFlagVariable		MergeFlagVariable		
	RenameVariables	oldVariable: "fam newVariable: "far	RenameVariables		
	update	Master	update	FillNew	
	NewRow	TRUE	NewRow	TRUE	
	KeepVariables		KeepVariables		
	DropVariables		DropVariables		

Full join, no By variable									
caseID	firstName	income	famID	rowID	lastName	famID2			
1	Abe	15000	11	NA	NA	NA			
2	Barry	NA	11	NA	NA	NA			
3	Chuck	17000	12	NA	NA	NA			
4	Dave	25000	12	NA	NA	NA			
5	Ed	21000	14	NA	NA	NA			
NA	NA	31000	NA	1	Lincoln	11			
NA	NA	32000	NA	2	Madison	12			
NA	NA	33000	NA	3	Nelson	13			
NA	NA	34000	NA	6	Quincy	14			
Python	DSmerge1.merge(DSmerge2, how="outer")								
R	ds4 <- full_join(DataMerge1, DataMerge2)								
SDTL									
MergeDatasets									
	MergeByVariables								
	FirstVariable								
	LastVariable								

	MergeFileDescription		MergeFileDescription	
	FileName	DSmerge1.sav	FileName	DSmerge2.sav
	MergeType	Cartesian	MergeType	Cartesian
	MergeFlagVariable		MergeFlagVariable	
	RenameVariables		RenameVariables	
	update	Master	update	FillNew
	NewRow	TRUE	NewRow	TRUE
	KeepVariables		KeepVariables	
	DropVariables		DropVariables	

Full join, no repeats in By variable

caseID	firstName	income.x	famID	lastName	income.y	famID2
1	Abe	15000	11	Lincoln	31000	11
2	Barry	NA	11	Madison	32000	12
3	Chuck	17000	12	Nelson	33000	13
4	Dave	25000	12	NA	NA	NA
5	Ed	21000	14	NA	NA	NA
6	NA	NA	NA	Quincy	34000	14

Python: DSmerge1.merge(DSmerge2a, on="caseID", how="outer", suffixes=(".x", ".y"))

R: ds3 <- full_join(DataMerge1, DataMerge2a, by=""caseID"")

SDTL

MergeDatasets

	MergeByVariable	caseID		
	FirstVariable			
	LastVariable			
	MergeFileDescription		MergeFileDescription	
	FileName	DSmerge1.sav	FileName	DSmerge2.sav
	MergeType	Cartesian	MergeType	Cartesian
	MergeFlagVariable		MergeFlagVariable	
	RenameVariables	oldVariable: "inco newVariable: "inc	RenameVariables	oldVariable: "income" newVariable: "income.y"
	update	Master	update	FillNew
	NewRow	TRUE	NewRow	TRUE
	KeepVariables		KeepVariables	
	DropVariables		DropVariables	

Full join, with repeats in By variable

caseID	firstName	income.x	famID	rowID	lastName	income.y
1	Abe	15000	11	1	Lincoln	31000
2	Barry	NA	11	1	Lincoln	31000
3	Chuck	17000	12	2	Madison	32000
4	Dave	25000	12	2	Madison	32000
5	Ed	21000	14	6	Quincy	34000
NA	NA	NA	13	3	Nelson	33000

Python	DSmerge1.merge(DSmerge2b, on="famID", how="outer", suffixes=(".x", ".y"))	sort=True causes the rows to be ordered by famID
R	ds13 <- full_join(DataMerge1, DataMerge2b, by="famID")	
Stata	use "DSmerge1.dta" , clear merge m:m famid using "DSmerge2c.dta"	

SDTL				
MergeDatasets				
	MergeByVariable	caseID		
	FirstVariable			
	LastVariable			
	MergeFileDescription		MergeFileDescription	
	FileName	DSmerge1.sav	FileName	DSmerge2.sav
	MergeType	Cartesian	MergeType	Cartesian
	MergeFlagVariable		MergeFlagVariable	
	RenameVariables	oldVariable: "income" newVariable: "income.x"	RenameVariables	oldVariable: "income" newVariable: "income.y"
	update	Master	update	FillNew
	NewRow	TRUE	NewRow	TRUE
	KeepVariables		KeepVariables	
	DropVariables		DropVariables	

Left join, no repeates in By variable

	caseID	firstName	income.x	famID	lastName	income.y	famID2
1	1	Abe	15000	11	Lincoln	31000	11
2	2	Barry	NA	11	Madison	32000	12
3	3	Chuck	17000	12	Nelson	33000	13
4	4	Dave	25000	12	NA	NA	NA
5	5	Ed	21000	14	NA	NA	NA

Python DSmerge1.merge(DSmerge2a, on="caseID", how="left", suffixes=(".x", ".y"))

R ds5 <- left_join(DataMerge1, DataMerge2a, by="caseID")

SDTL

MergeDatasets

	MergeByVariable	caseID		
	FirstVariable			
	LastVariable			
	MergeFileDescription		MergeFileDescription	
	FileName	DSmerge1.sav	FileName	DSmerge2.sav
	MergeType	Cartesian	MergeType	Cartesian
	MergeFlagVariable		MergeFlagVariable	
	RenameVariables	oldVariable: "incc newVariable: "inc	RenameVariables	oldVariable: "income" newVariable: "income.y"
	update	Master	update	FillNew
	NewRow	TRUE	NewRow	FALSE
	KeepVariables		KeepVariables	
	DropVariables		DropVariables	

Left join, with repeates in By variable						
caseID	firstName	income.x	famID	rowID	lastName	income.y
1	Abe	15000	11	1	Lincoln	31000
2	Barry	NA	11	1	Lincoln	31000
3	Chuck	17000	12	2	Madison	32000
4	Dave	25000	12	2	Madison	32000
5	Ed	21000	14	6	Quincy	34000
Python	DSmerge1.merge(DSmerge2b, on="famID", how="left", suffixes=(".x", ".y"))					
R	ds15 <- left_join(DataMerge1, DataMerge2b, by="famID")					
SDTL						
MergeDatasets						
	MergeByVariable	famID				
	FirstVariable					
	LastVariable					
	MergeFileDescription		MergeFileDescription			
	FileName	DSmerge1.sav	FileName	DSmerge2.sav		
	MergeType	Cartesian	MergeType	Cartesian		
	MergeFlagVariable		MergeFlagVariable			
	RenameVariables	oldVariable: "inco newVariable: "inc	RenameVariables	oldVariable: "income" newVariable: "income.y"		
	update	Master	update	FillNew		
	NewRow	TRUE	NewRow	FALSE		
	KeepVariables		KeepVariables			
	DropVariables		DropVariables			

