a. Identify and clean up any data items that need to be made uniform or transformed. Explain what you did.

We have used Microsoft Excel to clean the dataset to not complicate the task of visualizing the results clearly and transparently. We have categorized the dataset into layers and started the cleaning procedure.

The data has 37 columns originally from which we found out that keeping few columns wouldn't help our visualization in any way and decided to eliminate those columns so that the final data that can be used for generating visuals and summaries would be clear and concise on to point.

Elimination:

In the generalized columns, we have omitted Access code, Email address which are empty and IP Address, Latitude, and Longitude which are needed only to know the location from where the survey was taken. That issue has been already answered by the other existing columns city, state, and country.

In the survey, 11 questions were posted out of which 5 are having choices where the students had to choose from options given. There were few options such as "Non-binary" and "prefer not to answer." from Gender, "other" from the type of laptop used which were not opted by many of the students who took the survey. So, we have eliminated them.

Data in a few columns like Country of citizenship and Undergraduate degree has been identified with different names, they are changed to a common name and changes to the date of graduation are made similarly.

Incomplete surveys: There were 3 incomplete surveys noticed in the dataset which are eliminated as there was no data that they were showing.

Merging:

For a few columns, merging was needed to make the end task simple. The merging took place as follows.

Section (Q1): Categorized the sections where 1,4, DL represents 001,004 and DL1 respectively.

Gender (Q2): one single column for gender where M and F represent male and female respectively.

Laptop type (Q8): As there are only 2 types of laptops being used by all students, we have merged them into one column such that 1 and 2 represent Microsoft/Windows and Apple/MacBook respectively.

Employment status (Q10): 3 types of employments are combined into a single column and categorized as 1 – 3 were "Yes, Full Time", "Working, but not Full Time", "Not Working while attending Mason" are represented by 1,2,3 respectively.

Programming skill in Python (Q11): There are 5 levels of proficiency given here. We have categorized them where “Little/none”, “Some familiarity”, “Average user”, “Frequent use for projects”, “Fluent/expert” are rated from 1 to 5 respectively.
In this way, the dataset has been outfitted from 37 columns to 18 columns and made concise which would be easy to use.

b. Create summary statistics and visualizations for each of the 11 questions, categorized by section (001,004, DL1). For each, explain any interesting differences that you observe (or indicate no real difference)

Importing packages

```
library(ggplot2)
library(tidyr)
library(tidyverse)
library(plotrix)
library(fmsb)
library(RColorBrewer)
```

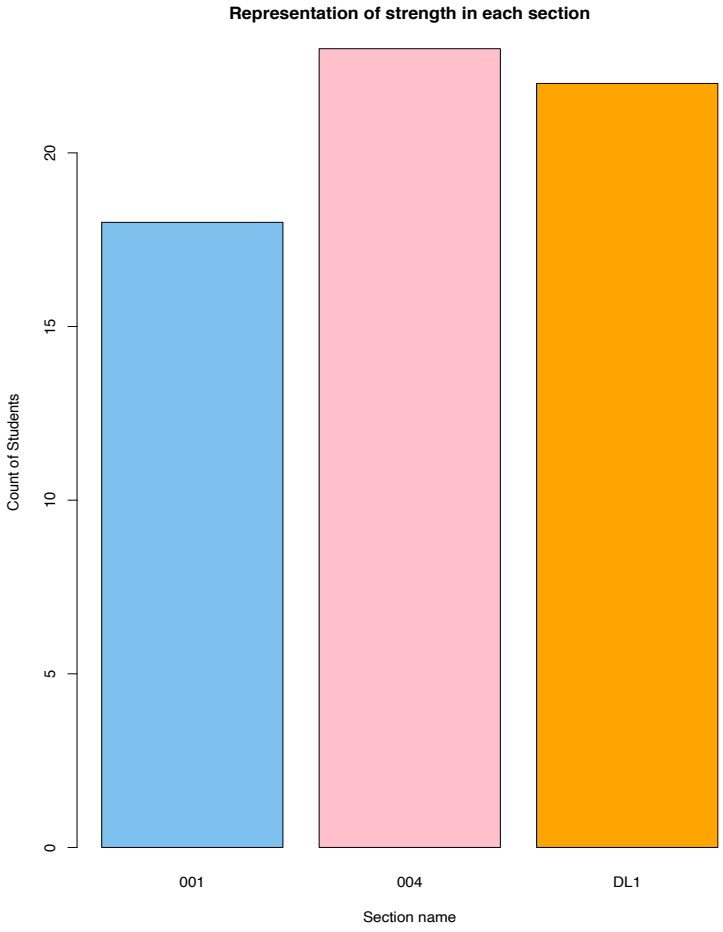
Loading dataset

```
setwd("~/Desktop/AIT580/Assignments/assignment-survey")
data=read.csv("~/AIT-580_DS.csv",header = TRUE,sep = ",")
view(data)
summary(data)
```

Section

sec=summary(data$Section)

barplot(sec,col=c("skyblue2","pink","orange"),xlab = "Section name",ylab = "Count of Students",main = "Representation of strength in each section",names.arg=c("001", "004", "DL1"))



section_summary<-summary(data$Section)

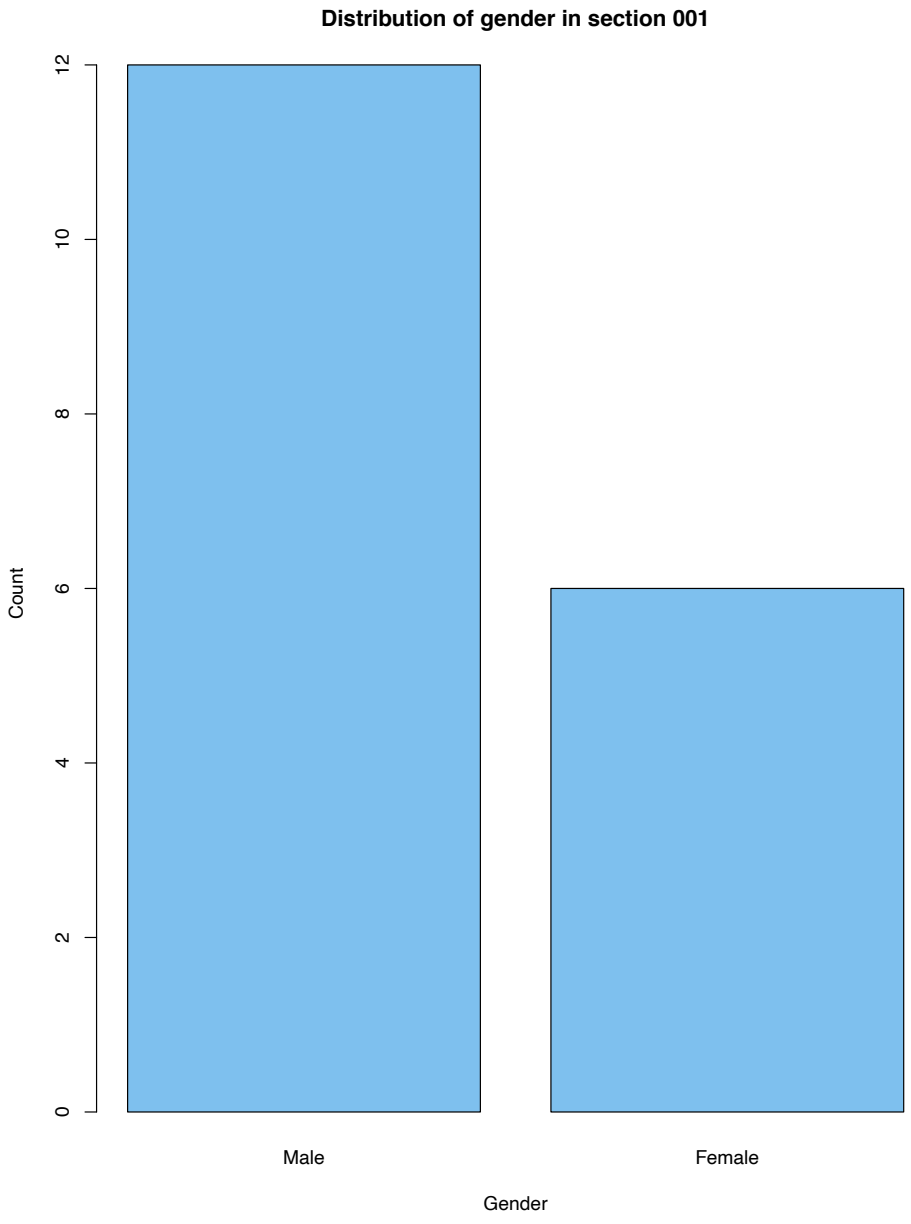
section_summary

```
> section_summary<-summary(data$Section)
> section_summary
 1  4 DL
18 23 22
```

Gender (categorized by section)

