

Delimited Text

Andrew Ba Tran

Contents

read_delim()	2
read_tsv()	3
Fixed width files	3

This is from the second chapter of learn.r-journalism.com.

Sometimes you'll encounter data with values that are delimited (separated) by characters other than commas. For example, I once received a spreadsheet delimited with emojis.

Why does this happen? This means the data was exported from a database and the user chose this option. Not all data can be the way we ideally want it but as far as this data structure goes, at least we're dealing with PDFs (which is possible but we won't be going over that process in this course)

If opened in a spreadsheet app, the delimited file would be interpreted like any other spreadsheet.

But this is how a pipe-delimited file looks like internally.

```
1 Office Name|Job Code|Job Title|Base Pay|Position ID|Employee Identifier|Original Hire Date
2 STATES ATTORNEY|1172|Assistant State's Attorney|$20,088.00|9510200|6ac7ba3e-d286-44f5-87a0-191dc415e23c|5/16/05
3 STATES ATTORNEY|1172|Assistant State's Attorney|$23,436.00|9510200|6ac7ba3e-d286-44f5-87a0-191dc415e23c|5/16/05
4 STATES ATTORNEY|1172|Assistant State's Attorney|$20,422.82|9510200|6ac7ba3e-d286-44f5-87a0-191dc415e23c|5/16/05
5 STATES ATTORNEY|1172|Assistant State's Attorney|$23,904.80|9510200|6ac7ba3e-d286-44f5-87a0-191dc415e23c|5/16/05
6 STATES ATTORNEY|1172|Assistant State's Attorney|$20,745.80|9510200|6ac7ba3e-d286-44f5-87a0-191dc415e23c|5/16/05
7 STATES ATTORNEY|1172|Assistant State's Attorney|$24,473.38|9510200|6ac7ba3e-d286-44f5-87a0-191dc415e23c|5/16/05
8 STATES ATTORNEY|1172|Assistant State's Attorney|$21,217.35|9510200|6ac7ba3e-d286-44f5-87a0-191dc415e23c|5/16/05
9 COUNTY ASSESSOR|5049|Residential Model Sr Anal III|$17,770.86|9500731|f313b1c3-1b1a-4b07-bb75-a8c850a91bac|9/28/98
10 COUNTY ASSESSOR|5049|Residential Model Sr Anal III|$20,800.67|9500731|f313b1c3-1b1a-4b07-bb75-a8c850a91bac|9/28/98
11 COUNTY ASSESSOR|5049|Residential Model Sr Anal III|$17,873.76|9500731|f313b1c3-1b1a-4b07-bb75-a8c850a91bac|9/28/98
12 COUNTY ASSESSOR|5049|Residential Model Sr Anal III|$20,904.80|9500731|f313b1c3-1b1a-4b07-bb75-a8c850a91bac|9/28/98
13 COUNTY ASSESSOR|5049|Residential Model Sr Anal III|$18,254.40|9500731|f313b1c3-1b1a-4b07-bb75-a8c850a91bac|9/28/98
14 COUNTY ASSESSOR|5049|Residential Model Sr Anal III|$21,375.20|9500731|f313b1c3-1b1a-4b07-bb75-a8c850a91bac|9/28/98
15 COUNTY ASSESSOR|5049|Residential Model Sr Anal III|$18,626.76|9500731|f313b1c3-1b1a-4b07-bb75-a8c850a91bac|9/28/98
16 COUNTY ASSESSOR|5049|Residential Model Sr Anal III|$21,802.46|9500731|f313b1c3-1b1a-4b07-bb75-a8c850a91bac|9/28/98
17 PROVIDENT HOSPITAL|1642|Attending Physician XII|$57,692.40|1100069|f888af25-5b0d-457a-83cc-3415f03ed7c6|12/16/13
18 PROVIDENT HOSPITAL|1642|Attending Physician XII|$67,307.80|1100069|f888af25-5b0d-457a-83cc-3415f03ed7c6|12/16/13
19 PROVIDENT HOSPITAL|1642|Attending Physician XII|$57,692.40|1100069|f888af25-5b0d-457a-83cc-3415f03ed7c6|12/16/13
20 PROVIDENT HOSPITAL|1642|Attending Physician XII|$67,307.80|1100069|f888af25-5b0d-457a-83cc-3415f03ed7c6|12/16/13
21 PROVIDENT HOSPITAL|1642|Attending Physician XII|$57,692.28|1100069|f888af25-5b0d-457a-83cc-3415f03ed7c6|12/16/13
22 PROVIDENT HOSPITAL|1642|Attending Physician XII|$67,307.66|1100069|f888af25-5b0d-457a-83cc-3415f03ed7c6|12/16/13
23 PROVIDENT HOSPITAL|1642|Attending Physician XII|$57,692.28|1100069|f888af25-5b0d-457a-83cc-3415f03ed7c6|12/16/13
24 PROVIDENT HOSPITAL|1642|Attending Physician XII|$67,824.96|1100069|f888af25-5b0d-457a-83cc-3415f03ed7c6|12/16/13
```