Right join no repeats in By variable

	caseID	firstName	income.x	famID	lastName	income.y	famID2
	1	Abe	15000	11	Lincoln	31000	11
	2	Barry	NA	11	Madison	32000	12
	3	Chuck	17000	12	Nelson	33000	13
	6	NA	NA	NA	Quincy	34000	14

Python: DSmerge1.merge(DSmerge2a, on="caseID", how="right", suffixes=(".x", ".y"))

R: ds6 <- right_join(DataMerge1, DataMerge2a, by="caseID")

SDTL

MergeDatasets

	MergeByVariable	caseID		
	FirstVariable			
	LastVariable			
	MergeFileDescription		MergeFileDescription	
	FileName	DSmerge1.sav	FileName	DSmerge2.sav
	MergeType	Cartesian	MergeType	Cartesian
	MergeFlagVariable		MergeFlagVariable	
	RenameVariables	oldVariable: "incc newVariable: "inc	RenameVariables	oldVariable: "income" newVariable: "income.y"
	update	Merge	update	FillNew
	NewRow	FALSE	NewRow	TRUE
	KeepVariables		KeepVariables	
	DropVariables		DropVariables	

Right join repeats in By variable

caseID	firstName	income.x	famID	rowID	lastName	income.y
1	Abe	15000	11	1	Lincoln	31000
2	Barry	NA	11	1	Lincoln	31000
3	Chuck	17000	12	2	Madison	32000
4	Dave	25000	12	2	Madison	32000
NA	NA	NA	13	3	Nelson	33000
5	Ed	21000	14	6	Quincy	34000

Python: DSmerge1.merge(DSmerge2b, on="famID", how="right", sort=True, suffixes=(".x", ".y"))

if sort=False, the row with missing caseID would come last

R: ds16 <- right_join(DataMerge1, DataMerge2b, by="famID")

SDTL

MergeDatasets

MergeByVariable	caseID			
FirstVariable				
LastVariable				
MergeFileDescription		MergeFileDescription		
FileName	DSmerge1.sav	FileName	DSmerge2.sav	
MergeType	Cartesian	MergeType	Cartesian	
MergeFlagVariable		MergeFlagVariable		
RenameVariables	oldVariable: "inco newVariable: "inc	RenameVariables	oldVariable: "income" newVariable: "income.y"	
update	FillNew	update	Master	
NewRow	FALSE	NewRow	TRUE	
KeepVariables		KeepVariables		
DropVariables		DropVariables		

Inner join no repeats in By variable

caseID	firstName	income.x	famID	lastName	income.y	famID2
1	Abe	15000	11	Lincoln	31000	11
2	Barry	NA	11	Madison	32000	12
3	Chuck	17000	12	Nelson	33000	13

Python: DSmerge1.merge(DSmerge2a, on="caseID", suffixes=(".x", ".y"))

R: ds7 <- inner_join(DataMerge1, DataMerge2a, by="caseID")

SDTL

MergeDatasets

	MergeByVariable	caseID		
	FirstVariable			
	LastVariable			
	MergeFileDescription		MergeFileDescription	
	FileName	DSmerge1.sav	FileName	DSmerge2.sav
	MergeType	Cartesian	MergeType	Cartesian
	MergeFlagVariable		MergeFlagVariable	
	RenameVariables	oldVariable: "inco newVariable: "inc	RenameVariables	oldVariable: "income" newVariable: "income.y"
	update	Master	update	FillNew
	NewRow	FALSE	NewRow	FALSE
	KeepVariables		KeepVariables	
	DropVariables		DropVariables	

Inner join repeats in By variable						
caseID	firstName	income.x	famID	rowID	lastName	income.y
1	Abe	15000	11	1	Lincoln	31000
2	Barry	NA	11	1	Lincoln	31000
3	Chuck	17000	12	2	Madison	32000
4	Dave	25000	12	2	Madison	32000
5	Ed	21000	14	6	Quincy	34000
Python	DSmerge1.merge(DSmerge2b, on="famID", suffixes=(".x", ".y"))					
R	ds17 <- inner_join(DataMerge1, DataMerge2b, by="famID")					
SDTL						
MergeDatasets						
	MergeByVariable	caseID				
	FirstVariable					
	LastVariable					
	MergeFileDescription		MergeFileDescription			
	FileName	DSmerge1.sav	FileName	DSmerge2.sav		
	MergeType	Cartesian	MergeType	Cartesian		
	MergeFlagVariable		MergeFlagVariable			
	RenameVariables	oldVariable: "inco newVariable: "inc	RenameVariables	oldVariable: "income" newVariable: "income.y"		
	update	Master	update	FillNew		
	NewRow	FALSE	NewRow	FALSE		
	KeepVariables		KeepVariables			
	DropVariables		DropVariables			

semi join, no repeats in By variable				
caseID	firstName	income	famID	
1	Abe	15000	11	
2	Barry	NA	11	
3	Chuck	17000	12	
R	ds8 <- semi_join(DataMerge1, DataMerge2a, by="caseID")			
SDTL				
MergeDatasets				
	MergeByVariable	caseID		
	FirstVariable			
	LastVariable			
	MergeFileDescription		MergeFileDescription	
	FileName	DSmerge1.sav	FileName	DSmerge2.sav
	MergeType	Cartesian	MergeType	Cartesian
	MergeFlagVariable		MergeFlagVariable	
	RenameVariables		RenameVariables	
	update	Master	update	Ignore
	NewRow	FALSE	NewRow	FALSE
	KeepVariables		KeepVariables	
	DropVariables		DropVariables	