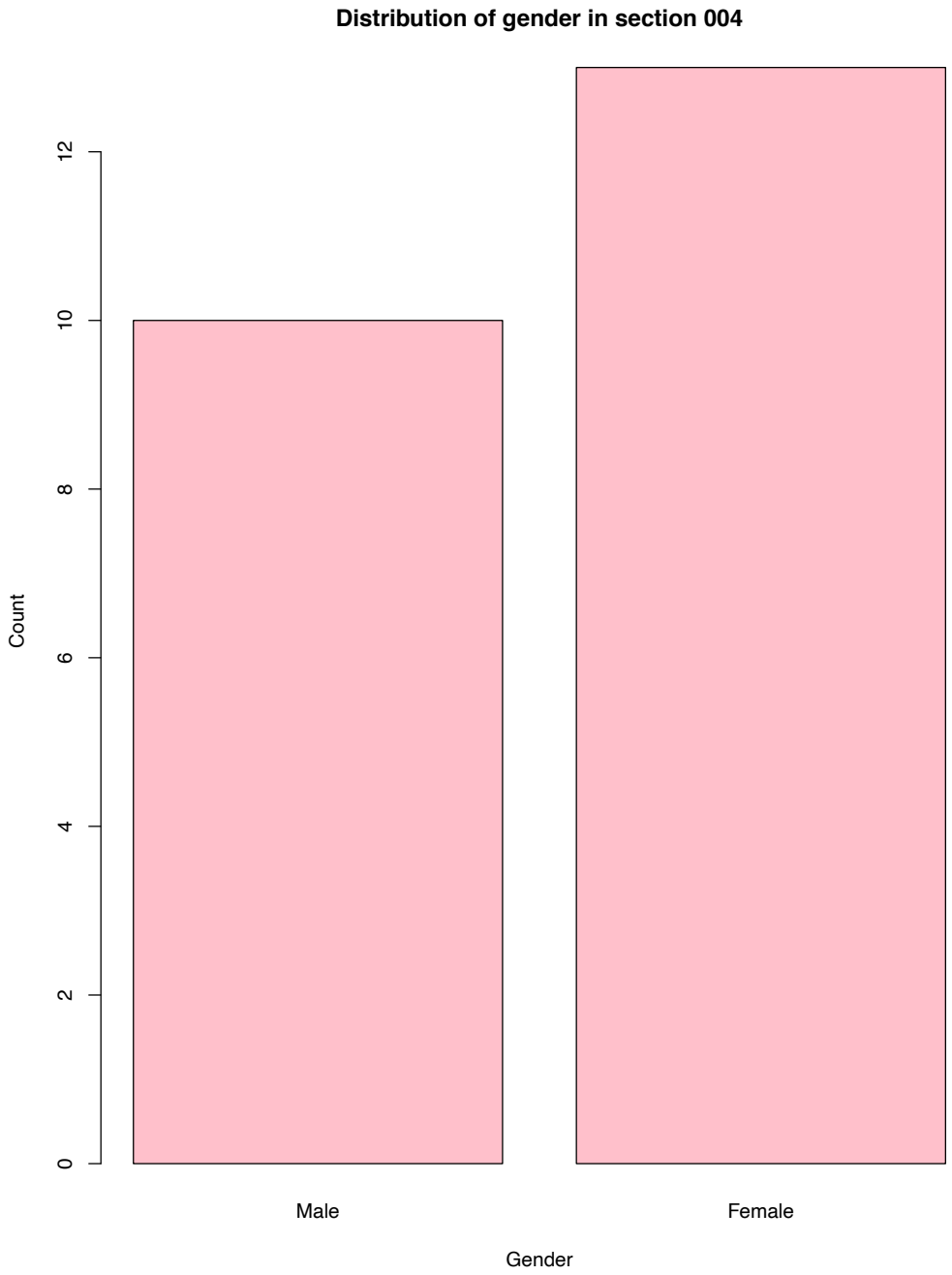
section 001

```
s1=filter(data,data$Section==1)
male1= count(filter(s1,s1$Gender=="M"))
female1=count(filter(s1,s1$Gender=="F"))
g1=unlist(c(male1,female1))
barplot(g1,names.arg=c('Male','Female'),main='Distribution of gender in section 001', xlab='Gender', ylab='Count',col="skyblue2" )
```



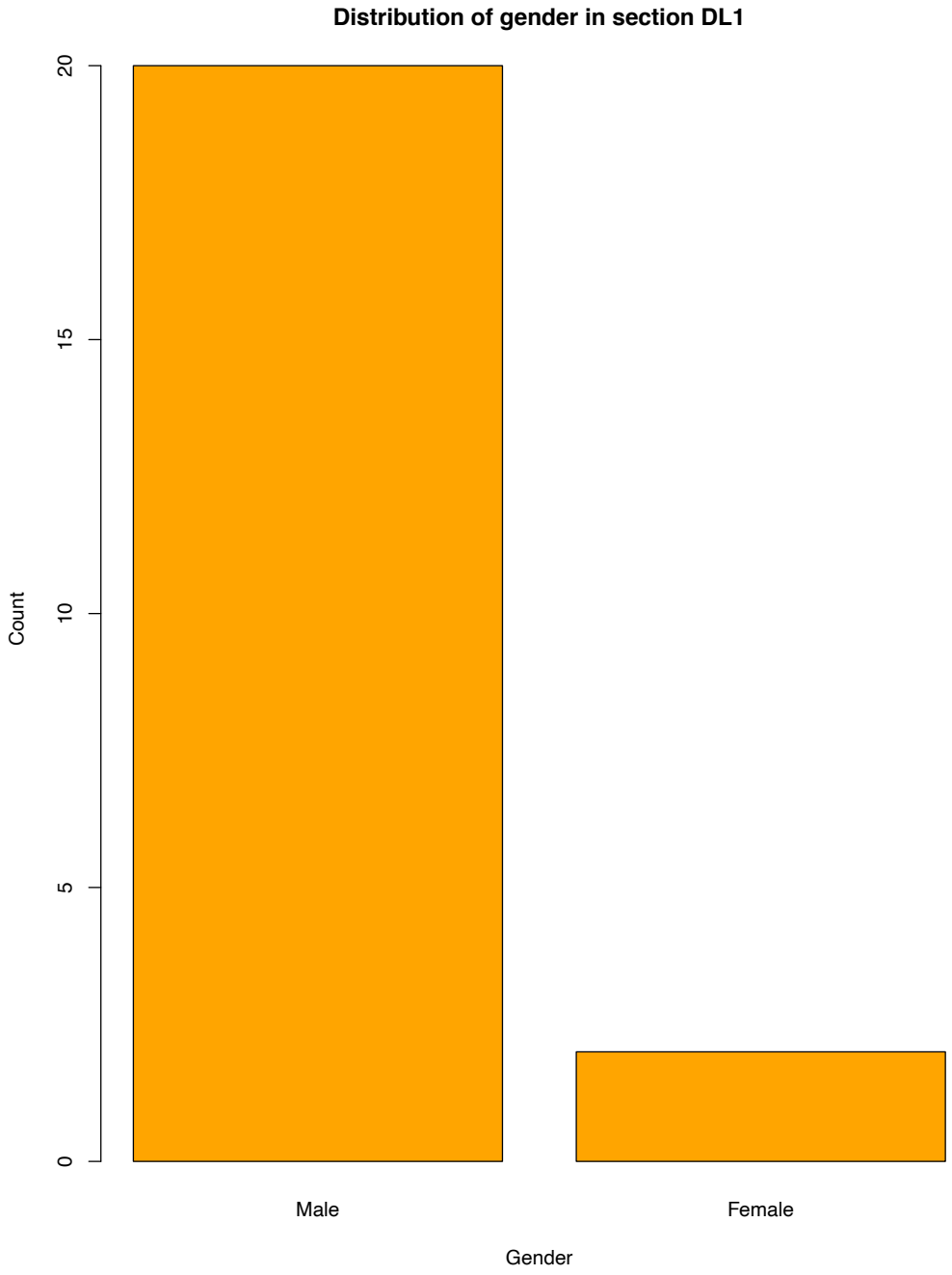
Section 004

```
s4=filter(data,data$Section==4)
male4= count(filter(s4,s4$Gender=="M"))
female4=count(filter(s4,s4$Gender=="F"))
g4=unlist(c(male4,female4))
barplot(g4,names.arg=c('Male','Female'),main='Distribution of gender in section 004', xlab='Gender', ylab='Count',col="pink" )
```



DL1 section

```
sd=filter(data,data$Section=="DL")
maled= count(filter(sd,sd$Gender=="M"))
femaled=count(filter(sd,sd$Gender=="F"))
gd=unlist(c(maled,femaled))
barplot(gd,names.arg=c('Male','Female'),main='Distribution of gender in section DL1', xlab='Gender', ylab='Count',col="orange" )
```



Gender_summary<-summary(data$Gender)

Gender_summary

```
> Gender_summary<-summary(data$Gender)
> Gender_summary
 F  M
21 42
```

