```
getwd()
setwd("xx/xx")
install.packages(“ “)
library(name)
        dplyr
        stringr
        DataCombine
        psych
        corrgram
        MASS
read.csv("file_name")
read.csv(file.choose())
str(data_name)
dim(data_name)
filter(data_name, variable, variable1 | variable2 & varibe 3)
mutate(data_name, "name_of_new_column" = new_variable,+,+)
group_by(data_name, variable)
summarise(data_name, variable, sum(variable), mean(variable),length(variable))
write.csv(data_name," file_name.csv ", row.names=FALSE)
write.table(data_name, "file name .txt", row.name=TRUE,sep="\t")
which(is.na)
table(is.na(data_name))
str_subset(data_name$variable,"[a-z,A-Z]")
str_detect(data_name$variable,str_subset_name)
str_replace_all(data_name,variable_name(bad_data),"desired")
as.numeric(data_name)
as.character(data_name)
as.factor(data_name)
factor(variable_name, levels=c(0,1,2,3), labels=c("a","b","c","d"))
ordered=TRUE
varible_name<-mdy_hm(variable_name)
unique(data_name)
tolower(variable$name)
str_rim(variable_name)
variable_name[na_loc]<-"unknown"
rep("value",length(index vector/pattern))
data.frame(from=current, to=replace)
FindReplace(data=data_name, Var=“variable”, new_replace_varible, form="from",to="to",exact=FALSE)
index_vector<-factor(index_vector, ordered=TRUE,levels=c("Low","Medium","High"))
summary(data_name)
sd(variable_name)
vr(variable_name)
sqrt(value)
name(index_vector)
name(index_vector)[column]<-” “
plot(variable_name)
        col="color"
        ylab="naming"
        xlab="naming"
```

```
hist(data_name)
        main="naming"
boxplot(data_name)
par(mfrow=c(value,value)
dev.off()
cut(data_name,value)
mosaicplot(data_name)
boxplot(variable~variable)
cor(data_name,data_name)
cor.test(data_name)
cor.test(data_name)[,value:value]
pairs(data_name)
corrgram(data_name[,x:x],order= ,main=" ", lower.panel = plot_type, upper.panel = plot_type, diag.panel = panel.minmax, text.panel = panel.txt)
        panel.conf
        panel.ellipse
        panel.shade
        panel.pie
cbind(variable,variable)
rbind(variable,variable)
        deparse.level=1
lm(dependent_variable(y) ~ independent_variable(x), data=data_name)
qqnorm(data_variable_lm$variable_in)
qqline(data_name_lm)
abline(0,0,lwd=3);abline(h=c(-6.5,6.5),lwd=3,lty=3)
predict.lm(data_name,new_variable,level= CI_calue, interval ="predict")
boxcox(data_name)
```