WWDC 2021 is on this week and many new fun things are being introduced, including some data science-friendly additions to the frameworks that come with Xcode 13 and available on macOS 12+ (and its *OS cousins).

Specifically, Apple has made tabular data a first-class citizen with the new TabularData app service.

A future post will have some more expository, but here's a sample of core operations including:

- reading in tabular data from CSV or JSON
- examining the structure
- working with columns and/or rows
- grouping and filtering operations
- transforming and removing columns

I've tagged this with `rstats` as there are R equivalents included for each operation so R folks can translate any Swift code they see in the future.

```
import TabularData

// define some basic formatting options for data frame output
let dOpts = FormattingOptions(maximumLineWidth: 80, maximumCellWidth:
10, maximumRowCount: 20, includesColumnTypes: true)

// read in a CSV file
// R: xdf <- read.csv("mtcars.csv")
var xdf = try! DataFrame.init(contentsOfCSVFile: URL(fileURLWithPath:
"mtcars.csv"))

// take a look at it
// R: print(xdf) # no more print() in further R equivalents; just
assume interactive or wrap with print
print(xdf.description(options: dOpts))
```

```
┌─────┬──────────┬──────────┬──────────┬──────────┬──────────┬──────────┬──────┐
│     │ mpg      │ cyl      │ disp     │ hp       │ drat     │ wt       │ 5
┆
│     │ <Double> │ <Int>    │ <Double> │ <Int>    │ <Double> │ <Double> │ more
┆
├─────┼──────────┼──────────┼──────────┼──────────┼──────────┼──────────┼──────┤
│ 0   │ 21.0     │ 6        │ 160.0    │ 110      │ 3.9      │ 2.62     │
┆
│ 1   │ 21.0     │ 6        │ 160.0    │ 110      │ 3.9      │ 2.875    │
┆
│ 2   │ 22.8     │ 4        │ 108.0    │ 93       │ 3.85     │ 2.32     │
┆
│ 3   │ 21.4     │ 6        │ 258.0    │ 110      │ 3.08     │ 3.215    │
┆
│ 4   │ 18.7     │ 8        │ 360.0    │ 175      │ 3.15     │ 3.44     │
┆
│ 5   │ 18.1     │ 6        │ 225.0    │ 105      │ 2.76     │ 3.46     │
```

```
| 6  | 14.3     | 8     | 360.0    | 245   | 3.21     | 3.57     |
| 7  | 24.4     | 4     | 146.7    | 62    | 3.69     | 3.19     |
| 8  | 22.8     | 4     | 140.8    | 95    | 3.92     | 3.15     |
| 9  | 19.2     | 6     | 167.6    | 123   | 3.92     | 3.44     |
| 10 | 17.8     | 6     | 167.6    | 123   | 3.92     | 3.44     |
| 11 | 16.4     | 8     | 275.8    | 180   | 3.07     | 4.07     |
| 12 | 17.3     | 8     | 275.8    | 180   | 3.07     | 3.73     |
| 13 | 15.2     | 8     | 275.8    | 180   | 3.07     | 3.78     |
| 14 | 10.4     | 8     | 472.0    | 205   | 2.93     | 5.25     |
| 15 | 10.4     | 8     | 460.0    | 215   | 3.0      | 5.424    |
| 16 | 14.7     | 8     | 440.0    | 230   | 3.23     | 5.345    |
| 17 | 32.4     | 4     | 78.7     | 66    | 4.08     | 2.2      |
| 18 | 30.4     | 4     | 75.7     | 52    | 4.93     | 1.615    |
| 19 | 33.9     | 4     | 71.1     | 65    | 4.22     | 1.835    |
+----+----------+-------+----------+-------+----------+----------+------+
| 12 more                                                                |
+------------------------------------------------------------------------+
```

```
// dimensions
// R: dim(xdf)
print(xdf.shape)
```

```
(rows: 32, columns: 11)
```

```
// head
// R: head(xdf)
print(xdf.prefix(5).description(options: dOpts))
```

```
+---+----------+-------+----------+-------+----------+----------+------+
|   | mpg      | cyl   | disp     | hp    | drat     | wt       | 5    |
|   | <Double> | <Int> | <Double> | <Int> | <Double> | <Double> | more |
+---+----------+-------+----------+-------+----------+----------+------+
| 0 | 21.0     | 6     | 160.0    | 110   | 3.9      | 2.62     |
| 1 | 21.0     | 6     | 160.0    | 110   | 3.9      | 2.875    |
```

```
| 2 | 22.8     | 4     | 108.0    | 93    | 3.85     | 2.32     |
┆
| 3 | 21.4     | 6     | 258.0    | 110   | 3.08     | 3.215    |
┆
| 4 | 18.7     | 8     | 360.0    | 175   | 3.15     | 3.44     |
┆
└───┴──────────┴───────┴──────────┴───────┴──────────┴──────────┴──────┘

// tail
// R: tail(xdf)
print(xdf.suffix(5).description(options: dOpts))

┌───┬──────────┬──────┬──────────┬──────┬──────────┬──────────┬──────┐
│   | mpg      | cyl  | disp     | hp   | drat     | wt       | 5
┆
│   | <Double> | <Int> | <Double> | <Int> | <Double> | <Double> | more
┆
├───┼──────────┼──────┼──────────┼──────┼──────────┼──────────┼──────┤
| 27 | 30.4     | 4     | 95.1     | 113   | 3.77     | 1.513    |
┆
| 28 | 15.8     | 8     | 351.0    | 264   | 4.22     | 3.17     |
┆
| 29 | 19.7     | 6     | 145.0    | 175   | 3.62     | 2.77     |
┆
| 30 | 15.0     | 8     | 301.0    | 335   | 3.54     | 3.57     |
┆
| 31 | 21.4     | 4     | 121.0    | 109   | 4.11     | 2.78     |
┆
└───┴──────────┴───────┴──────────┴───────┴──────────┴──────────┴──────
──────┴──────┘

// column summaries
// summary(xdf)
print(xdf.summaryOfAllColumns().description(options: dOpts))

┌───┬────────────┬────────────┬────────────┬────────────┬────────────┬──────┐
│   | count(mpg) | uniqueCou… | top(mpg)   | topFreque… | count(cyl) | 39
┆
│   | <Int>      | <Int>      | <Double>   | <Int>      | <Int>      |
more ┆
├───┼────────────┼────────────┼────────────┼────────────┼────────────┼──────┤
| 0 | 32         | 25         | 21.4       | 2          | 32         |
┆
└───┴────────────┴────────────┴────────────┴────────────┴────────────┴──────
────────┴──────┘

// sort it
// R: library(tidyverse) # assume this going forward for R examples
// R: arrange(xdf, cyl)
xdf.sort(on: "cyl")

print(xdf.description(options: dOpts))
```