Age

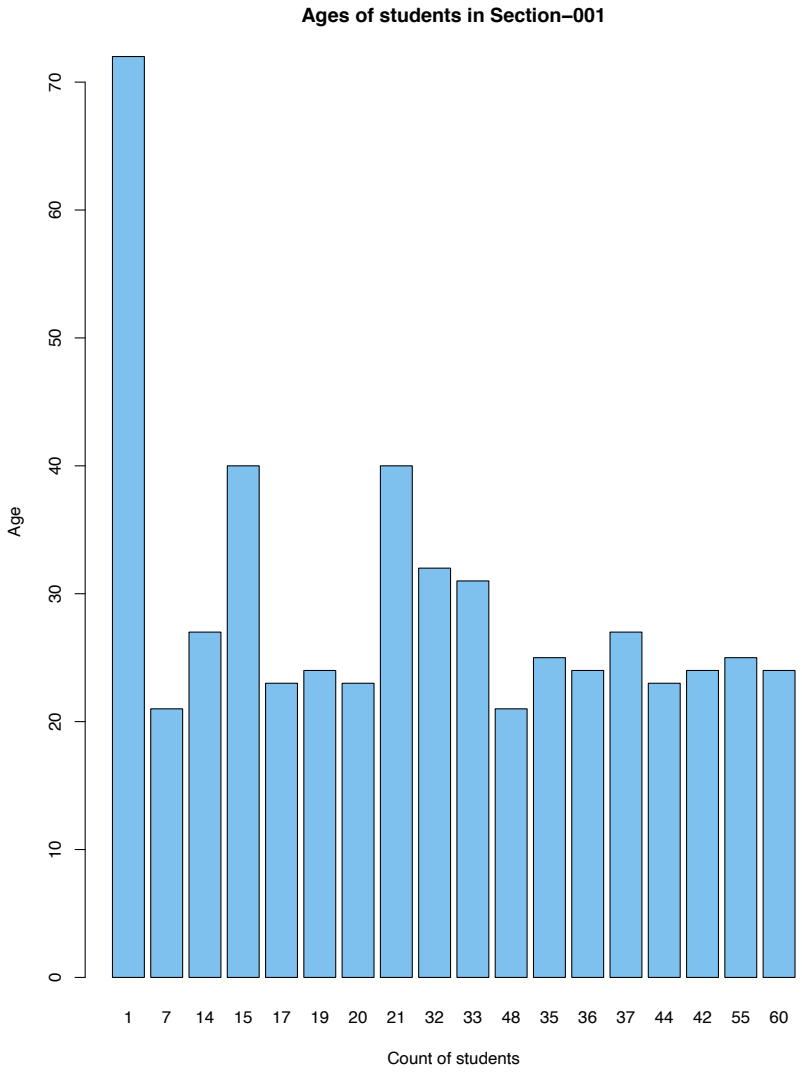
Section 001

s1=filter(data,data$Section==1)

age1=t(data.matrix(s1$Age))

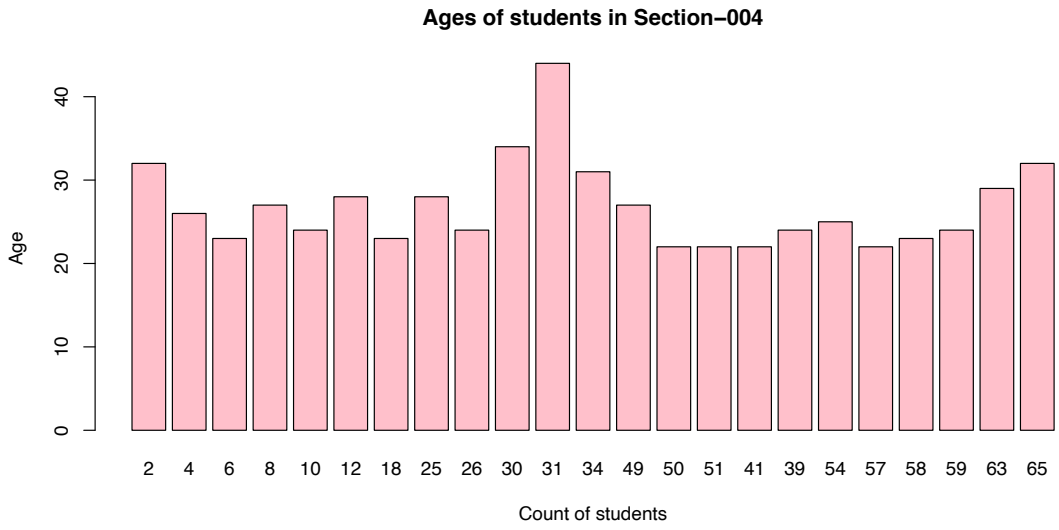
colnames(age1)=c(s1$No.)

barplot(age1,width = 2,xlab = "Count of students",ylab = "Age",main = "Ages of students in Section-001",col = "skyblue2")



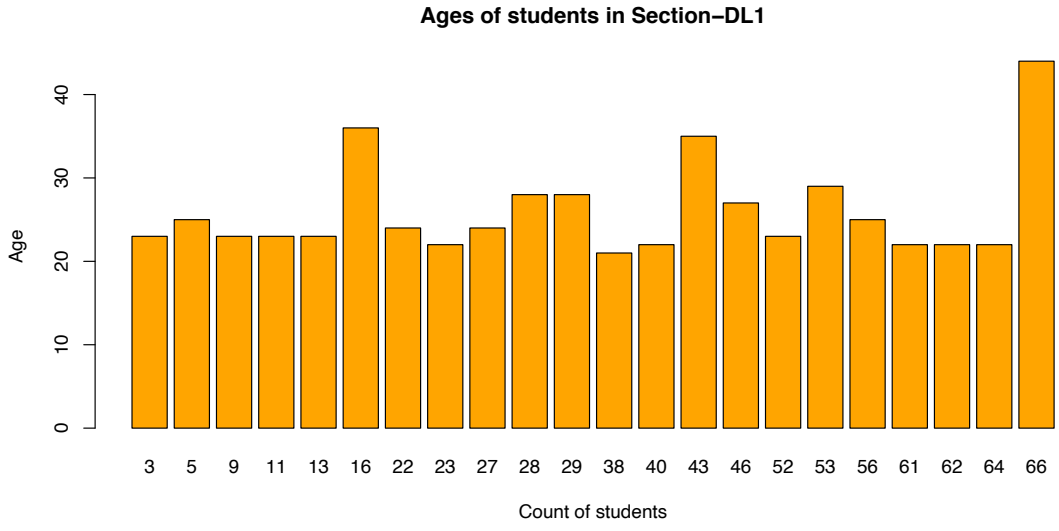
Section 004

```
s4=filter(data,data$Section=="4")
age4=t(data.matrix(s4$Age))
colnames(age4)=c(s4$No.)
barplot(age4,width = 2,xlab = "Count of students",ylab = "Age",main = "Ages of students in Section-004",col = "pink")
```



Section DL1

```
sd=filter(data,data$Section=="DL")
aged=t(data.matrix(sd$Age))
colnames(aged)=c(sd$No.)
barplot(aged,width = 2,xlab = "Count of students",ylab = "Age",main = "Ages of students in Section-DL1",col = "orange")
```



Ages_summary<-summary(data$Age)

Ages_summary

```
> Ages_summary<-summary(data$Age)
> Ages_summary
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  21.00   23.00   24.00   27.19   28.00   72.00
>
```

Height

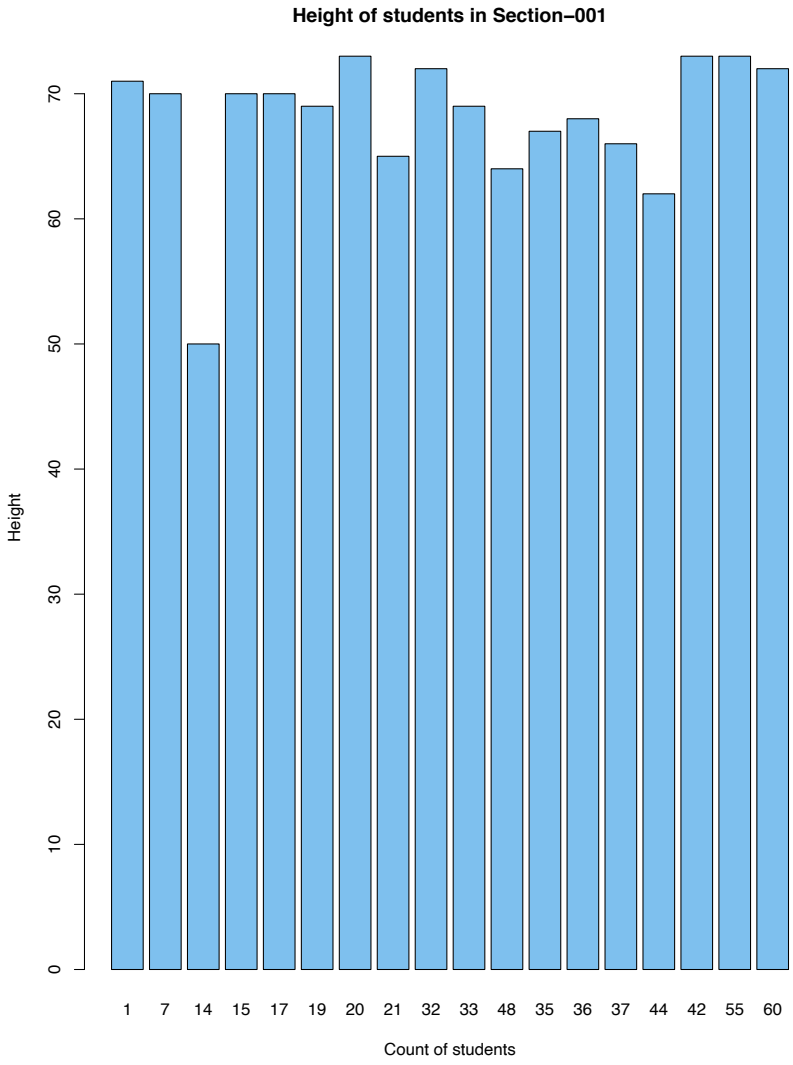
Section 001

s1=filter(data,data$Section==1)

height1=t(data.matrix(s1$Height..Inches.))

