Doing linear regression on the computer requires the following skills

Plotting the data. Suppose the data is stored as two variables `x` and `y`. Then either of these two commands will make a scatterplot

```
> plot(x,y)
> plot(y ~ x)
```

Notice the position of the `x` and `y`. When the tilde is used the response (`y`) variable goes on the left.

If the data is stored in a dataframe, `d` with column names `x` and `y` you must tell the computer where to find the names. This can be done by attaching `d` as in

```
> attach(d)
> plot(y ~ x)
```

Or, by adding the `data = d` command to `plot()`

```
> plot(y ~ x, data = d)
```

Suppose in what follows you have certain values of `x` and `y`

```
> x = c(1,2,3,4,5)
> y = c(1,3,2,4,5)
```

Finding correlations To find the correlation between `x` and `y` we use the `cor()` function:

```
> cor(x,y)
[1] 0.9
```

Finding the regression coefficients To find the regression coefficients requires a new function, `lm()`. It is used like the `plot()` command above

```
> lm(y ~ x)                     # also with data=d

Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)            x
        0.3          0.9
```

This will return the two coeffiecients. Often it is best to store the output in variable as with

```
> res = lm(y ~ x, data = d)
> res

Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)            x
        0.3          0.9
```

Plotting the regression line The `abline()` function will add a regression line. You use the result of the `lm()` function

```
> plot(y ~ x)
> abline(res)
```

Makeing predictions The `lm()` returns the estiamates for β_0 and β_1. Recall the predicted value for a given x is just $\hat{y} = \beta_0 + \beta_1 x$ In our example, the predicted value for $x = 1.5$ would be

```
> 0.3 + 0.9 * 1.3
[1] 1.47
```

1 The babies dataset

We will look at data in the babies dataset. This dataset has a number of variables recorded about newborn babies in California in the 60's. To read in the data type these commands exactly

```
> f = "http://www.math.csi.cuny.edu/verzani/R/regression.R"
> source(url(f))
```

Now a variable `babies` exists. To view it you can type

```
> edit(babies)
```

It's a big dataset with many variables.
First we attach it so the variables are available without much fuss

```
> attach(babies)
```

Nest we look at the relationship between the Mother's age `age` and the Father's age `dage`. We can plot it with

```
> plot(age ~ dage)
```

A linear trend is apparent: older men have older wives, although there are exceptions.

1.1 Explain why the line $y = x$ is not in the center of the data.

What is the correlation?

```
> cor(age,dage)
[1] 0.8283
```

Positive and close to 1, but we could guess that couldn't we?

1.2 Why is the correlation close to 1? Why is it positive?

We can find the regression line with

```
> res = lm(age ~ dage)
res
>
Call:
lm(formula = age ~ dage)

Coefficients:
(Intercept)         dage
      5.515        0.719
```

The regression line is added to the scatterplot with

```
> abline(res)
```

Finally, what is the predicted age for a 39 year old male?

```
> 5.515 + 0.719 * 39
[1] 33.56
```

2 More problems

2.1 Is there a relationship between birthweight `bwt` and gestation time `gestation`? Do all of the above for these two variables, and comment. That is find the correlation, the regression coefficients, make the plot with regression and discuss the linear model for this relationship.

2.2 Look at the relationship between dad's height `dht` and dad's weight `dwt`. Find all of the above, and discuss if the linear model seems appropriate.

2.3 Repeat with the mothers height and weight. (`ht` and `wt`)

2.4 Make histograms of `ht`,`wt`,`dht` and `dwt` and see if any of the data sets look normally distributed. Discuss.

2.5 The command `pairs(babies)` will make several different scatterplots at once. Which seem to have the largest value of R? Which are the closest to 0?