semi join, with repeats in By variable				
caseID	firstName	income	famID	
1	Abe	15000	11	
2	Barry	NA	11	
3	Chuck	17000	12	
4	Dave	25000	12	
5	Ed	21000	14	
R	ds18 <- semi_join(DataMerge1, DataMerge2b, by="famID")			
SDTL				
MergeDatasets				
	MergeByVariables	caseID		
	FirstVariable			
	LastVariable			
	MergeFileDescription		MergeFileDescription	
	FileName	DSmerge1.sav	FileName	DSmerge2.sav
	MergeType	Cartesian	MergeType	Cartesian
	MergeFlagVariable		MergeFlagVariable	
	RenameVariables		RenameVariables	
	update	Master	update	FillNew
	NewRow	FALSE	NewRow	FALSE
	KeepVariables		KeepVariables	
	DropVariables		DropVariables	

caseID	firstName	income	famID	
4	Dave	25000	12	
5	Ed	21000	14	
R	ds9 <- anti_join(DataMerge1, DataMerge2a, by="caseID")			
SDTL				
MergeDatasets				
	MergeByVariable	caseID		
	FirstVariable			
	LastVariable			

	MergeFileDescription		MergeFileDescription	
	FileName	DSmerge1.sav	FileName	DSmerge2.sav
	MergeType	Unmatched	MergeType	Unmatched
	MergeFlagVariable		MergeFlagVariable	
	RenameVariables		RenameVariables	
	update	Master	update	Ignore
	NewRow	TRUE	NewRow	FALSE
	KeepVariables		KeepVariables	
	DropVariables		DropVariables	

caseid	firstn~e	income	famid2	lastname	_merge
1	Abe	15000	11	Lincoln	matched
2	Barry	.	11	Lincoln	matched
3	Chuck	17000	12	Madison	matched
4	Dave	25000	12	Madison	matched
5	Ed	21000	14	Quincy	matched
3		33000	13	Nelson	using only
Stata	merge m:1 famid using "DSmerge2a.dta"				
SDTL					
MergeDatasets					
	MergeByVariable	famID			
	FirstVariable				
	LastVariable				
	MergeFileDescription		MergeFileDescription		
	FileName	DSmerge1.sav	FileName	DSmerge2.sav	
	MergeType	ManyToOne	MergeType	OneToMany	
	MergeFlagVariable		MergeFlagVariable		
	RenameVariables		RenameVariables	oldVariable: "familyID2" newVariable: "familyID"	
	update	Master	update	FillNew	
	NewRow	TRUE	NewRow	TRUE	
	KeepVariables		KeepVariables		
	DropVariables		DropVariables		

caseid	firstname	income	famid	lastname	famid2
1	Abe	15000	11	Lincoln	11
2	Barry	32000	11	Madison	12
3	Chuck	17000	12	Nelson	13
4	Dave	25000	12		.
5	Ed	21000	14		.
6		34000	.	Quincy	14

Stata: merge 1:1 caseid using "DSmerge2a.dta" , update

SAS:
```
update DSmerge2a DSmerge1 UPDATEMODE=missingcheck;
by caseID;
```

contents are the same, but this results in the following column order: caseID, lastName, income, famID2, firstName, famID

SDTL

MergeDatasets

MergeByVariable	caseID			
FirstVariable				
LastVariable				
MergeFileDescription		MergeFileDescription		
FileName	DSmerge1.sav	FileName	DSmerge2.sav	
MergeType	OneToOne	MergeType	OneToOne	
MergeFlagVariable		MergeFlagVariable		
RenameVariables		RenameVariables		
update	UpdateNulls	update	UpdateMissing	
NewRow	TRUE	NewRow	TRUE	
KeepVariables		KeepVariables		
DropVariables		DropVariables		

caseid	firstname	income	famid	lastname	famid2
1	Abe	31000	11	Lincoln	11
2	Barry	32000	11	Madison	12
3	Chuck	33000	12	Nelson	13
4	Dave	25000	12		.
5	Ed	21000	14		.
6		34000	.	Quincy	14
Stata	merge 1:1 caseid using "DSmerge2a.dta" , update replace				
SDTL					
MergeDatasets					
	MergeByVariable	caseID			
	FirstVariable				
	LastVariable				
	MergeFileDescription		MergeFileDescription		
	FileName	DSmerge1.sav	FileName	DSmerge2.sav	
	MergeType	OneToOne	MergeType	OneToOne	
	MergeFlagVariable		MergeFlagVariable		
	RenameVariables		RenameVariables		
	update	Master	update	Replace	
	NewRow	TRUE	NewRow	TRUE	
	KeepVariables		KeepVariables		
	DropVariables		DropVariables		

Ignore columns not in the master dataframe, and update values that are missing in the master dataframe for Updateping columns							
caseID	firstName	income	famID				
1	Abe	15000	11				
2	Barry	32000	11				
3	Chuck	17000	12				
4	Dave	25000	12				
5	Ed	21000	14				
Python	DSmerge1.update(DSmerge2, overwrite=False)			this acts in-place on DSmerge1			
SDTL							
MergeDatasets							
	MergeByVariables						
	FirstVariable						
	LastVariable						
	MergeFileDescription		MergeFileDescription				
	FileName	DSmerge1.sav	FileName	DSmerge2.sav			
	MergeType	Sequential	MergeType	Sequential			
	MergeFlagVariable		MergeFlagVariable				
	RenameVariables		RenameVariables				
	update	Master	update	UpdateMissing			
	NewRow	TRUE	NewRow	FALSE			
	KeepVariables		KeepVariables				
	DropVariables		DropVariables	lastName, famID2, rowID			

With nomissingcheck option, any value in the SASUpdate dataframe that differs from Master will replace it, regardless of missing / user-missing / non-missing status

caseID	firstName	income	famID	lastName
1	Abe	_	.	Lincoln
2	Barry	32000	A	Madison
3	Chuck	.	13	Nelson
4	Dave	5000	15	Polk
5	Ed	21000	14	
6		34000	.	Quincy
7		A	16	Roosevelt