And this is how a tab-delimited file looks.

```
1 Office Name Job Code Job Title Base Pay Position ID Employee Identifier Original Hire Date
2 STATES ATTORNEY 1172 Assistant State's Attorney $20,088.00 9510200 6ac7ba3e-d286-44f5-87a0-191dc415e23c 5/16/05
3 STATES ATTORNEY 1172 Assistant State's Attorney $23,436.00 9510200 6ac7ba3e-d286-44f5-87a0-191dc415e23c 5/16/05
4 STATES ATTORNEY 1172 Assistant State's Attorney $20,422.82 9510200 6ac7ba3e-d286-44f5-87a0-191dc415e23c 5/16/05
5 STATES ATTORNEY 1172 Assistant State's Attorney $23,904.80 9510200 6ac7ba3e-d286-44f5-87a0-191dc415e23c 5/16/05
6 STATES ATTORNEY 1172 Assistant State's Attorney $20,745.80 9510200 6ac7ba3e-d286-44f5-87a0-191dc415e23c 5/16/05
7 STATES ATTORNEY 1172 Assistant State's Attorney $24,473.38 9510200 6ac7ba3e-d286-44f5-87a0-191dc415e23c 5/16/05
8 STATES ATTORNEY 1172 Assistant State's Attorney $21,217.35 9510200 6ac7ba3e-d286-44f5-87a0-191dc415e23c 5/16/05
9 COUNTY ASSESSOR 5049 Residential Model Sr Anal III $17,770.86 9500731 f313b1c3-1b1a-4b07-bb75-a8c850a91bac 9/28/98
10 COUNTY ASSESSOR 5049 Residential Model Sr Anal III $20,800.67 9500731 f313b1c3-1b1a-4b07-bb75-a8c850a91bac 9/28/98
11 COUNTY ASSESSOR 5049 Residential Model Sr Anal III $17,873.76 9500731 f313b1c3-1b1a-4b07-bb75-a8c850a91bac 9/28/98
12 COUNTY ASSESSOR 5049 Residential Model Sr Anal III $20,904.80 9500731 f313b1c3-1b1a-4b07-bb75-a8c850a91bac 9/28/98
13 COUNTY ASSESSOR 5049 Residential Model Sr Anal III $18,254.40 9500731 f313b1c3-1b1a-4b07-bb75-a8c850a91bac 9/28/98
14 COUNTY ASSESSOR 5049 Residential Model Sr Anal III $21,375.20 9500731 f313b1c3-1b1a-4b07-bb75-a8c850a91bac 9/28/98
15 COUNTY ASSESSOR 5049 Residential Model Sr Anal III $18,626.76 9500731 f313b1c3-1b1a-4b07-bb75-a8c850a91bac 9/28/98
16 COUNTY ASSESSOR 5049 Residential Model Sr Anal III $21,802.46 9500731 f313b1c3-1b1a-4b07-bb75-a8c850a91bac 9/28/98
17 PROVIDENT HOSPITAL 1642 Attending Physician XII $57,692.40 1100069 f888af25-5b0d-457a-83cc-3415f03ed7c6 12/16/13
18 PROVIDENT HOSPITAL 1642 Attending Physician XII $67,307.80 1100069 f888af25-5b0d-457a-83cc-3415f03ed7c6 12/16/13
19 PROVIDENT HOSPITAL 1642 Attending Physician XII $57,692.40 1100069 f888af25-5b0d-457a-83cc-3415f03ed7c6 12/16/13
20 PROVIDENT HOSPITAL 1642 Attending Physician XII $67,307.80 1100069 f888af25-5b0d-457a-83cc-3415f03ed7c6 12/16/13
21 PROVIDENT HOSPITAL 1642 Attending Physician XII $57,692.28 1100069 f888af25-5b0d-457a-83cc-3415f03ed7c6 12/16/13
22 PROVIDENT HOSPITAL 1642 Attending Physician XII $67,307.66 1100069 f888af25-5b0d-457a-83cc-3415f03ed7c6 12/16/13
23 PROVIDENT HOSPITAL 1642 Attending Physician XII $57,692.28 1100069 f888af25-5b0d-457a-83cc-3415f03ed7c6 12/16/13
24 PROVIDENT HOSPITAL 1642 Attending Physician XII $67,824.96 1100069 f888af25-5b0d-457a-83cc-3415f03ed7c6 12/16/13
```

