Notes about Org

with examples

Tarak Kharrat[1] and Georgi N. Boshnakov[2]

[1]Salford Business School, University of Salford, UK.
[2]School of Mathematics, University of Manchester, UK.

These notes are when working with Org version 9.1.2.

1 General notes

- If the code of the chunks is evaluated when exporting the file, e.g. `C-c C-e l p` , the results in the org file are not updated (the updated stuff appears in the exported document). Maybe this depends on options but we better be aware of this.
  Note that the results of calculation of inline code are inserted in the buffer.
-
- TODO: use the facility from main.pdf to provide visual distinction between code and results.
- Exporting nothing (`:exports none`)

Here is a chunk that loads `xtable` but has `:exports none` and should not appear. This is currently unpredictable. Even during a single session it may reappear!

- I changed the inline blocks to return raw values (for the moment), since it seems better that the numbers are passed to TEX wthout being wrapped in texttt.
  However 'raw' has the unfortunate effect that it doesn't wrap the result of the inline block in ..., so repeated evaluation will produce repeated values.
  TODO - there may be some option to avoid this, otherwise 'raw' may have to be removed. (but see below for a workflow that woeks until the whole file is recalculated.
- I adopted the following workflow while editing:
  i. Most of the time, I simply export the (`C-c C-e l p`). This recomputed the inline blocks but doesn't put the results in the org buffer, only in the exported TEX file.
  ii. When I change a chunk, I recalculate only that chunk (`C-c C-c`). Then export as in i.
  TODO: This will fail for objects with the same names are at different places in the doc. Change them.
  The trouble is that if the whole file is recomputed (`C-c C-v C-b`) the inline chunks with the 'raw' option are computed and the results put in the file (as noted above). Then during export all such results are duplicated.

2 Formatting tables

Org provides some formatting and produces tables from matrices and data frames but seems to ignore the column names and may be all attributes.

We will use `xtable` for some printing.

```
library(xtable)
```

Here we print (a subset of) a data frame, as `R` would print it on the console. We add a character column to be sure that this works with mixed type columns.

```
fewCars <- head(cars)
fewCars$char <- letters[1:nrow(fewCars)]
fewCars$num <- 1:nrow(fewCars) * pi
fewCars
```

```
  speed dist char       num
1     4    2    a  3.141593
2     4   10    b  6.283185
3     7    4    c  9.424778
4     7   22    d 12.566371
5     8   16    e 15.707963
6     9   10    f 18.849556
```

A proper table can be obtained using `xtable`.

```
xfew <- xtable(fewCars, digits = 5)
print(xfew, floating = FALSE)
```

	speed	dist	char	num
1	4.00000	2.00000	a	3.14159
2	4.00000	10.00000	b	6.28319
3	7.00000	4.00000	c	9.42478
4	7.00000	22.00000	d	12.56637
5	8.00000	16.00000	e	15.70796
6	9.00000	10.00000	f	18.84956

It is worth storing the result of calling `xtable` in a variable and printing it separately, since the function `xtable()` has one set of options, while the print method for objects of type `xtable` has other options. For example:

```
xfew <- xtable(fewCars, caption = "Information about some cars.")
print(xfew)
```

	speed	dist	char	num
1	4.00	2.00	a	3.14
2	4.00	10.00	b	6.28
3	7.00	4.00	c	9.42
4	7.00	22.00	d	12.57
5	8.00	16.00	e	15.71
6	9.00	10.00	f	18.85

Table 1: Information about some cars.