SAS

```
data DSmerge4;
   update DSmerge1 DSmerge3 UPDATEMODE=nomissingcheck;
   by caseID;
```

SDTL

MergeDatasets					
MergeByVariable	**VariableSymbolExpression**				
	variableName	caseID			
MergeDatasets	**MergeFileDescription**		**MergeFileDescription**		
	fileName	DSmerge1	fileName	DSmerge3	
	mergeType	OneToOne	mergeType	OneToOne	
	update	FillNew	update	Master	
	newRow	TRUE	newRow	TRUE	
consumesDatafra	DSmerge1	Dsmerge3			
producesDatafra	DSmerge4				

With missingcheck option, non-missing values in the SASUpdate dataframe will replace values in master, as will user-defined missing, but system missing will not

caseID	firstName	income	famID	lastName
1	Abe	.	11	Lincoln
2	Barry	32000	A	Madison
3	Chuck	17000	13	Nelson
4	Dave	5000	15	Polk
5	Ed	21000	14	
6		34000	.	Quincy
7		A	16	Roosevelt

SAS

```
data DSmerge4;
   update DSmerge1 DSmerge3 UPDATEMODE=missingcheck;
   by caseID;
```

SDTL

MergeDatasets					
mergeByVariable	**VariableSymbolExpression**				
	variableName	caseID			
MergeDatasets	**MergeFileDescription**		**MergeFileDescription**		
	fileName	DSmerge1	fileName	DSmerge3	
	mergeType	OneToOne	mergeType	OneToOne	
	update	UpdateMissing	Update	Master	
	newRow	TRUE	newRow	TRUE	
consumesDatafra	DSmerge1	Dsmerge3			
producesDatafra	DSmerge4				

caseID	lastname	income	famID2	firstname	famID
1	Lincoln	15000	11	Abe	11
2	Madison	.	12	Barry	11
3	Nelson	17000	13	Chuck	12
4		25000	.	Dave	12
5		21000	.	Ed	14
6	Quincy	34000	14		.

SAS

```
update DSmerge2a DSmerge1 UPDATEMODE=nomissingcheck;
by caseID;
```

SDTL

MergeDatasets					
mergeByVariables	**VariableSymbolExpression**				
	variableName	caseID			
MergeDatasets	**MergeFileDescription**		**MergeFileDescription**		
	fileName	DSmerge2a	fileName	DSmerge1	
	mergeType	OneToOne	mergeType	OneToOne	
	update	FillNew	update	Master	
	newRow	TRUE	newRow	TRUE	

caseID	lastname	income	famID2	firstname	famID
1	Lincoln	15000	11	Abe	11
2	Madison	32000	12	Barry	11
3	Nelson	17000	13	Chuck	12
4		25000	.	Dave	12
5		21000	.	Ed	14
6	Quincy	34000	14		.

SAS

```
update DSmerge2a DSmerge1 UPDATEMODE=missingcheck;
by caseID;
```

SDTL

MergeDatasets					
mergeByVariable	**VariableSymbolExpression**				
	variableName	caseID			
MergeDatasets	**MergeFileDescription**		**MergeFileDescription**		
	fileName	DSmerge2a	fileName	DSmerge1	
	mergeType	Cartesian	mergeType	Cartesian	
	update	UpdateMissing	update	Master	
	newRow	TRUE	newRow	TRUE	

caseID	lastname	income	famID2
1	Lincoln	15000	11
2	Madison	32000	12
3	Nelson	17000	13
6	Quincy	34000	14

SAS

```
data DSmerge2a;
   modify DSmerge2a DSmerge1 UPDATEMODE=missingcheck;
   by caseID;
```

this acts on DSmerge2a in-place

SDTL

MergeDatasets								
mergeByVariable	**VariableSymbolExpression**							
	variableName	caseID						
MergeDatasets	**MergeFileDescription**		**MergeFileDescription**					
	fileName	DSmerge2a	fileName	DSmerge1				
	mergeType	Cartesian	mergeType	Cartesian				
	update	UpdateMissing	update	Master				
	newRow	TRUE	newRow	FALSE				
			dropVariables	**VariableSymbolExpression**		**VariableSymbolExpression**		
				variableName	firstName	variableName	famID	
consumesDatafra	DSmerge2a	DSmerge1						
producesDatafra	DSmerge2a							

caseID	lastname	income	famID2
1	Lincoln	15000	11
2	Madison	.	12
3	Nelson	17000	13
6	Quincy	34000	14

SAS

```
data DSmerge2a;
   modify DSmerge2a DSmerge1 UPDATEMODE=nomissingcheck;
   by caseID;
```

this acts on DSmerge2a in-place

SDTL

MergeDatasets								
mergeByVariables	**VariableSymbolExpression**							
	variableName	caseID						
MergeDatasets	**MergeFileDescription**		**MergeFileDescription**					
	fileName	DSmerge2a	fileName	DSmerge1				
	mergeType	OneToOne	mergeType	OneToOne				
	update	FillNew	update	Master				
	newRow	TRUE	newRow	FALSE				
			dropVariables	**VariableSymbolExpression**		**VariableSymbolExpression**		
				variableName	firstName	variableName	famID	
consumesDataframe	DSmerge2a	DSmerge1						
producesDataframe	DSmerge2a							

Ignore columns not in the master dataframe, and replace values in the master dataframe when columns overlap

caseID	firstName	income	famID			
1	Abe	31000	11			
2	Barry	32000	11			
3	Chuck	33000	12			
4	Dave	34000	12			
5	Ed	21000	14			
SAS	data DSmerge1; modify DSmerge1 DSmerge2a ; by caseID;				this acts on DSmerge1 in-place	
SDTL						
MergeDatasets						
	MergeByVariable	caseID				
	FirstVariable					
	LastVariable					
	MergeFileDescription		MergeFileDescription			
	FileName	DSmerge1.sav	FileName	DSmerge2a.sav		
	MergeType	OneToOne	MergeType	OneToOne		
	MergeFlagVariable		MergeFlagVariable			
	RenameVariables		RenameVariables			
	update	Master	update	Replace		
	NewRow	TRUE	NewRow	FALSE		
	KeepVariables		KeepVariables			
	DropVariables		DropVariables	lastName, famID2		