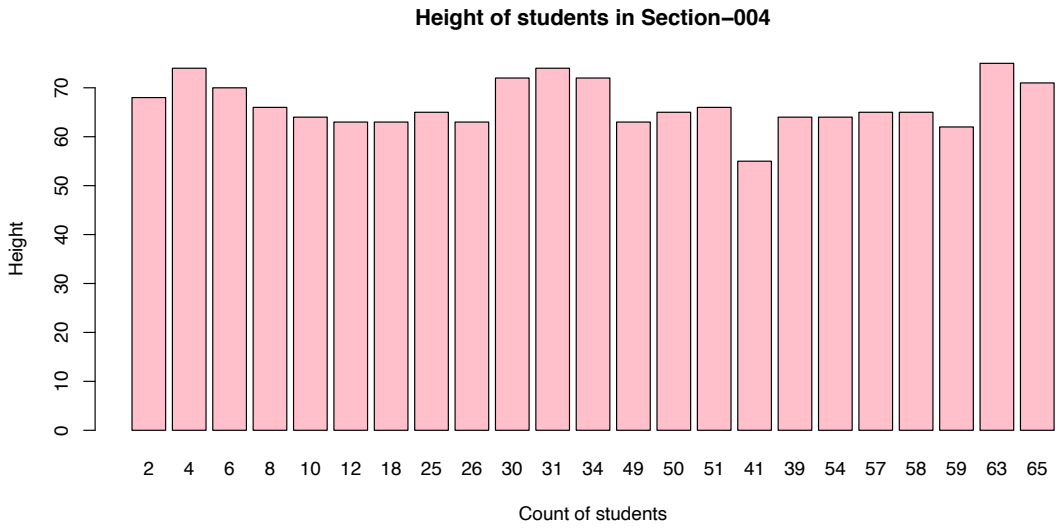
colnames(height1)=c(s1$No.)

barplot(height1,width = 2,xlab = "Count of students",ylab = "Height",main = "Height of students in Section-001",col = "skyblue2")



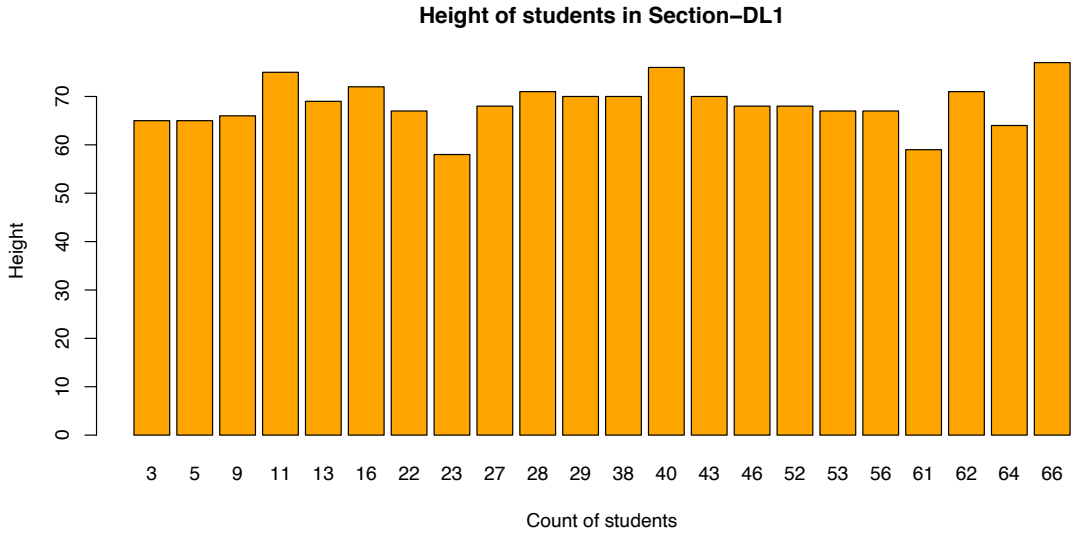
Section 004

```
s4=filter(data,data$Section==4)
height4=t(data.matrix(s4$Height..Inches.))
colnames(height4)=c(s4$No.)
barplot(height4,width = 2,xlab = "Count of students",ylab = "Height",main = "Height of students in Section-004",col = "pink")
```



Section DL1

```
sd=filter(data,data$Section=="DL")
heightd=t(data.matrix(sd$Height..Inches.))
colnames(heightd)=c(sd$No.)
barplot(heightd,width = 2,xlab = "Count of students",ylab = "Height",main = "Height of students in Section-DL1",col = "orange")
```



```
Heightininches_summary<-summary(data$Height)
Heightininches_summary
```

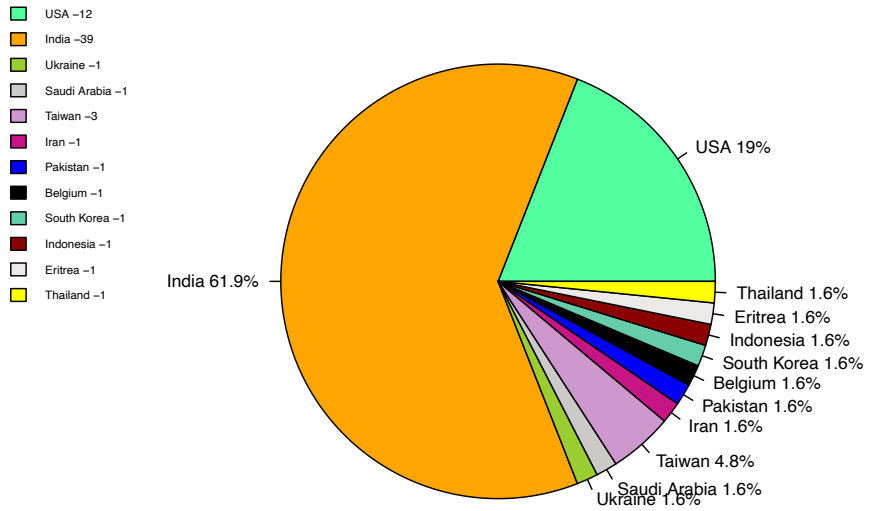
```
> Heightininches_summary<-summary(data$Height)
> Heightininches_summary
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  50.00   65.00   68.00   67.56   71.00   77.00
```

Country of citizenship

Pie chart for all sections

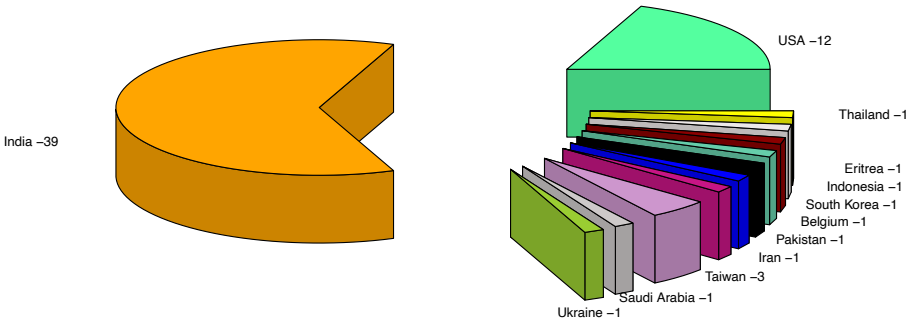
```
unique(data$Country.of.Citizenship)
usa0=count(filter(data,data$Country.of.Citizenship=="USA"))
india0=count(filter(data,data$Country.of.Citizenship=="India"))
ukraine0=count(filter(data,data$Country.of.Citizenship=="Ukraine"))
saudiarabia0=count(filter(data,data$Country.of.Citizenship=="Saudi Arabia"))
taiwan0=count(filter(data,data$Country.of.Citizenship=="Taiwan"))
iran0=count(filter(data,data$Country.of.Citizenship=="Iran"))
pakistan0=count(filter(data,data$Country.of.Citizenship=="Pakistan"))
belgium0=count(filter(data,data$Country.of.Citizenship=="Belgium"))
southkorea0=count(filter(data,data$Country.of.Citizenship=="South Korea"))
indonesia0=count(filter(data,data$Country.of.Citizenship=="Indonesia"))
eritrea0=count(filter(data,data$Country.of.Citizenship=="Eritrea"))
thailand0=count(filter(data,data$Country.of.Citizenship=="Thailand"))
countries0=unlist(c(usa0,india0,ukraine0,saudiarabia0,taiwan0,iran0,pakistan0,belgium
0,southkorea0,indonesia0,eritrea0,thailand0))
countrynames=c("USA","India","Ukraine","Saudi
Arabia","Taiwan","Iran","Pakistan","Belgium","South
Korea","Indonesia","Eritrea","Thailand")
countrylabels=round(countries0/sum(countries0) * 100, 1)
countrylabels=paste(countrylabels, "%", sep="")
countrypie=paste(countrynames,countrylabels,sep=" ")
countrylegend=paste(countrynames,- countries0,sep=" ")
colors=c("seagreen1","orange","yellowgreen","snow3","plum3","mediumvioletred","blu
e","black","aquamarine3","darkred","snow2","yellow1")
pie(countries0,labels=countrypie,explode=0.1,main="Diversity in AIT580 spring 2020 -
Total 63",cex=0.75,col = colors)+
  legend("topleft", c(countrylegend), cex=0.48,fill=colors,bty="n")
```