In base R, the way to import these files is to use the `read.table()` function.

You pass it the location of the file (in this case, it's in the sub directory "data") and whether it has a header row or not and what separator symbol to look for

```
# read.table(file, header=logical_value, sep="delimiter")
df1 <- read.table("data/Employee_Payroll_Pipe.txt", header=TRUE, sep="|")
```

View(df1)

```
# a \t indicates a tab (and a \n indicates a line break, like pressing enter in a document)
df2 <- read.table("data/Employee_Payroll_Tab.txt", header=TRUE, sep="\t")
```

View(df2)

	Office.Name	Job.Code	Job.Title	Base.Pay	Position.ID	Em
1	STATES ATTORNEY	1172	Assistant States Attorney \$23,904.80 9510200 6ac...	\$20,745.80	9510200	6ac
2	STATES ATTORNEY	1172	Assistant States Attorney \$24,473.38 9510200 6ac...	\$21,217.35	9510200	6ac
3	COUNTY ASSESSOR	5049	Residential Model Sr Anal III	\$17,770.86	9500731	f31
4	COUNTY ASSESSOR	5049	Residential Model Sr Anal III	\$20,800.67	9500731	f31
5	COUNTY ASSESSOR	5049	Residential Model Sr Anal III	\$17,873.76	9500731	f31
6	COUNTY ASSESSOR	5049	Residential Model Sr Anal III	\$20,904.80	9500731	f31
7	COUNTY ASSESSOR	5049	Residential Model Sr Anal III	\$18,254.40	9500731	f31
8	COUNTY ASSESSOR	5049	Residential Model Sr Anal III	\$21,375.20	9500731	f31
9	COUNTY ASSESSOR	5049	Residential Model Sr Anal III	\$18,626.76	9500731	f31
10	COUNTY ASSESSOR	5049	Residential Model Sr Anal III	\$21,802.46	9500731	f31

read_delim()

The downsides of using the base `read.table()` function are the same as using base `read.csv()`

- Naming schemes aren't consistent
- Slow loading
- Turns strings into Factors automatically