	mpg	cyl	disp	hp	drat	wt	5
	<Double>	<Int>	<Double>	<Int>	<Double>	<Double>	more
0	22.8	4	108.0	93	3.85	2.32	
1	24.4	4	146.7	62	3.69	3.19	
2	22.8	4	140.8	95	3.92	3.15	
3	32.4	4	78.7	66	4.08	2.2	
4	30.4	4	75.7	52	4.93	1.615	
5	33.9	4	71.1	65	4.22	1.835	
6	21.5	4	120.1	97	3.7	2.465	
7	27.3	4	79.0	66	4.08	1.935	
8	26.0	4	120.3	91	4.43	2.14	
9	30.4	4	95.1	113	3.77	1.513	
10	21.4	4	121.0	109	4.11	2.78	
11	21.0	6	160.0	110	3.9	2.62	
12	21.0	6	160.0	110	3.9	2.875	
13	21.4	6	258.0	110	3.08	3.215	
14	18.1	6	225.0	105	2.76	3.46	
15	19.2	6	167.6	123	3.92	3.44	
16	17.8	6	167.6	123	3.92	3.44	
17	19.7	6	145.0	175	3.62	2.77	
18	18.7	8	360.0	175	3.15	3.44	
19	14.3	8	360.0	245	3.21	3.57	