Diversity in AIT580 spring 2020 – Total 63



pie3D(countries0,radius=0.6,labels=countrylegend,explode=0.4,main="3D pie of AIT spring 2020 - Total 63",labelcex= 0.75,col = colors)

3D pie of AIT spring 2020 – Total 63



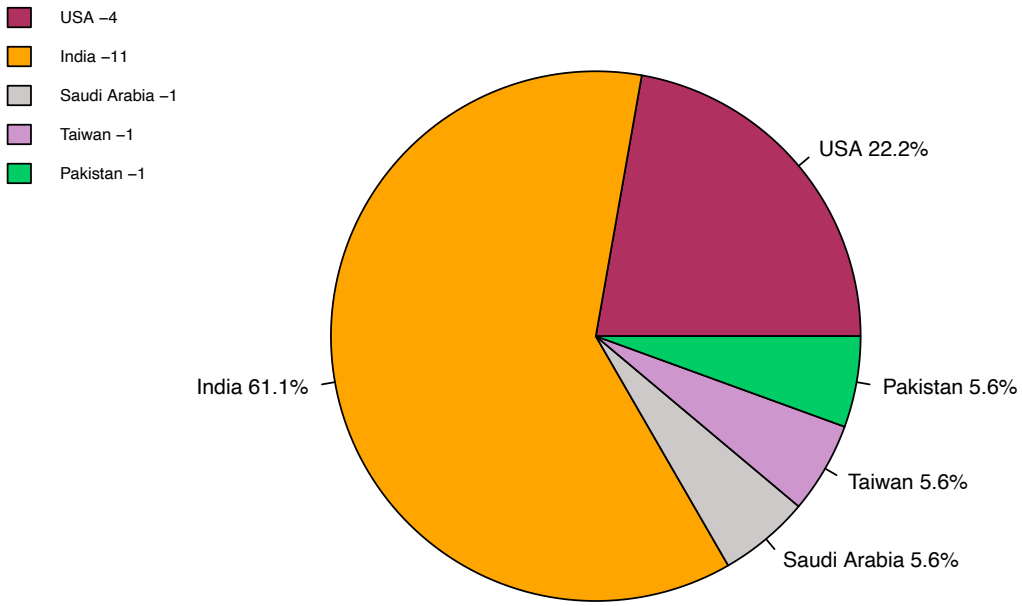
All lines of below highlighted code needs to be executed for displaying section wise piecharts

```
usa=(filter(data,data$Country.of.Citizenship=="USA"))

india=(filter(data,data$Country.of.Citizenship=="India"))

ukraine=(filter(data,data$Country.of.Citizenship=="Ukraine"))

saudiarabia=(filter(data,data$Country.of.Citizenship=="Saudi Arabia"))

taiwan=(filter(data,data$Country.of.Citizenship=="Taiwan"))

iran=(filter(data,data$Country.of.Citizenship=="Iran"))

pakistan=(filter(data,data$Country.of.Citizenship=="Pakistan"))

belgium=(filter(data,data$Country.of.Citizenship=="Belgium"))

southkorea=(filter(data,data$Country.of.Citizenship=="South Korea"))

indonesia=(filter(data,data$Country.of.Citizenship=="Indonesia"))

eritrea=(filter(data,data$Country.of.Citizenship=="Eritrea"))

thailand=(filter(data,data$Country.of.Citizenship=="Thailand"))
```

Section 001

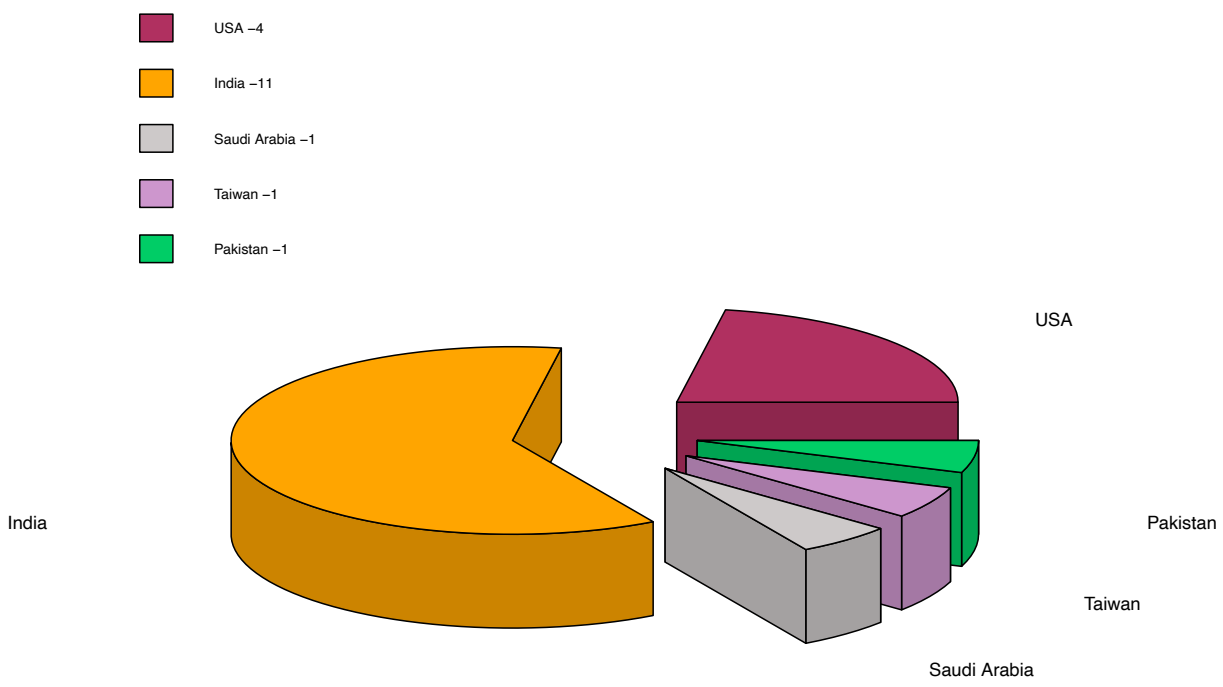
```
unique((unique(filter(data,data$Section==1)))$Country.of.Citizenship)
usa1=count(filter(usa,usa$Section==1))
india1=count(filter(india,india$Section==1))
saudiarabia1=count(filter(saudiarabia,saudiarabia$Section==1))
taiwan1=count(filter(taiwan,taiwan$Section==1))
pakistan1=count(filter(pakistan,pakistan$Section==1))
countries1=unlist(c(usa1,india1,saudiarabia1,taiwan1,pakistan1))
countrynames1=c("USA","India","Saudi Arabia","Taiwan","Pakistan")
countrylabels1=round(countries1/sum(countries1) * 100, 1)
countrylabels1=paste(countrylabels1, "%", sep="")
countrypie1=paste(countrynames1,countrylabels1,sep=" ")
countrylegend1=paste(countrynames1,- countries1,sep=" ")
colors1=c("maroon","orange","snow3","plum3","springgreen3")
```

```
pie(countries1,labels=countrypie1,explode=0.1,main="Diversity in AIT580 spring 2020 Section-001 (Total- 18) ",cex=0.75,col = colors1)+
  legend("topleft", c(countrylegend1), cex=0.6,fill=colors1,bty="n")
```



pie3D(countries1,radius=0.6,labels=countrynames1,explode=0.2,main="3D pie of AIT spring 2020 Section-001 (Total- 18)",labelcex= 0.75,col = colors1)+

legend("topleft", c(countrylegend1), cex=0.6,fill=colors1,bty="n")



Section 004

unique((unique(filter(data,data$Section==4)))$Country.of.Citizenship)

usa4=count(filter(usa,usa$Section==4))

india4=count(filter(india,india$Section==4))

ukraine4=count(filter(ukraine,ukraine$Section==4))

iran4=count(filter(iran,iran$Section==4))

belgium4=count(filter(belgium,belgium$Section==4))

eritrea4=count(filter(eritrea,eritrea$Section==4))

countries4=unlist(c(usa4,india4,ukraine4,iran4,belgium4,eritrea4))

countrynames4=c("USA","India","Ukraine","Iran","Belgium","Eritrea")

countrylabels4=round(countries4/sum(countries4) * 100, 1)

