brain2.r: Explanation of R code

Goal:
   demonstrate ways to include quadratic and cross-product terms in a multiple regression

This uses the brain data set, in brain.txt. The first few lines read the data and compute the log transform of each variable.

Storing a quadratic or cross product term in a new variable:
```
brain$logbody2 <- brain$logbody^2
```
Much of this is familiar. The `^` operator is the "raise to a power" operator. `x^2` computes x squared. `x^3` would compute x cubed. You could also write `brain$logbody*brain$logbody`.

Crossproduct terms are computed and saved in the obvious way.

The quadratic or crossproduct variable is used in a multiple regression just like any other variable. This is illustrated by the first `summary(lm())`.

Computing a quadratic term "on the fly": `I(logbody^2)`
You do not need to create a new variable. You can compute a quadratic term in the lm() model. It would make sense to write `logbody^2`, but that doesn't work because of the way R interprets formulae. You have to "protect" the computation by enclosing the computation in `I( )`.

If you don't say a quadratic term in the summary of what is supposed to be a quadratic model, you probably forgot the `I()`.

Computing cross product terms "on the fly": `logbody:loglitter`
Because cross product terms are commonly used in models, R makes it easy to produce them. Just combine the two variables using a colon in between. You do not need to protect this computation.

Short cut to write a models with cross product terms: `logbody*loglitter`
The `*` makes it even easier to write models with cross product terms. `A*B` is interpreted as A + B + A:B. So the results of the model fit with `logbody*loglitter` are the same as those from writing out all three terms.

Cross products of continuous and factor variables:
Nothing new - Write the cross product in the model formula. There, R doesn't care whether the components are two continuous variables, a continuous and a factor, or two factor variables.