12 more

```
// read in a JSON File
// R: xdf2 <- jsonlite::fromJSON("mtcars.json")
var xdf2 = try! DataFrame.init(contentsOfJSONFile: URL(fileURLWithPath:
"mtcars.json"))

// bind the rows together
// R: xdf <- bind_rows(xdf, xdf2)
xdf.append(xdf2)

// get the new summary
// R: summary(xdf)
print(xdf.summaryOfAllColumns().description(options: dOpts))
```

```
┌──┬──────────┬──────────┬──────────┬──────────┬──────────┬──────┐
│  │ count(mpg) │ uniqueCou… │ top(mpg) │ topFreque… │ count(cyl) │ 39
┆
│  │ <Int>    │ <Int>    │ <Double> │ <Int>    │ <Int>    │
more ┆
├──┼──────────┼──────────┼──────────┼──────────┼──────────┼──────┤
│ 0 │ 64       │ 25       │ 21.4     │ 4        │ 64       │
┆
└──┴──────────┴──────────┴──────────┴──────────┴──────────┴─────
──────┴──────┘
```

```
// basic filtering
// R: xdf.filter(cyl == 6)
print( xdf.filter(on: "cyl", Int.self) { (val) in val == 6 } )
```

```
┌──┬────────┬─────┬────────┬─────┬────────┬────────┬──────┐
│  │ mpg    │ cyl │ disp   │ hp  │ drat   │ wt     │ 5
┆
│  │ <Double> │ <Int> │ <Double> │ <Int> │ <Double> │ <Double> │ more
┆
├──┼────────┼─────┼────────┼─────┼────────┼────────┼──────┤
│ 11 │ 21.0   │ 6   │ 160.0  │ 110 │ 3.9    │ 2.62   │
┆
│ 12 │ 21.0   │ 6   │ 160.0  │ 110 │ 3.9    │ 2.875  │
┆
│ 13 │ 21.4   │ 6   │ 258.0  │ 110 │ 3.08   │ 3.215  │
┆
│ 14 │ 18.1   │ 6   │ 225.0  │ 105 │ 2.76   │ 3.46   │
┆
│ 15 │ 19.2   │ 6   │ 167.6  │ 123 │ 3.92   │ 3.44   │
┆
│ 16 │ 17.8   │ 6   │ 167.6  │ 123 │ 3.92   │ 3.44   │
┆
│ 17 │ 19.7   │ 6   │ 145.0  │ 175 │ 3.62   │ 2.77   │
┆
│ 32 │ 21.0   │ 6   │ 160.0  │ 110 │ 3.9    │ 2.62   │
┆
│ 33 │ 21.0   │ 6   │ 160.0  │ 110 │ 3.9    │ 2.875  │
┆
```

```
| 35 | 21.4     | 6     | 258.0    | 110   | 3.08     | 3.215    |
┆
├────┼──────────┼───────┼──────────┼───────┼──────────┼──────────┼──────┤
┆ 4 more
┆
└──────────────────────────────────────────────────────────────────────┘

// group by a column
// R: group_by(xdf, cyl)
print(xdf.grouped(by: "cyl"))

4
┌────┬──────────┬───────┬──────────┬───────┬──────────┬──────────┬──────┐
┆    | mpg      | cyl   | disp     | hp    | drat     | wt       | 5    ┆
┆    | <Double> | <Int> | <Double> | <Int> | <Double> | <Double> | more ┆
├────┼──────────┼───────┼──────────┼───────┼──────────┼──────────┼──────┤
| 0  | 22.8     | 4     | 108.0    | 93    | 3.85     | 2.32     |
┆
| 1  | 24.4     | 4     | 146.7    | 62    | 3.69     | 3.19     |
┆
| 2  | 22.8     | 4     | 140.8    | 95    | 3.92     | 3.15     |
┆
| 3  | 32.4     | 4     | 78.7     | 66    | 4.08     | 2.2      |
┆
| 4  | 30.4     | 4     | 75.7     | 52    | 4.93     | 1.615    |
┆
| 5  | 33.9     | 4     | 71.1     | 65    | 4.22     | 1.835    |
┆
| 6  | 21.5     | 4     | 120.1    | 97    | 3.7      | 2.465    |
┆
| 7  | 27.3     | 4     | 79.0     | 66    | 4.08     | 1.935    |
┆
| 8  | 26.0     | 4     | 120.3    | 91    | 4.43     | 2.14     |
┆
| 9  | 30.4     | 4     | 95.1     | 113   | 3.77     | 1.513    |
┆
├────┼──────────┼───────┼──────────┼───────┼──────────┼──────────┼──────┤
┆ 12 more
┆
└──────────────────────────────────────────────────────────────────────┘

6
┌────┬──────────┬───────┬──────────┬───────┬──────────┬──────────┬──────
┆    | mpg      | cyl   | disp     | hp    | drat     | wt       | 5
┆
┆    | <Double> | <Int> | <Double> | <Int> | <Double> | <Double> | more
┆
├────┼──────────┼───────┼──────────┼───────┼──────────┼──────────┼──────┤
| 11 | 21.0     | 6     | 160.0    | 110   | 3.9      | 2.62     |
┆
| 12 | 21.0     | 6     | 160.0    | 110   | 3.9      | 2.875    |
┆
```