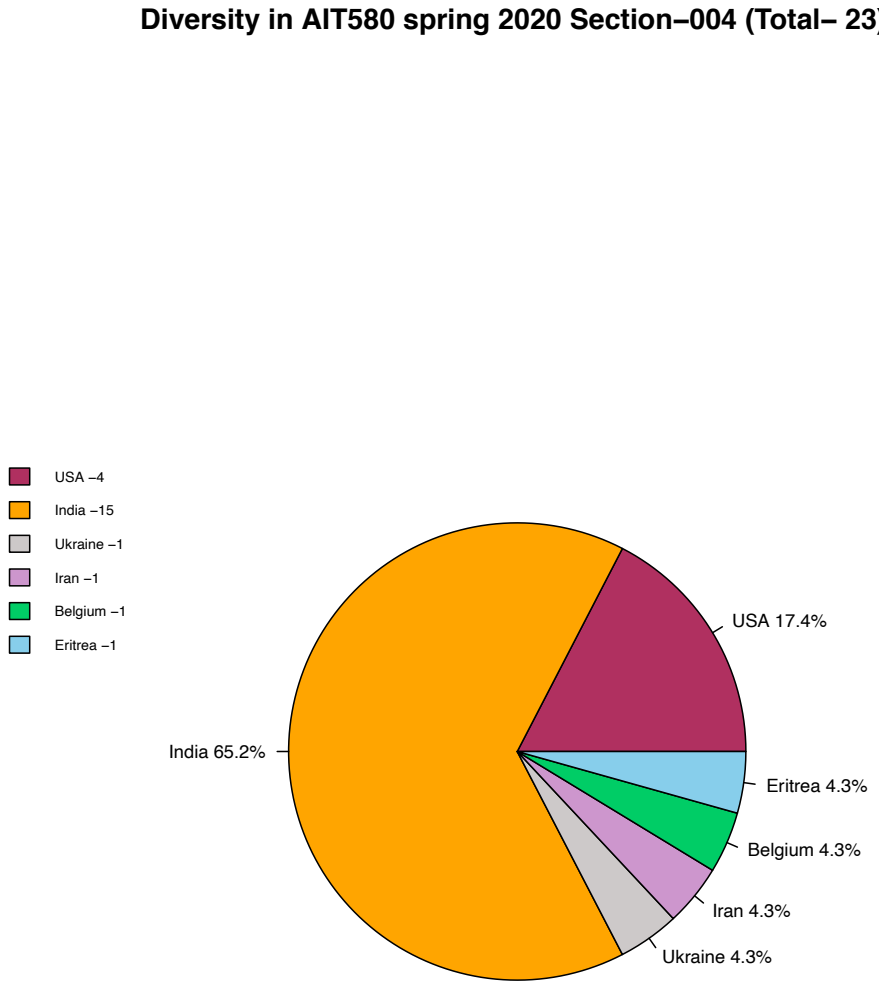
countrylabels4=paste(countrylabels4, "%", sep="")

countrypie4=paste(countrynames4,countrylabels4,sep=" ")

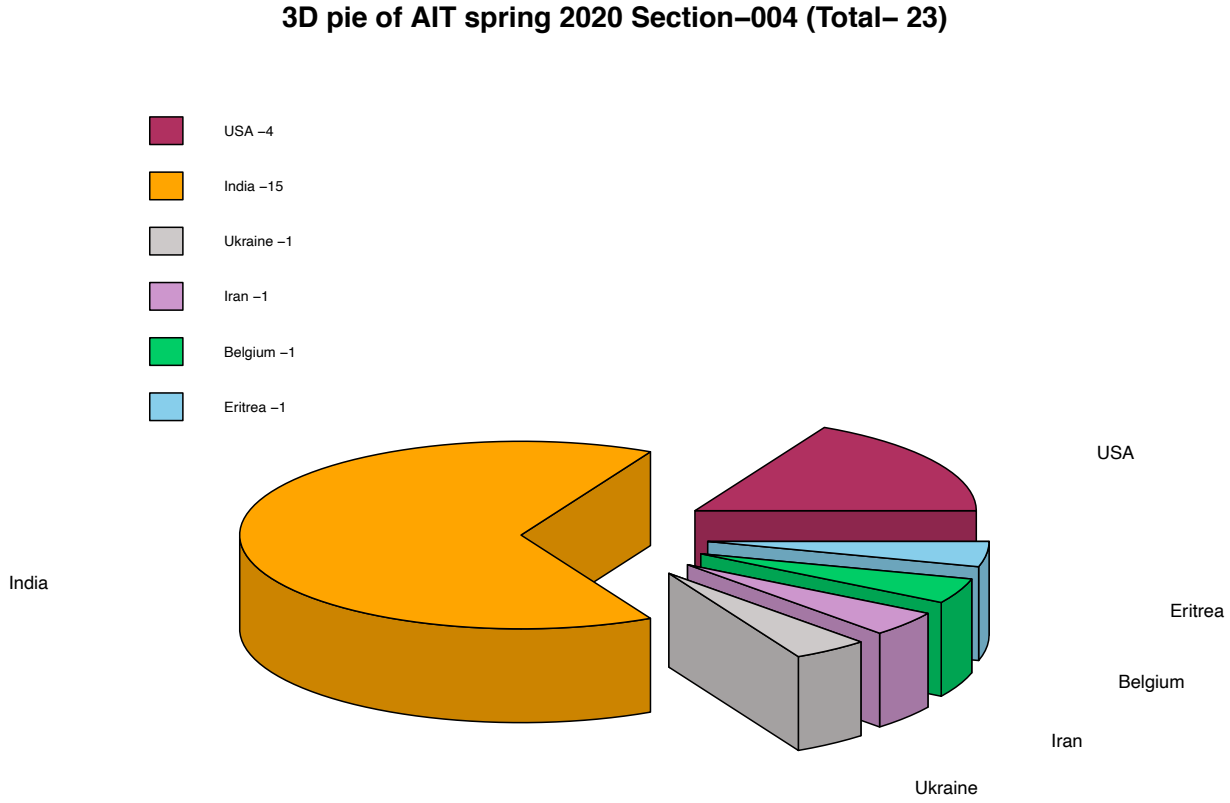
countrylegend4=paste(countrynames4,- countries4,sep=" ")

colors4=c("maroon","orange","snow3","plum3","springgreen3","skyblue")

```
pie(countries4,labels=countrypie4,explode=0.1,main="Diversity in AIT580 spring 2020 Section-001 (Total- 23) ",cex=0.75,col = colors4)+
  legend("topleft", c(countrylegend4), cex=0.6,fill=colors4,bty="n")
```



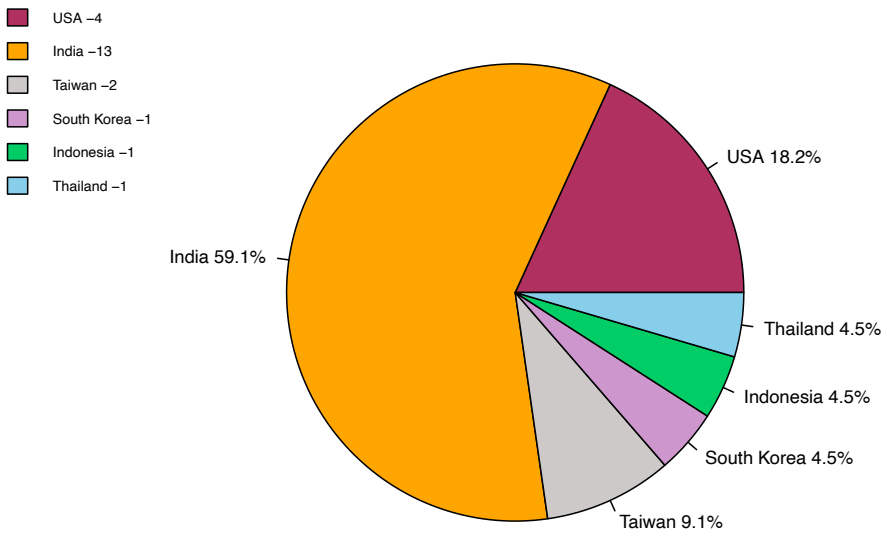
pie3D(countries4,radius=0.6,labels=countrynames4,explode=0.2,main="3D pie of AIT spring 2020 Section-001 (Total- 23)",labelcex= 0.75,col = colors4)+
legend("topleft", c(countrylegend4), cex=0.6,fill=colors4,bty="n")