Ignore columns not in the master dataframe, and replace values in the master dataframe when columns overlap						
caseID	firstName	income	famID			
1	Abe	31000	11			
2	Barry	32000	11			
3	Chuck	33000	12			
4	Dave	34000	12			
5	Ed	21000	14			
Python	DSmerge1.update(DSmerge2)				this acts on DSmerge1 in-place	
SDTL						
MergeDatasets						
	MergeByVariables					
	FirstVariable					
	LastVariable					
	MergeFileDescription		MergeFileDescription			
	FileName	DSmerge1.sav	FileName	DSmerge2.sav		
	MergeType	Sequential	MergeType	Sequential		
	MergeFlagVariable		MergeFlagVariable			
	RenameVariables		RenameVariables			
	update	Master	update	Replace		
	NewRow	TRUE	NewRow	FALSE		
	KeepVariables		KeepVariables			
	DropVariables		DropVariables	lastName, famID2, rowID		

caseID	firstName	income	famID	rowID	lastName
1	Abe	15000	11	1	Lincoln
2	Barry		11	1	Lincoln
3	Chuck	17000	12	2	Madison
4	Dave	25000	12	2	Madison
5	Ed	21000	14	6	Quincy

SPSS

```
MATCH FILES /FILE=='DSmerge1.sav'
/TABLE='DSmerge2.sav'
/RENAME famID2=famID
/BY famID
.
```

SDTL

MergeDatasets

MergeByVariable	famID			
FirstVariable				
LastVariable				
MergeFileDescription		MergeFileDescription		
FileName	DSmerge1.sav	FileName	DSmerge2.sav	
MergeType	ManyToOne	MergeType	OneToMany	
MergeFlagVariable		MergeFlagVariable		
RenameVariables		RenameVariables	oldVariable: "familyID2" newVariable: "familyID"	
update	Master	update	Ignore	
NewRow	TRUE	NewRow	FALSE	
KeepVariables		KeepVariables		
DropVariables		DropVariables		

caseID	firstName	income	famID	lastName	famID2	inDS1	inDS2
1	Abe	15000	11	Lincoln	11	1	1
2	Barry		11	Madison	12	1	1
3	Chuck	17000	12	Nelson	13	1	1
4	Dave	25000	12			1	0
5	Ed	21000	14			1	0
6		34000		Quincy	14	0	1

SPSS

```
MATCH FILES /FILE=='DSmerge1.sav'
/IN inDS1
/FILE='DSmerge2.sav'
/RENAME rowID=caseID
/IN inDS2
/BY caseID
.
```

SDTL

MergeDatasets

MergeByVariable	caseID			
FirstVariable				
LastVariable				
MergeFileDescription		MergeFileDescription		
FileName	DSmerge1.sav	FileName	DSmerge2.sav	
MergeType	OneToOne	MergeType	OneToOne	
MergeFlagVariab	inDS1	MergeFlagVariab	inDS2	
RenameVariables		RenameVariables	oldVariable: "rowID" newVariable: "caseID"	
update	Master	update	FillNew	
NewRow	TRUE	NewRow	TRUE	
KeepVariables		KeepVariables		

	DropVariables		DropVariables				

Instead of creating a boolean column for each dataframe, Python combines the information into one new column with string values

caseID	firstName	income_x	famID	lastName	income_y	famID2	_merge
1	Abe	15000	11	Lincoln	31000	11	both
2	Barry	NaN	11	Madison	32000	12	both
3	Chuck	17000	12	Nelson	33000	13	both
4	Dave	25000	12	NaN	NaN	NaN	left_only
5	Ed	21000	14	NaN	NaN	NaN	left_only
6	NaN	NaN	NaN	Quincy	34000	14	right_only

Python: `DSmerge1.merge(DSmerge2.rename(columns={"rowID": "caseID"}), on="caseID", how="outer", indicator=True)`

MergeDatasets

MergeByVariable	caseID			
FirstVariable				
LastVariable				
MergeFileDescription		MergeFileDescription		
FileName	DSmerge1.sav	FileName	DSmerge2.sav	
MergeType	OneToOne	MergeType	OneToOne	
MergeFlagVariab	_merge	MergeFlagVariable		
RenameVariables		RenameVariable:	oldVariable: "rowID" newVariable: "caseID"	
update	Master	update	FillNew	
NewRow	TRUE	NewRow	TRUE	
KeepVariables		KeepVariables		
DropVariables		DropVariables		

caseID	firstName	income	famID	lastName	famID2	_merge
1	Abe	15000	11	Lincoln	11	3
2	Barry		11	Madison	12	3
3	Chuck	17000	12	Nelson	13	3
4	Dave	25000	12			1
5	Ed	21000	14			1
6		34000		Quincy	14	2

Stata creates a variable named _merge with the following codes

Use SDTL **SetValueLabels** to assign labels to _merge

value	label	
1	master	observation appeared in master file only
2	using	observation appeared in other file only
3	match	observation appeared in both, missing values updated
4	match_update	observation appeared in both, conflicting nonmissing values
5	match_conflict	

See also MergeFlagVariable 4

Stata	merge 1:1 caseid using "DSmerge2a.dta"

SDTL	
MergeDatasets	
	MergeByVariable
	FirstVariable
	LastVariable

MergeFileDescription	
FileName	DSmerge1.sav
MergeType	**OneToOne**
MergeFlagVariab	_merge
RenameVariables	
update	**Master**
NewRow	TRUE
KeepVariables	
DropVariables	

MergeFileDescription	
FileName	DSmerge2.sav
MergeType	**OneToOne**
MergeFlagVariab	
RenameVariables	oldVariable: "rowID" newVariable: "caseID"
update	**FillNew**
NewRow	TRUE
KeepVariables	
DropVariables	

caseID	firstName	income	famID	lastName	famID2	Merge_code
1	Abe	15000	11	Lincoln	11	3
2	Barry		11	Madison	12	3
3	Chuck	17000	12	Nelson	13	3
4	Dave	25000	12			1
5	Ed	21000	14			1
6		34000		Quincy	14	2