13	21.4	6	258.0	110	3.08	3.215	
14	18.1	6	225.0	105	2.76	3.46	
15	19.2	6	167.6	123	3.92	3.44	
16	17.8	6	167.6	123	3.92	3.44	
17	19.7	6	145.0	175	3.62	2.77	
32	21.0	6	160.0	110	3.9	2.62	
33	21.0	6	160.0	110	3.9	2.875	
35	21.4	6	258.0	110	3.08	3.215	

4 more

8

	mpg	cyl	disp	hp	drat	wt	5
	<Double>	<Int>	<Double>	<Int>	<Double>	<Double>	more
18	18.7	8	360.0	175	3.15	3.44	
19	14.3	8	360.0	245	3.21	3.57	
20	16.4	8	275.8	180	3.07	4.07	
21	17.3	8	275.8	180	3.07	3.73	
22	15.2	8	275.8	180	3.07	3.78	
23	10.4	8	472.0	205	2.93	5.25	
24	10.4	8	460.0	215	3.0	5.424	
25	14.7	8	440.0	230	3.23	5.345	
26	15.5	8	318.0	150	2.76	3.52	
27	15.2	8	304.0	150	3.15	3.435	

18 more

```
// number of groups
// R: group_by(xdf, cyl) %>% group_keys() %>% nrow()
print(xdf.grouped(by: "cyl").count)

3

// group, manipulate (in this case, filter), and re-combine
// R: group_by(xdf) %>% filter(mpg < 20) %>% ungroup()
print(
  xdf.grouped(by: "cyl").mapGroups { (val) in
    val.filter(on: "mpg", Double.self) { (val) in val! < 20 }.base
  }.ungrouped()
)
```

	mpg	disp	hp	drat	wt	qsec	5 more
	<Double>	<Double>	<Int>	<Double>	<Double>	<Double>	
0	22.8	108.0	93	3.85	2.32	18.61	
1	24.4	146.7	62	3.69	3.19	20.0	
2	22.8	140.8	95	3.92	3.15	22.9	
3	32.4	78.7	66	4.08	2.2	19.47	
4	30.4	75.7	52	4.93	1.615	18.52	
5	33.9	71.1	65	4.22	1.835	19.9	
6	21.5	120.1	97	3.7	2.465	20.01	
7	27.3	79.0	66	4.08	1.935	18.9	
8	26.0	120.3	91	4.43	2.14	16.7	
9	30.4	95.1	113	3.77	1.513	16.9	

