13.1 The cost of a home depends on the number of bedrooms in the house. Suppose the following data is recorded for homes in a given town:

price (in thousands)	300	250	400	550	317	389	425	289	389	559
No. bedrooms	3	3	4	5	4	3	6	3	4	5

Make a scatterplot, and fit the data with a regression line. On the same graph, test the hypothesis that an extra bedroom costs $60,000 against the alternative that it costs more.

Solution:

```
price=scan()
300000 250000 400000 550000 317000 389000 425000 289000 389000 559000
# double enter
bedrooms=scan()
3 3 4 5 4 3 6 3 4 5
# double enter

X=price; Y=bedrooms

lmRes=lm(X ~ Y)
plot(Y, X)
abline(lmRes)

H0: extra bedroom costs 60000 (b1=60000)
HA: extra bedroom costs more (b1>60000)
едностранен тест

e=resid(lmRes)
n=length(e)
Seps=sqrt(sum(e**2)/(n-2)) # Residual standard error on n-2 degrees of freedom
b1h=cov(Y, X)/var(Y) # b1h=coef(lmRes)[['Y']]
SEb1=Seps/sqrt(sum((Y-mean(Y))**2))
b1=60000
t=(b1h-b1)/SEb1
pval=pt(t, n - 2, lower.tail = F)
if(pval>0.05) print("no evidence to reject H0") else print("reject H0 in favour of HA")
# [1] "reject H0 in favour of HA"
```

13.2 It is well known that the more beer you drink, the more your blood alcohol level rises. Suppose we have the following data on student consumption:

Student	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Beers	5	2	9	8	3	7	3	5	3	5
BAL	0.10	0.03	0.19	0.12	0.04	0.095	0.07	0.06	0.02	0.05

Make a scatterplot and fit the data with a regression line. Test the hypothesis that another beer rises your BAL by 0.02 percent against the alternative that it is less.

Solution:

```
BAL=scan()
0.10 0.03 0.19 0.12 0.04 0.095 0.07 0.06 0.02 0.05
# double enter
Beers=scan()
5 2 9 8 3 7 3 5 3 5
# double enter

X=BAL; Y=Beers

lmRes=lm(X ~ Y)
plot(Y, X)
abline(lmRes)

H0: another beer rises your BAL by 0.02 (b1=0.02)
HA: another beer rises your BAL by less than 0.02 (b1<0.02)
едностранен тест

e=resid(lmRes)
n=length(e)
Seps=sqrt(sum(e**2)/(n-2)) # Residual standard error on n-2 degrees of freedom
b1h=cov(Y, X)/var(Y) # b1h=coef(lmRes)[['Y']]
SEb1=Seps/sqrt(sum((Y-mean(Y))**2))
b1=0.02
t=(b1h-b1)/SEb1
pval=pt(t, n - 2, lower.tail = T)
if(pval>0.05) print("no evidence to reject H0") else print("reject H0 in favour of HA")
# [1] "no evidence to reject H0"
```

13.3 For the same Blood alcohol data, do a hypothesis test that the intercept is 0 with a two sided alternative.

Solution:

```
BAL=scan()
0.10 0.03 0.19 0.12 0.04 0.095 0.07 0.06 0.02 0.05
# double enter
Beers=scan()
5 2 9 8 3 7 3 5 3 5
# double enter

X=BAL; Y=Beers

H0: b0=0
HA: b0<>0
двустранен тест

lmRes=lm(X ~ Y)
s=summary(lmRes)
pval=s$coefficients[,4][1]
if(pval>0.05) print("no evidence to reject H0") else print("reject H0 in favour of HA")
# [1] "no evidence to reject H0"
```

13.4 The lapse rate is the rate at which temperature drops as you increase elevation. Some hardy students were interested in checking empirically if the lapse rate of 9.8 degrees C/km was accurate for their hiking. To investigate, they grabbed their thermometers and their Suunto wrist altimeters and found the following data on their hike

elevation(ft)	600	1000	1250	1600	1800	2100	2500	2900
temperature(F)	56	54	56	50	47	49	47	45

Draw a scatter plot with regression line, and investigate if the lapse rate is 9.8C/km. (First, it helps to convert to the rate of change in Fahrenheit per feet with is 5.34 degrees per 1000 feet.) Test the hypothesis that the lapse rate is 5.34 degrees per 1000 feet against the alternative that it is less than this.

Solution:

According to the conditions we have to check if the regression equation is

$$y = \beta_0 - 5{,}34x$$

Let us first build up our simple linear regression model and then to test the hypothesis if the slope is $\beta_1 = -5{,}34$.

```
elevation=c(600,1000,1250,1600,1800,2100,2500,2900)/1000
temperature=c(56,54,56,50,47,49,47,45)

X=elevation
Y=temperature

lmRes=lm(Y~X)
plot(X,Y)
abline(lmRes)
```

H0: b1 = -5,34
HA: b1 < (-5,34)
едностранен тест

```
e=resid(lmRes)
n=length(e)
Seps=sqrt(sum(e**2)/(n-2)) # Residual standard error on n-2 degrees of freedom
b1h=cov(X, Y)/var(X) # b1h=coef(lmRes)[['X']]
SEb1=Seps/sqrt(sum((X-mean(X))**2))
b1=-5.34
t=(b1-b1h)/SEb1
pval=pt(t, n - 2, lower.tail = T)
if(pval>0.05) print("no evidence to reject H0") else print("reject H0 in favour of HA")
# [1] "no evidence to reject H0"
```