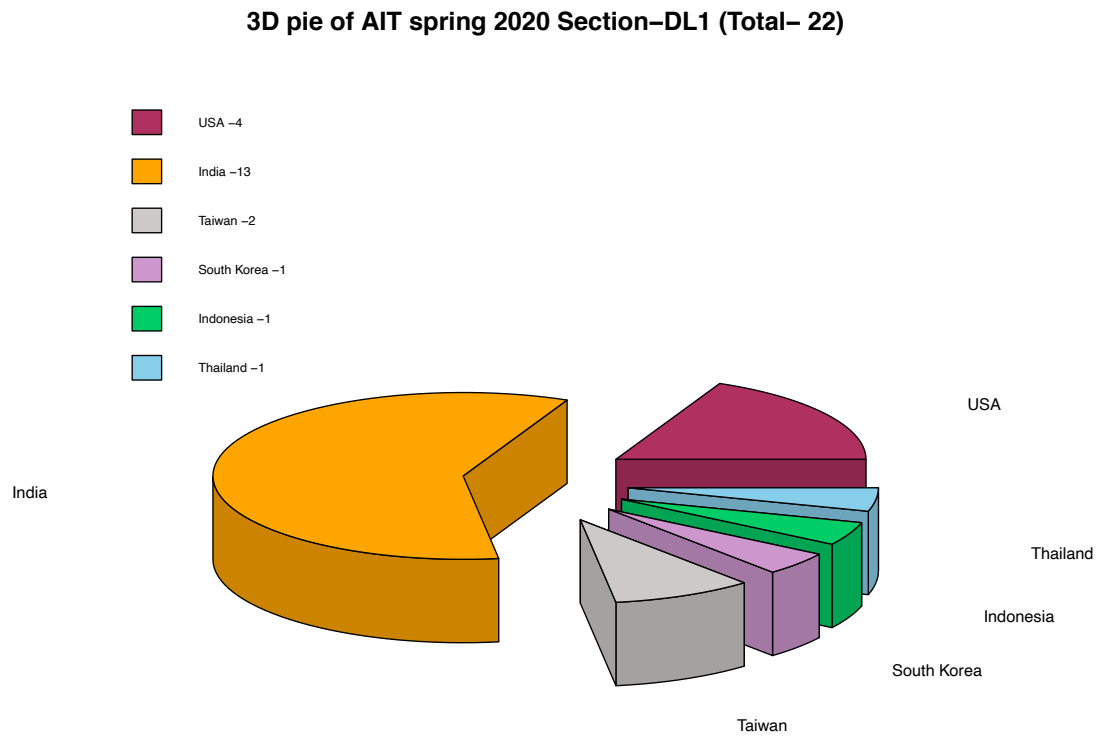
Section DL

unique((unique(filter(data,data$Section=="DL")))$Country.of.Citizenship)
usadl=count(filter(usa,usa$Section=="DL"))
indiadl=count(filter(india,india$Section=="DL"))
taiwandl=count(filter(taiwan,taiwan$Section=="DL"))
southkoreadl=count(filter(southkorea,southkorea$Section=="DL"))
indonesiadl=count(filter(indonesia,indonesia$Section=="DL"))
thailanddl=count(filter(thailand,thailand$Section=="DL"))
countriesdl=unlist(c(usadl,indiadl,taiwandl,southkoreadl,indonesiadl,thailanddl))
countrynamesdl=c("USA","India","Taiwan","South Korea","Indonesia","Thailand")
countrylabelsdl=round(countriesdl/sum(countriesdl) * 100, 1)
countrylabelsdl=paste(countrylabelsdl, "%", sep="")
countrypiedl=paste(countrynamesdl,countrylabelsdl,sep=" ")
countrylegenddl=paste(countrynamesdl,- countriesdl,sep=" ")
colorsdl=c("maroon","orange","snow3","plum3","springgreen3","skyblue")

```
pie(countriesdl,labels=countrypiedl,explode=0.1,main="Diversity in AIT580 spring 2020 Section-001 (Total- 22) ",cex=0.75,col = colorsdl)+
  legend("topleft", c(countrylegenddl), cex=0.6,fill=colorsdl,bty="n")
```



pie3D(countriesdl,radius=0.6,labels=countrynamesdl,explode=0.2,main="3D pie of AIT spring 2020 Section-001 (Total- 22)",labelcex= 0.75,col = colorsdl)+
legend("topleft", c(countrylegenddl), cex=0.6,fill=colorsdl,bty="n")



CountryofCitizenship_summary<-summary(data$Country.of.Citizenship)
section_Countryofcitizenship

```
> CountryofCitizenship_summary<-summary(data$Country.of.Citizenship)
> CountryofCitizenship_summary
     Belgium      Eritrea        India    Indonesia         Iran     Pakistan
           1            1           39            1            1            1
Saudi Arabia  South Korea       Taiwan     Thailand      Ukraine          USA
           1            1            3            1            1           12
```

Undergraduate degree

All lines of below highlighted code needs to be executed for displaying section wise piecharts

computerscience=(filter(data,data$Undergraduate.Degree=="Computer Science"))

intelops=(filter(data,data$Undergraduate.Degree=="Intelligence Operations"))

others=(filter(data,data$Undergraduate.Degree=="Bachelor's in other courses"))

electronics=(filter(data,data$Undergraduate.Degree=="Electronics"))

mathematics=(filter(data,data$Undergraduate.Degree=="Mathematics"))

```
is=(filter(data,data$Undergraduate.Degree=="Information Systems"))
surveyeng=(filter(data,data$Undergraduate.Degree=="Surveying Engineering"))
business=(filter(data,data$Undergraduate.Degree=="Business"))
physics=(filter(data,data$Undergraduate.Degree=="Physics"))
elecmedia=(filter(data,data$Undergraduate.Degree=="Electronics and Media"))
archeng=(filter(data,data$Undergraduate.Degree=="Architectural engineering"))
compapplication=(filter(data,data$Undergraduate.Degree=="Computer Application"))
managementscience=(filter(data,data$Undergraduate.Degree=="Management
Science"))
pgraduate=(filter(data,data$Undergraduate.Degree=="Pathway graduate"))
civil=(filter(data,data$Undergraduate.Degree=="Civil engineering"))
mathgerman=(filter(data,data$Undergraduate.Degree=="Mathematics and German"))
cybersec=(filter(data,data$Undergraduate.Degree=="AIT - Cyber Security"))
ba=(filter(data,data$Undergraduate.Degree=="Business Administration"))
eleccomm=(filter(data,data$Undergraduate.Degree=="Electronics and
Communication"))
it=(filter(data,data$Undergraduate.Degree=="Information Technology"))
as=(filter(data,data$Undergraduate.Degree=="Applied statistics"))
mech=(filter(data,data$Undergraduate.Degree=="Mechanical Engineering"))
biology=(filter(data,data$Undergraduate.Degree=="Biology"))
indeng=(filter(data,data$Undergraduate.Degree=="Industrial Engineering"))
eleceng=(filter(data,data$Undergraduate.Degree=="Electrical Engineering"))
politicals=(filter(data,data$Undergraduate.Degree=="Political Science"))
```