Stata: merge 1:1 caseid using "DSmerge2a.dta" , generate(Merge_code)

Stata generates a new variable named Merge_code with the following codes

Use SDTL SetValueLabels to assign labels to Merge_code

SDTL

value	label	
1	master	observation appeared in master file only
2	using	observation appeared in other file only
3	match	observation appeared in both files
4	match_update	observation appeared in both, missing values updated
5	match_conflict	observation appeared in both, conflicting nonmissing values

See also MergeFlagVariable 3

MergeDatasets

MergeFileDescription		MergeFileDescription	
MergeByVariable	caseID		
FirstVariable			
LastVariable			
FileName	DSmerge1.sav	FileName	DSmerge2.sav
MergeType	**OneToOne**	MergeType	**OneToOne**
MergeFlagVariab	Merge_code	MergeFlagVariab	
RenameVariables		RenameVariables	oldVariable: "rowID" newVariable: "caseID"
update	**Master**	update	**FillNew**
NewRow	TRUE	NewRow	TRUE
KeepVariables		KeepVariables	
DropVariables		DropVariables	

caseID	firstName	income	famID	rowID	lastName	firstINfam	lastINfam
1	Abe	15000	11	1	Lincoln	1	0
2	Barry		11	1	Lincoln	0	1
3	Chuck	17000	12	2	Madison	1	0
4	Dave	25000	12	2	Madison	0	1
5	Ed	21000	14	6	Quincy	1	1

SPSS

```
MATCH FILES /FILE=='DSmerge1.sav'
/TABLE='DSmerge2.sav'
/RENAME famID2=famID
/first firstINfam
/last lastINfam
/BY famID    .
```

SDTL

MergeDatasets

MergeByVariable	famID			
FirstVariable	firstINfam			
LastVariable	lastINfam			
MergeFileDescription		MergeFileDescription		
FileName	DSmerge1.sav	FileName	DSmerge2.sav	
MergeType	ManyToOne	MergeType	OneToMany	
MergeFlagVariab		MergeFlagVariab		
RenameVariables		RenameVariables	oldVariable: "familyID2" newVariable: "familyID"	
update	Master	update	Ignore	
NewRow	TRUE	NewRow	FALSE	
KeepVariables		KeepVariables		
DropVariables		DropVariables		

Inner join, no repeats in by variable

caseID	firstName	income_x	famID	rowID	lastName	income_y	famID2
1	Abe	15000	11	1	Lincoln	31000	11
2	Barry	NaN	11	2	Madison	32000	12
3	Chuck	17000	12	3	Nelson	33000	13

Python: `DSmerge1.merge(DSmerge2, left_on="caseID", right_on="rowID")`

SDTL

MergeDatasets

MergeByVariables	caseID		
FirstVariable			
LastVariable			
MergeFileDescription		MergeFileDescription	
FileName	DSmerge1.sav	FileName	DSmerge2.sav
MergeType	Cartesian	MergeType	Cartesian
MergeFlagVariable		MergeFlagVariable	
RenameVariables	oldVariable: "inco newVariable: "inc	RenameVariables	oldVariable: "income" newVariable: "income.y"
update	Master	update	FillNew
NewRow	FALSE	NewRow	FALSE
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	rowID

Inner join, repeats in by variable

caseID	firstName	income_x	famID	rowID	lastName	income_y	famID2
1	Abe	15000	11	1	Lincoln	31000	11
2	Barry	NaN	11	1	Lincoln	31000	11
3	Chuck	17000	12	2	Madison	32000	12
4	Dave	25000	12	2	Madison	32000	12
5	Ed	21000	14	6	Quincy	34000	14

Python: `DSmerge1.merge(DSmerge2, left_on="famID", right_on="famID2")`

SDTL

MergeDatasets

MergeByVariables	famID		
FirstVariable			
LastVariable			
MergeFileDescription		MergeFileDescription	
FileName	DSmerge1.sav	FileName	DSmerge2.sav
MergeType	Cartesian	MergeType	Cartesian
MergeFlagVariable		MergeFlagVariable	
RenameVariables	oldVariable: "inco newVariable: "inc	RenameVariables	oldVariable: "income" newVariable: "income.y"
update	Master	update	FillNew
NewRow	FALSE	NewRow	FALSE
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	famID2

Left join, no repeats in by variable

caseID	firstName	income_x	famID	rowID	lastName	income_y	famID2
1	Abe	15000	11	1	Lincoln	31000	11
2	Barry	NaN	11	2	Madison	32000	12
3	Chuck	17000	12	3	Nelson	33000	13
4	Dave	25000	12	NaN	NaN	NaN	NaN
5	Ed	21000	14	NaN	NaN	NaN	NaN

Python: `DSmerge1.merge(DSmerge2, left_on="caseID", right_on="rowID", how="left")`

SDTL

MergeDatasets

MergeByVariables	caseID		
FirstVariable			
LastVariable			
MergeFileDescription		MergeFileDescription	
FileName	DSmerge1.sav	FileName	DSmerge2.sav
MergeType	Cartesian	MergeType	Cartesian
MergeFlagVariable		MergeFlagVariable	
RenameVariables	oldVariable: "inco newVariable: "inc	RenameVariables	oldVariable: "income" newVariable: "income.y"
update	Master	update	FillNew
NewRow	TRUE	NewRow	FALSE
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	rowID

Left join, repeats in by variable

caseID	firstName	income_x	famID	rowID	lastName	income_y	famID2
1	Abe	15000	11	1	Lincoln	31000	11
2	Barry	NaN	11	1	Lincoln	31000	11
3	Chuck	17000	12	2	Madison	32000	12
4	Dave	25000	12	2	Madison	32000	12
5	Ed	21000	14	6	Quincy	34000	14

Python: `DSmerge1.merge(DSmerge2, left_on="famID", right_on="famID2", how="left")`