To read in delimited pipe files use `read_delim()` from **readr**

```
## If you don't have readr installed yet, uncomment and run the line below
# install.packages("readr")
library(readr)
df1 <- read_delim("data/Employee_Payroll_Pipe.txt", delim="|")
```

```
## Parsed with column specification:
## cols(
##   `Office Name` = col_character(),
##   `Job Code` = col_integer(),
##   `Job Title` = col_character(),
##   `Base Pay` = col_character(),
##   `Position ID` = col_integer(),
##   `Employee Identifier` = col_character(),
##   `Original Hire Date` = col_character()
## )
df1
```

```
## # A tibble: 23 x 7
##   `Office Name` `Job Code` `Job Title` `Base Pay` `Position ID`
```

```
##      <chr>                <int> <chr>                <chr>                <int>
## 1 STATES ATTORN~          1172 Assistant State's ~ " $20,088.~      9510200
## 2 STATES ATTORN~          1172 Assistant State's ~ " $23,436.~      9510200
## 3 STATES ATTORN~          1172 Assistant State's ~ " $20,422.~      9510200
## 4 STATES ATTORN~          1172 Assistant State's ~ " $23,904.~      9510200
## 5 STATES ATTORN~          1172 Assistant State's ~ " $20,745.~      9510200
## 6 STATES ATTORN~          1172 Assistant State's ~ " $24,473.~      9510200
## 7 STATES ATTORN~          1172 Assistant State's ~ " $21,217.~      9510200
## 8 COUNTY ASSESS~          5049 Residential Model ~ " $17,770.~      9500731
## 9 COUNTY ASSESS~          5049 Residential Model ~ " $20,800.~      9500731
## 10 COUNTY ASSESS~         5049 Residential Model ~ " $17,873.~      9500731
## # ... with 13 more rows, and 2 more variables: `Employee
## #   Identifier` <chr>, `Original Hire Date` <chr>
```

read_tsv()

To read in tab delimited pipe files use `read_tsv()` from **readr**

```
df2 <- read_tsv("data/Employee_Payroll_Tab.txt")
```

```
## Parsed with column specification:
## cols(
##   `Office Name` = col_character(),
##   `Job Code` = col_integer(),
##   `Job Title` = col_character(),
##   `Base Pay` = col_character(),
##   `Position ID` = col_integer(),
##   `Employee Identifier` = col_character(),
##   `Original Hire Date` = col_character()
## )
```

```
df2
```

```
## # A tibble: 23 x 7
##   `Office Name` `Job Code` `Job Title`      `Base Pay` `Position ID`
##   <chr>        <int> <chr>        <chr>        <int>
## 1 STATES ATTORN~    1172 Assistant State's A~ $20,088.00    9510200
## 2 STATES ATTORN~    1172 Assistant State's A~ $23,436.00    9510200
## 3 STATES ATTORN~    1172 Assistant State's A~ $20,422.82    9510200
## 4 STATES ATTORN~    1172 Assistant State's A~ $23,904.80    9510200
## 5 STATES ATTORN~    1172 Assistant State's A~ $20,745.80    9510200
## 6 STATES ATTORN~    1172 Assistant State's A~ $24,473.38    9510200
## 7 STATES ATTORN~    1172 Assistant State's A~ $21,217.35    9510200
## 8 COUNTY ASSESS~    5049 Residential Model S~ $17,770.86    9500731
## 9 COUNTY ASSESS~    5049 Residential Model S~ $20,800.67    9500731
## 10 COUNTY ASSESS~    5049 Residential Model S~ $17,873.76    9500731
## # ... with 13 more rows, and 2 more variables: `Employee
## #   Identifier` <chr>, `Original Hire Date` <chr>
```

Fixed width files

Sometimes you'll get data with fixed width columns.

It'll look like this.