182 more

```
// look at one column
// R: xdf$cyl
print( xdf["cyl"] )
```

cyl

```
| <Int> |
|-------|
| 4     |
| 4     |
| 4     |
| 4     |
| 4     |
| 4     |
| 4     |
| 4     |
| 4     |
| 4     |
|-------|
| 54 m… |
`-------'
```

```
// combine two columns and look at it
// R: mutate(xdf, cyl_mpg = sprintf("%s:%s", cyl, mpg) %>% select(-cyl,
-mpg)
// R: unite(xdf, cyl_mpg, cyl, mpg, sep = ":") # alternate way
xdf.combineColumns("cyl", "mpg", into: "cyl_mpg") { (val1: Int?, val2:
Double?) -> String in
  String(val1 ?? 0) + ":" + String(val2 ?? 0.0)
}

print(xdf["cyl_mpg"])
```

```
,----------.
| cyl_mpg  |
| <String> |
|----------|
| 4:22.8   |
| 4:24.4   |
| 4:22.8   |
| 4:32.4   |
| 4:30.4   |
| 4:33.9   |
| 4:21.5   |
| 4:27.3   |
| 4:26.0   |
| 4:30.4   |
|----------|
| 54 more  |
`----------'
```

```
// look at the colnames (^^ removes "cyl" and "mpg"
// R: colnames(xdf)
print(xdf.columns.map{ col in col.name })

["cyl_mpg", "disp", "hp", "drat", "wt", "qsec", "vs", "am", "gear",
"carb"]
```

```
// turn an Int into a Double
// R: xdf$hp <- as.double(xdf$hp) # or use dplyr::mutate()
xdf.transformColumn("hp") { (val1: Int?) -> Double? in
  Double(val1 ?? 0)
}

print(xdf["hp"])
```

```
┌─────────┐
│ hp      │
│ <Double>│
├─────────┤
│ 93.0    │
│ 62.0    │
│ 95.0    │
│ 66.0    │
│ 52.0    │
│ 65.0    │
│ 97.0    │
│ 66.0    │
│ 91.0    │
│ 113.0   │
├─────────┤
┆ 54 more ┆
└─────────┘
```

```
// look at the coltypes
// R: sapply(mtcars, typeof)
print(xdf.columns.map{ col in col.wrappedElementType })

[Swift.String, Swift.Double, Swift.Double, Swift.Double, Swift.Double,
Swift.Double, Swift.Int, Swift.Int, Swift.Int, Swift.Int]

// distinct horsepower
// R: distinct(xdf, hp)
print(xdf["hp"].distinct())
```

```
┌─────────┐
│ hp      │
│ <Double>│
├─────────┤
│ 93.0    │
│ 62.0    │
│ 95.0    │
│ 66.0    │
│ 52.0    │
│ 65.0    │
│ 97.0    │
│ 91.0    │
│ 113.0   │
│ 109.0   │
├─────────┤
```

```
┆ 12 more  ┆
└──────────┘

// row slices
// R: xdf[10,]
print(xdf.rows[10])

┌──┬─────────┬─────────┬─────────┬─────────┬─────────┬─────────┬──────┐
│  │ cyl_mpg │ disp    │ hp      │ drat    │ wt      │ qsec    │
4  ┆
│  │ <String>│ <Double>│ <Double>│ <Double>│ <Double>│ <Double>│
more ┆
├──┼─────────┼─────────┼─────────┼─────────┼─────────┼─────────┼──────┤
│ 10 │ 4:21.4  │ 121.0   │ 109.0   │ 4.11    │ 2.78    │ 18.6
│    ┆
└──┴─────────┴─────────┴─────────┴─────────┴─────────┴─────────
┴──────────┴──────┘

// R: xdf[3:10,]
print(xdf.rows[3...10])

Rows(base:
┌──┬─────────┬─────────┬─────────┬─────────┬─────────┬─────────┬──────┐
│  │ cyl_mpg │ disp    │ hp      │ drat    │ wt      │ qsec    │
4  ┆
│  │ <String>│ <Double>│ <Double>│ <Double>│ <Double>│ <Double>│
more ┆
├──┼─────────┼─────────┼─────────┼─────────┼─────────┼─────────┼──────┤
│ 0 │ 4:22.8  │ 108.0   │ 93.0    │ 3.85    │ 2.32    │ 18.61   │
┆
│ 1 │ 4:24.4  │ 146.7   │ 62.0    │ 3.69    │ 3.19    │ 20.0    │
┆
│ 2 │ 4:22.8  │ 140.8   │ 95.0    │ 3.92    │ 3.15    │ 22.9    │
┆
│ 3 │ 4:32.4  │ 78.7    │ 66.0    │ 4.08    │ 2.2     │ 19.47   │
┆
│ 4 │ 4:30.4  │ 75.7    │ 52.0    │ 4.93    │ 1.615   │ 18.52   │
┆
│ 5 │ 4:33.9  │ 71.1    │ 65.0    │ 4.22    │ 1.835   │ 19.9    │
┆
│ 6 │ 4:21.5  │ 120.1   │ 97.0    │ 3.7     │ 2.465   │ 20.01   │
┆
│ 7 │ 4:27.3  │ 79.0    │ 66.0    │ 4.08    │ 1.935   │ 18.9    │
┆
│ 8 │ 4:26.0  │ 120.3   │ 91.0    │ 4.43    │ 2.14    │ 16.7    │
┆
│ 9 │ 4:30.4  │ 95.1    │ 113.0   │ 3.77    │ 1.513   │ 16.9    │
┆
├──┴─────────┴─────────┴─────────┴─────────┴─────────┴─────────┴──────┤
┆ 54 more
┆
└──────────────────────────────────────────────────────────────────┘
```

```
, subranges: _RangeSet(3..<11))
```