There is an R code set at the end of this lab and posted on the course web page that will prove useful for doing this lab. **Use it in answering/doing the following.** This lab concerns computing and the Glass/Phosphor example used in lecture.

1. Make a data frame for use in this example. Include "glasslevel," "phosphorlevel," "tubetype," and "current" variables, treating the first 3 as "factor" variables. Print out the data frame and the default `summary()` applied to the data frame.

2. Plot the current requirement against the tubetype variable. Then plot the current requirement against the phosphorlevel variable. Why does the second of these fail to make much sense (fail to be useful)?

3. Set an option for your session that will make R adopt the definition of effects that we used in lecture by issuing the command

```r
> options(contrasts = rep("contr.sum", 2))
```

Then do a one-way analysis of the data, using `lm()` and the "factor" tubetype. Compute $s_p$ using the one-way output and then a "margin of error" to attach to each of the $r = 6$ sample means, namely

$$\Delta = t \frac{s_p}{\sqrt{m}}$$

4. Use the `Hmisc` and `stats` packages to make the kind of interaction plot enhanced with the $\pm \Delta$ error bars illustrated in class. (You may need to install the `Rtools` package before trying to install the `Hmisc` package in order for it to install correctly.)

5. In the `lm()` output, exactly how are the "estimated coefficients" related to the 6 sample means? That is, show which b's get added to make each one of the $\bar{y}_i$'s.

6. Now do a two-way analysis of the current requirements using `lm()`. Exactly what are the 6 estimated coefficients produced by the `lm()` call?

7. Find a $p$-value for testing $H_0 : a \beta_j = 0 \ \forall i, j$ (i.e. that there are no interactions between glass and phosphor) on the two-way output. In rough terms, how is it related to the confidence intervals for Glass-Phosphor interactions presented in lecture?

8. Redo the one-way analysis of the current requirements data using dummy variables and a multiple linear regression viewpoint. (See Section 9.3.2 of Vardeman and Jobe.) How are the coefficients produced related to those on the one-way output? How is $s_{sp}$ related to $s_p$?
9. Redo the two-way analysis of the current requirements data using dummy variables and a multiple linear regression viewpoint. How are the coefficients produced related to those on the two-way output? How is $s_{SF}$ related to $s_p$?

**Code Set for Stat 401B Laboratory #9**

Here is code for Stat 401B Lab 9 and analysis of the Current Requirements Data

First make a data frame

```r
tubetype<-c(rep(1,3),rep(2,3),rep(3,3),rep(4,3),rep(5,3),rep(6,3))
glasslevel<-c(rep(1,9),rep(2,9))
phosphorlevel<-rep(c(rep(1,3),rep(2,3),rep(3,3)),2)

tubetype<-as.factor(tubetype)
glasslevel<-as.factor(glasslevel)
phosphorlevel<-as.factor(phosphorlevel)


GlassPhosphor<-data.frame(tubetype,glasslevel,phosphorlevel,current)
GlassPhosphor
summary(GlassPhosphor)

plot(tubetype,current,xlab="Tube Type",ylab="Current Requirement")
plot(phosphorlevel,current,xlab="Phosphor",ylab="Current Requirement")
```
Before using `lm()` to analyze these data, we must set an option that tells R that we're using the definitions of effects presented in class:

```r
options(contrasts = rep("contr.sum", 2))
```

```r
one.out<-lm(current~tubetype,data=GlassPhosphor)
summary(one.out)
predict(one.out)
anova(one.out)
m<-3
sp<-sqrt(t(one.out$residuals)%*%one.out$residuals/one.out$df.residual)
sp
Delt<-qt(.975,one.out$df.residual)*sp/sqrt(m)
Delt
```

# Install the Hmisc Package and load it

# Load the stats package before proceeding to make an interaction plot

```r
interaction.plot(phosphorlevel,glasslevel,current,fun=mean,type="l", ylim=c(210,320))
errbar(x=1,y=mean(current[1:3]),yminus=mean(current[1:3])-Delt,
      yplus=mean(current[1:3])+Delt,add=TRUE)
errbar(x=2,y=mean(current[4:6]),yminus=mean(current[4:6])-Delt,
      yplus=mean(current[4:6])+Delt,add=TRUE)
errbar(x=3,y=mean(current[7:9]),yminus=mean(current[7:9])-Delt,
      yplus=mean(current[7:9])+Delt,add=TRUE)
errbar(x=1,y=mean(current[10:12]),yminus=mean(current[10:12])-Delt,
      yplus=mean(current[10:12])+Delt,add=TRUE)
```
errbar(x=2,y=mean(current[13:15]),yminus=mean(current[13:15])-Delt, 
        yplus=mean(current[13:15])+Delt,add=TRUE)
errbar(x=3,y=mean(current[16:18]),yminus=mean(current[16:18])-Delt, 
        yplus=mean(current[16:18])+Delt,add=TRUE)

#Now proceed to two-way analysis
summary(lm(current~glasslevel+phosphorlevel+glasslevel:phosphorlevel, 
            data=GlassPhosphor))
predict(lm(current~glasslevel+phosphorlevel+glasslevel:phosphorlevel, 
            data=GlassPhosphor),se.fit=TRUE)

#Here is a Two-way ANOVA for the problem
anova(lm(current~glasslevel+phosphorlevel+glasslevel:phosphorlevel, 
            data=GlassPhosphor))

#Now approach the analysis through MLR (See Section 9.3.2 of Vardeman 
and Jobe)
#First make dummy variables
dumT1<-c(rep(1,3),rep(0,12),rep(-1,3))
dumT2<-c(rep(0,3),rep(1,3),rep(0,9),rep(-1,3))
dumT3<-c(rep(0,6),rep(1,3),rep(0,6),rep(-1,3))
dumT4<-c(rep(0,9),rep(1,3),rep(0,3),rep(-1,3))
dumT5<-c(rep(0,12),rep(1,3),rep(-1,3))
dumA1<-c(rep(1,9),rep(-1,9))
dumB1<-c(rep(c(rep(1,3),rep(0,3),rep(-1,3)),2))
dumB2<-c(rep(c(rep(0,3),rep(1,3),rep(-1,3)),2))
#Now do MLRs and see what is produced for fitted coefficients and ANOVAs

```
summary(lm(current~dumT1+dumT2+dumT3+dumT4+dumT5))
anova(lm(current~dumT1+dumT2+dumT3+dumT4+dumT5))

```