03/04/2013											Page	1
Period 01 Thru 03												
4:16 pm												
Company 200												
Entry	Per.	Post Date	GL Account	Description	Srce.	Cflow	Ref.	Post	Debit	Credit	Alloc.	
16524	01	10/17/2012	3930621977	TXNPUES	S1	Yes	RHMXPWCP	Yes		5,007.10	No	
191675	01	01/14/2013	2368183100	OUNHQEX XUFQONY	S1	No		Yes		43,537.00	Yes	
191667	01	01/14/2013	3714468136	GHAKASC QHJXDFM	S1	Yes		Yes	3,172.53		Yes	
191673	01	01/14/2013	2632703881	PAHFSAP LUVIKXZ	S1	No		Yes	983.21		No	
80495	01	11/21/2012	2766389794	XDZANTV	S1	Yes	TGZGMOXG	Yes		903.78	Yes	
80507	01	11/21/2012	4609266335	BWWYEZL	S1	Yes	USUKVMZO	Yes		670.31	No	
80509	01	11/21/2012	1092717420	QJYPKVO	S1	No	DNUNTASS	Yes		848.50	Yes	
80497	01	11/21/2012	3386366766	SOQLCMU	S1	Yes	BRHUMGJR	Yes		7.31	Yes	
191669	01	01/14/2013	5905893739	FYIWNKA QUAFDKD	S1	Yes		Yes	9,167.93		Yes	
191671	01	01/14/2013	2749355876	CBMJTLP NGFSEIS	S1	Yes		Yes	746.70		Yes	
191674	01	01/14/2013	4530359106	OTAVZGH ZUQFISZ	S1	Yes		No	7,035.74		Yes	
244819	01	02/04/2013	4679391677	EGHLQTI ABE	S1	Yes		No		89,947.13	No	
96062	01	11/30/2012	5996493062	KTSVTADFF EHEHFMX	S1	Yes	UBNQLRCC	Yes	7.10		Yes	
16527	01	10/17/2012	5595769375	ILCVJYC	S1	Yes	HCVZOOMY	Yes		321.19	Yes	
191670	01	01/14/2013	1948028853	RPPDCWC UWODNIO	S1	Yes		No	9,293.80		No	
191672	01	01/14/2013	4938823703	CTMDXXP HXOXVFF	S1	Yes		No	175.00		Yes	
191668	01	01/14/2013	4207018603	DBZZULF QGDZQMD	S1	Yes		Yes	206.26		Yes	

Just use the `read_fwf()` function from the **readr** package.

This is what it needs– pulled from typing `?read_fwf` in the console:

```
read_fwf(file, col_positions, col_types = NULL, locale = default_locale(),
  na = c("", "NA"), comment = "", trim_ws = TRUE, skip = 0,
  n_max = Inf, guess_max = min(n_max, 1000), progress = show_progress())
```

A couple of important things you need for this to work:

- Pass the widths of each column to the variable
- The names of those columns

There are many methods for this, so be sure to check out the documentation.

After looking at the raw data, the header starts on line 7. So be sure to pass that information to the

```
data_location <- "data/fixed_width_example.txt"
```

```
fixed_example <- read_fwf(data_location, skip=9, fwf_widths(c(8, 2, 12, 12, 29, 3,6, 9, 5, 18, 20, 8),
```

```
head(fixed_example)
```

```
## # A tibble: 6 x 12
##   entry per post_date gl_account description source cflow ref post
##   <int> <chr> <chr>      <dbl> <chr>      <chr> <chr> <chr> <chr>
## 1 16524 01    10/17/2012 3930621977 TXNPUES     S1    Yes  RHM~  Yes
## 2 191675 01    01/14/2013 2368183100 OUNHQEX XUF~ S1    No   <NA>  Yes
## 3 191667 01    01/14/2013 3714468136 GHAKASC QHJ~ S1    Yes  <NA>  Yes
## 4 191673 01    01/14/2013 2632703881 PAHFSAP LUV~ S1    No   <NA>  Yes
## 5 80495 01    11/21/2012 2766389794 XDZANTV     S1    Yes  TGZG~  Yes
## 6 80507 01    11/21/2012 4609266335 BWWYEZL     S1    Yes  USUK~  Yes
## # ... with 3 more variables: debit <dbl>, credit <dbl>, alloc <chr>
```

So the example above took a lot of work– I had to manually count the spaces of each column and then pass on the column names for each one. Sometimes there's a data dictionary that includes all this, which makes it way easier.