SDTL

MergeDatasets

MergeByVariables	famID		
FirstVariable			
LastVariable			
MergeFileDescription		MergeFileDescription	
FileName	DSmerge1.sav	FileName	DSmerge2.sav
MergeType	Cartesian	MergeType	Cartesian
MergeFlagVariab		MergeFlagVariab	
RenameVariables	oldVariable: "inco newVariable: "inc	RenameVariables	oldVariable: "income" newVariable: "income.y"
update	Master	update	FillNew
NewRow	TRUE	NewRow	FALSE
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	famID2

Right join, no repeats in by variable

caseID	firstName	income_x	famID	rowID	lastName	income_y	famID2
1	Abe	15000	11	1	Lincoln	31000	11
2	Barry	NaN	11	2	Madison	32000	12
3	Chuck	17000	12	3	Nelson	33000	13
NaN	NaN	NaN	NaN	6	Quincy	34000	14

Python: `DSmerge1.merge(DSmerge2, left_on="caseID", right_on="rowID", how="right")`

SDTL

MergeDatasets

MergeByVariables	caseID		
FirstVariable			
LastVariable			
MergeFileDescription		MergeFileDescription	
FileName	DSmerge1.sav	FileName	DSmerge2.sav
MergeType	Cartesian	MergeType	Cartesian
MergeFlagVariable		MergeFlagVariable	
RenameVariables	oldVariable: "inco newVariable: "inc	RenameVariables	oldVariable: "income" newVariable: "income.y"
update	Master	update	FillNew
NewRow	FALSE	NewRow	TRUE
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	rowID

Right join, repeats in by variable

caseID	firstName	income_x	famID	rowID	lastName	income_y	famID2
1	Abe	15000	11	1	Lincoln	31000	11
2	Barry	NaN	11	1	Lincoln	31000	11
3	Chuck	17000	12	2	Madison	32000	12
4	Dave	25000	12	2	Madison	32000	12
5	Ed	21000	14	6	Quincy	34000	14
NaN	NaN	NaN	NaN	3	Nelson	33000	13

Python: `DSmerge1.merge(DSmerge2, left_on="famID", right_on="famID2", how="right")`

SDTL

MergeDatasets

MergeByVariables	famID		
FirstVariable			
LastVariable			
MergeFileDescription		MergeFileDescription	
FileName	DSmerge1.sav	FileName	DSmerge2.sav
MergeType	Cartesian	MergeType	Cartesian
MergeFlagVariable		MergeFlagVariable	
RenameVariables	oldVariable: "inco newVariable: "inc	RenameVariables	oldVariable: "income" newVariable: "income.y"
update	Master	update	FillNew
NewRow	FALSE	NewRow	TRUE
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	famID2

Outer join, no repeats in by variable

caseID	firstName	income_x	famID	rowID	lastName	income_y	famID2
1	Abe	15000	11	1	Lincoln	31000	11
2	Barry	NaN	11	2	Madison	32000	12
3	Chuck	17000	12	3	Nelson	33000	13
4	Dave	25000	12	NaN	NaN	NaN	NaN
5	Ed	21000	14	NaN	NaN	NaN	NaN

Outer join, repeats in by variable

caseID	firstName	income_x	famID	rowID	lastName	income_y	famID2
1	Abe	15000	11	1	Lincoln	31000	11
2	Barry	NaN	11	1	Lincoln	31000	11
3	Chuck	17000	12	2	Madison	32000	12
4	Dave	25000	12	2	Madison	32000	12
5	Ed	21000	14	6	Quincy	34000	14

NaN	NaN	NaN	NaN	6	Quincy	34000	14

Python: DSmerge1.merge(DSmerge2, left_on="caseID", right_on="rowID", how="outer")

SDTL

MergeDatasets

MergeByVariables	caseID		
FirstVariable			
LastVariable			
MergeFileDescription		MergeFileDescription	
FileName	DSmerge1.sav	FileName	DSmerge2.sav
MergeType	Cartesian	MergeType	Cartesian
MergeFlagVariable		MergeFlagVariable	
RenameVariables	oldVariable: "inco newVariable: "inc	RenameVariables	oldVariable: "income" newVariable: "income.y"
update	Master	update	FillNew
NewRow	TRUE	NewRow	TRUE
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	rowID

NaN	NaN	NaN	NaN	3	Nelson	33000	13

Python: DSmerge1.merge(DSmerge2, left_on="famID", right_on="famID2", how="outer")

SDTL

MergeDatasets

MergeByVariables	famID		
FirstVariable			
LastVariable			
MergeFileDescription		MergeFileDescription	
FileName	DSmerge1.sav	FileName	DSmerge2.sav
MergeType	Cartesian	MergeType	Cartesian
MergeFlagVariable		MergeFlagVariable	
RenameVariables	oldVariable: "inco newVariable: "inc	RenameVariables	oldVariable: "income" newVariable: "income.y"
update	Master	update	FillNew
NewRow	TRUE	NewRow	TRUE
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	famID2

Inner Join

caseID	firstName	income_x	famID	rowID	lastName	income_y	famID2
1	Abe	15000	11	1	Lincoln	31000	11

Python: `DSmerge1.merge(DSmerge2, left_on=["caseID", "famID"], right_on=["rowID", "famID2"])`

SDTL

MergeDatasets

MergeByVariable	caseID, famID		
FirstVariable			
LastVariable			
MergeFileDescription		MergeFileDescription	
FileName	DSmerge1.sav	FileName	DSmerge2.sav
MergeType	OneToOne	MergeType	OneToOne
MergeFlagVariab		MergeFlagVariab	
RenameVariable	oldVariable: "inco newVariable: "inc	RenameVariable	oldVariable: "income" newVariable: "income.y"
update	Master	update	FillNew
NewRow	FALSE	NewRow	FALSE
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	rowID, famID2

Right Join

caseID	firstName	income_x	famID	rowID	lastName	income_y	famID2
1	Abe	15000	11	1	Lincoln	31000	11
NaN	NaN	NaN	NaN	2	Madison	32000	12
NaN	NaN	NaN	NaN	3	Nelson	33000	13
NaN	NaN	NaN	NaN	6	Quincy	34000	14

Python: `DSmerge1.merge(DSmerge2, left_on=["caseID", "famID"], right_on=["rowID", "famID2"], how="right")`

SDTL

MergeDatasets

MergeByVariable	caseID, famID		
FirstVariable			
LastVariable			
MergeFileDescription		MergeFileDescription	
FileName	DSmerge1.sav	FileName	DSmerge2.sav
MergeType	OneToOne	MergeType	OneToOne
MergeFlagVariab		MergeFlagVariab	
RenameVariable	oldVariable: "inco newVariable: "inc	RenameVariable	oldVariable: "income" newVariable: "income.y"
update	Master	update	FillNew
NewRow	FALSE	NewRow	TRUE
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	rowID, famID2

Left Join

caseID	firstName	income_x	famID	rowID	lastName	income_y	famID2
1	Abe	15000	11	1	Lincoln	31000	11
2	Barry	NaN	11	NaN	NaN	NaN	NaN
3	Chuck	17000	12	NaN	NaN	NaN	NaN
4	Dave	25000	12	NaN	NaN	NaN	NaN
5	Ed	21000	14	NaN	NaN	NaN	NaN

Python: `DSmerge1.merge(DSmerge2, left_on=["caseID", "famID"], right_on=["rowID", "famID2"], how="left")`

SDTL

MergeDatasets

MergeByVariable	caseID, famID		
FirstVariable			
LastVariable			
MergeFileDescription		MergeFileDescription	
FileName	DSmerge1.sav	FileName	DSmerge2.sav
MergeType	OneToOne	MergeType	OneToOne
MergeFlagVariable		MergeFlagVariable	
RenameVariable	oldVariable: "inco newVariable: "inc	RenameVariable	oldVariable: "income" newVariable: "income.y"
update	Master	update	FillNew
NewRow	TRUE	NewRow	FALSE
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	rowID, famID2

Outer Join

caseID	firstName	income_x	famID	rowID	lastName	income_y	famID2
1	Abe	15000	11	1	Lincoln	31000	11
2	Barry	NaN	11	NaN	NaN	NaN	NaN
3	Chuck	17000	12	NaN	NaN	NaN	NaN
4	Dave	25000	12	NaN	NaN	NaN	NaN
5	Ed	21000	14	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	2	Madison	32000	12
NaN	NaN	NaN	NaN	3	Nelson	33000	13
NaN	NaN	NaN	NaN	6	Quincy	34000	14

Python: `DSmerge1.merge(DSmerge2, left_on=["caseID", "famID"], right_on=["rowID", "famID2"], how="outer")`

SDTL

MergeDatasets

MergeByVariable	caseID, famID		
FirstVariable			
LastVariable			
MergeFileDescription		MergeFileDescription	
FileName	DSmerge1.sav	FileName	DSmerge2.sav
MergeType	OneToOne	MergeType	OneToOne
MergeFlagVariable		MergeFlagVariable	
RenameVariable	oldVariable: "inco newVariable: "inc	RenameVariable	oldVariable: "income" newVariable: "income.y"
update	Master	update	FillNew
NewRow	TRUE	NewRow	TRUE
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	rowID, famID2

Index Merge, Inner Join / Right Join

caseID	firstName	income_x	famID	rowID	lastName	income_y	famID2
1	Abe	15000	11	1	Lincoln	31000	11
2	Barry	NaN	11	2	Madison	32000	12
3	Chuck	17000	12	3	Nelson	33000	13
4	Dave	25000	13	6	Quincy	34000	14

Python

```
left.merge(right, left_index=True, right_index=True)
```

SDTL

MergeDatasets

MergeByVariables

FirstVariable

LastVariable

MergeFileDescription		MergeFileDescription	
FileName	**DSmerge1.sav**	FileName	**DSmerge2.sav**
MergeType	**Sequential**	MergeType	**Sequential**
MergeFlagVariab		MergeFlagVariab	
RenameVariables		RenameVariables	
update	**Master**	update	**FillNew**
NewRow	**FALSE**	NewRow	**FALSE**
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	

Python

```
left.merge(right, left_index=True, right_index=True, how="right")
```

SDTL

MergeDatasets

MergeByVariables

FirstVariable

LastVariable

MergeFileDescription		MergeFileDescription	
FileName	**DSmerge1.sav**	FileName	**DSmerge2.sav**
MergeType	**Sequential**	MergeType	**Sequential**
MergeFlagVariab		MergeFlagVariab	
RenameVariables		RenameVariables	
update	**Master**	update	**FillNew**
NewRow	**FALSE**	NewRow	**TRUE**
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	

Index Merge, Left Join / Outer Join

caseID	firstName	income_x	famID	rowID	lastName	income_y	famID2
1	Abe	15000	11	1	Lincoln	31000	11
2	Barry	NaN	11	2	Madison	32000	12
3	Chuck	17000	12	3	Nelson	33000	13
4	Dave	25000	13	6	Quincy	34000	14
5	Ed	21000	14	NaN	NaN	NaN	NaN

Python

```
left.merge(right, left_index=True, right_index=True, how="left")
```

SDTL

MergeDatasets

MergeByVariables

FirstVariable

LastVariable

MergeFileDescription		MergeFileDescription	
FileName	**DSmerge1.sav**	FileName	**DSmerge2.sav**
MergeType	**Sequential**	MergeType	**Sequential**
MergeFlagVariab		MergeFlagVariab	
RenameVariables		RenameVariables	
update	**Master**	update	**FillNew**
NewRow	**TRUE**	NewRow	**FALSE**
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	

Python

```
left.merge(right, left_index=True, right_index=True, how="outer")
```

SDTL

MergeDatasets

MergeByVariables

FirstVariable

LastVariable

MergeFileDescription		MergeFileDescription	
FileName	**DSmerge1.sav**	FileName	**DSmerge2.sav**
MergeType	**Sequential**	MergeType	**Sequential**
MergeFlagVariab		MergeFlagVariab	
RenameVariables		RenameVariables	
update	**Master**	update	**FillNew**
NewRow	**TRUE**	NewRow	**TRUE**
KeepVariables		KeepVariables	
DropVariables		DropVariables	
MergeByNames		MergeByNames	