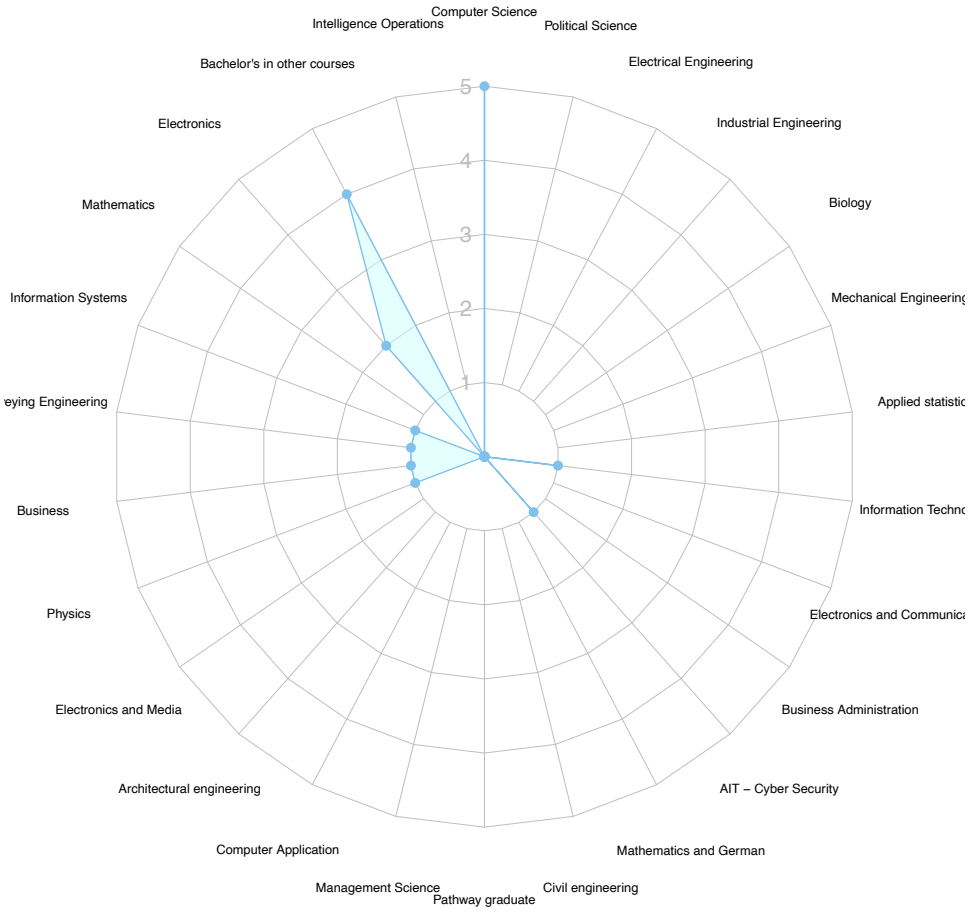
Section 001

```
unique((unique(filter(data,data$Section==1)))$Undergraduate.Degree)
computerscience1=count(filter(computerscience,computerscience$Section==1))
intelops1=count(filter(intelops,intelops$Section==1))
others1=count(filter(others,others$Section==1))
electronics1=count(filter(electronics,electronics$Section==1))
mathematics1=count(filter(mathematics,mathematics$Section==1))
```

```
is1=count(filter(is,is$Section==1))
surveyeng1=count(filter(surveyeng,surveyeng$Section==1))
business1=count(filter(business,business$Section==1))
physics1=count(filter(physics,physics$Section==1))
elecmedia1=count(filter(elecmedia,elecmedia$Section==1))
archeng1=count(filter(archeng,archeng$Section==1))
compapplication1=count(filter(compapplication,compapplication$Section==1))
managementscience1=count(filter(managementscience,managementscience$Section==
1))
pgraduate1=count(filter(pgraduate,pgraduate$Section==1))
civil1=count(filter(civil,civil$Section==1))
mathgerman1=count(filter(mathgerman,mathgerman$Section==1))
cybersec1=count(filter(cybersec,cybersec$Section==1))
ba1=count(filter(ba,ba$Section==1))
eleccomm1=count(filter(eleccomm,eleccomm$Section==1))
it1=count(filter(it,it$Section==1))
as1=count(filter(as,as$Section==1))
mech1=count(filter(mech,mech$Section==1))
biology1=count(filter(biology,biology$Section==1))
indeng1=count(filter(indeng,indeng$Section==1))
eleceng1=count(filter(eleceng,eleceng$Section==1))
politicals1=count(filter(politicals,politicals$Section==1))
degree1=as.data.frame(c(computerscience1,intelops1,others1,electronics1,mathematic
s1,is1,surveyeng1,business1,physics1,elecmedia1,archeng1,compapplication1,managem
entscience1,pgraduate1,civil1,mathgerman1,cybersec1,ba1,eleccomm1,it1,as1,mech1,b
iology1,indeng1,eleceng1,politicals1))
a=t(matrix(unlist(degree1)))
a=as.data.frame(a)
colnames(a)=c("Computer Science","Intelligence Operations","Bachelor's in other
courses","Electronics","Mathematics","Information Systems","Surveying
Engineering","Business","Physics","Electronics and Media","Architectural
engineering","Computer Application","Management Science","Pathway graduate","Civil
engineering","Mathematics and German","AIT - Cyber Security","Business
Administration","Electronics and Communication","Information Technology","Applied
statistics","Mechanical Engineering","Biology","Industrial Engineering","Electrical
Engineering","Political Science")
rownames(a)=("Section - 001")
a=rbind(rep(5,26),rep(1,26),a)
```

```
radarchart( a  , axistype=1,
        pcol="skyblue2" , pfcol=rgb(0.8,1,1,0.5) , plwd=1 ,
        cglcol="grey", cglty=1.1, axislabcol="grey", caxislabels=c(1:5), cglwd=0.4,
        vlcex=0.55,title=paste("Undergraduate degrees of students in AIT580 spring 2020: Section-001")
)
```

Undergraduate degrees of students in AIT580 spring 2020: Section–001

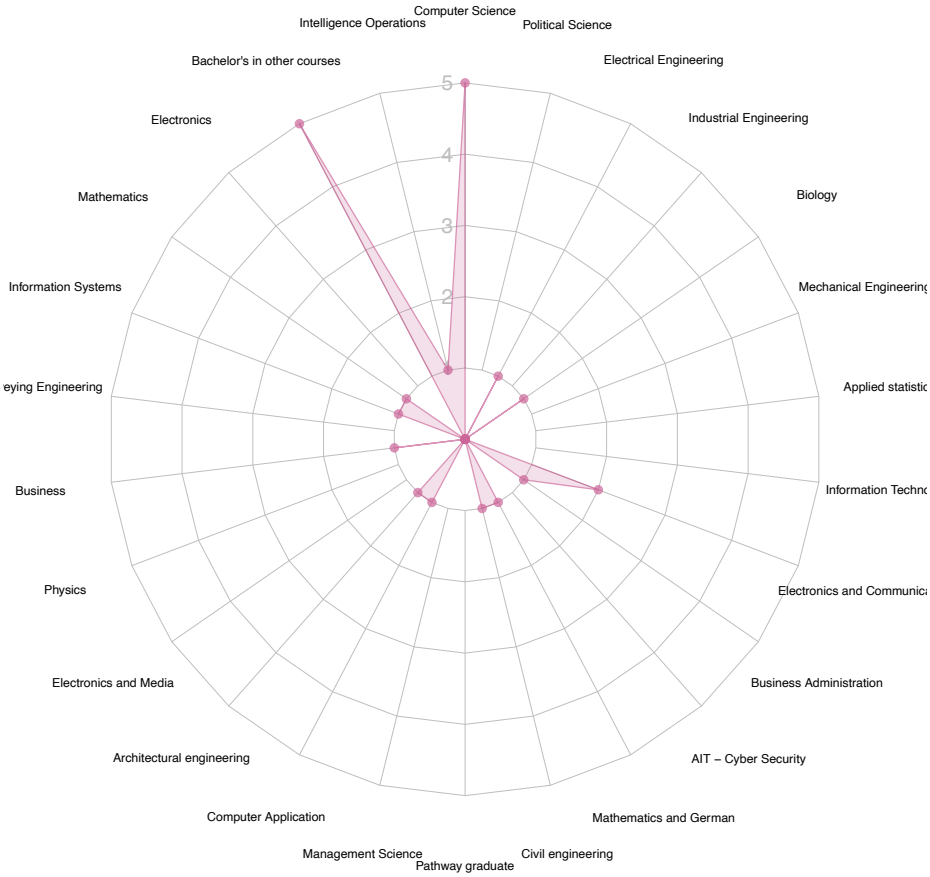


Section 004

```
unique((unique(filter(data,data$Section==4)))$Undergraduate.Degree)
computerscience4=count(filter(computerscience,computerscience$Section==4))
intelops4=count(filter(intelops,intelops$Section==4))
others4=count(filter(others,others$Section==4))
electronics4=count(filter(electronics,electronics$Section==4))
mathematics4=count(filter(mathematics,mathematics$Section==4))
is4=count(filter(is,is$Section==4))
surveyeng4=count(filter(surveyeng,surveyeng$Section==4))
business4=count(filter(business,business$Section==4))
physics4=count(filter(physics,physics$Section==4))
elecmedia4=count(filter(elecmedia,elecmedia$Section==4))
archeng4=count(filter(archeng,archeng$Section==4))
compapplication4=count(filter(compapplication,compapplication$Section==4))
managementscience4=count(filter(managementscience,managementscience$Section==
4))
pgraduate4=count(filter(pgraduate,pgraduate$Section==4))
civil4=count(filter(civil,civil$Section==4))
mathgerman4=count(filter(mathgerman,mathgerman$Section==4))
cybersec4=count(filter(cybersec,cybersec$Section==4))
ba4=count(filter(ba,ba$Section==4))
eleccomm4=count(filter(eleccomm,eleccomm$Section==4))
it4=count(filter(it,it$Section==4))
as4=count(filter(as,as$Section==4))
mech4=count(filter(mech,mech$Section==4))
biology4=count(filter(biology,biology$Section==4))
indeng4=count(filter(indeng,indeng$Section==4))
eleceng4=count(filter(eleceng,eleceng$Section==4))
politicals4=count(filter(politicals,politicals$Section==4))
degree4=as.data.frame(c(computerscience4,intelops4,others4,electronics4,mathematic
s4,is4,surveyeng4,business4,physics4,elecmedia4,archeng4,compapplication4,managem
entscience4,pgraduate4,civil4,mathgerman4,cybersec4,ba4,eleccomm4,it4,as4,mech4,b
iology4,indeng4,eleceng4,politicals4))
b=t(matrix(unlist(degree4)))
b=as.data.frame(b)
colnames(b)=c("Computer Science","Intelligence Operations","Bachelor's in other
courses","Electronics","Mathematics","Information Systems","Surveying
Engineering","Business","Physics","Electronics and Media","Architectural
```

```
engineering","Computer Application","Management Science","Pathway graduate","Civil engineering","Mathematics and German","AIT - Cyber Security","Business Administration","Electronics and Communication","Information Technology","Applied statistics","Mechanical Engineering","Biology","Industrial Engineering","Electrical Engineering","Political Science")
rownames(b)=("Section - 004")
b=rbind(rep(5,26),rep(1,26),b)

radarchart( b  , axistype=1,
        pcol=rgb(0.8,0.4,0.6,0.7) , pfcol=rgb(0.8,0.4,0.6,0.2) , plwd=1 ,
        cglcol="grey", cglty=1.1, axislabcol="grey", caxislabels=c(1:5), cglwd=0.4,
        vlcex=0.55,title=paste("Undergraduate degrees of students in AIT580 spring 2020: Section-004")
)
```

Undergraduate degrees of students in AIT580 spring 2020: Section–004

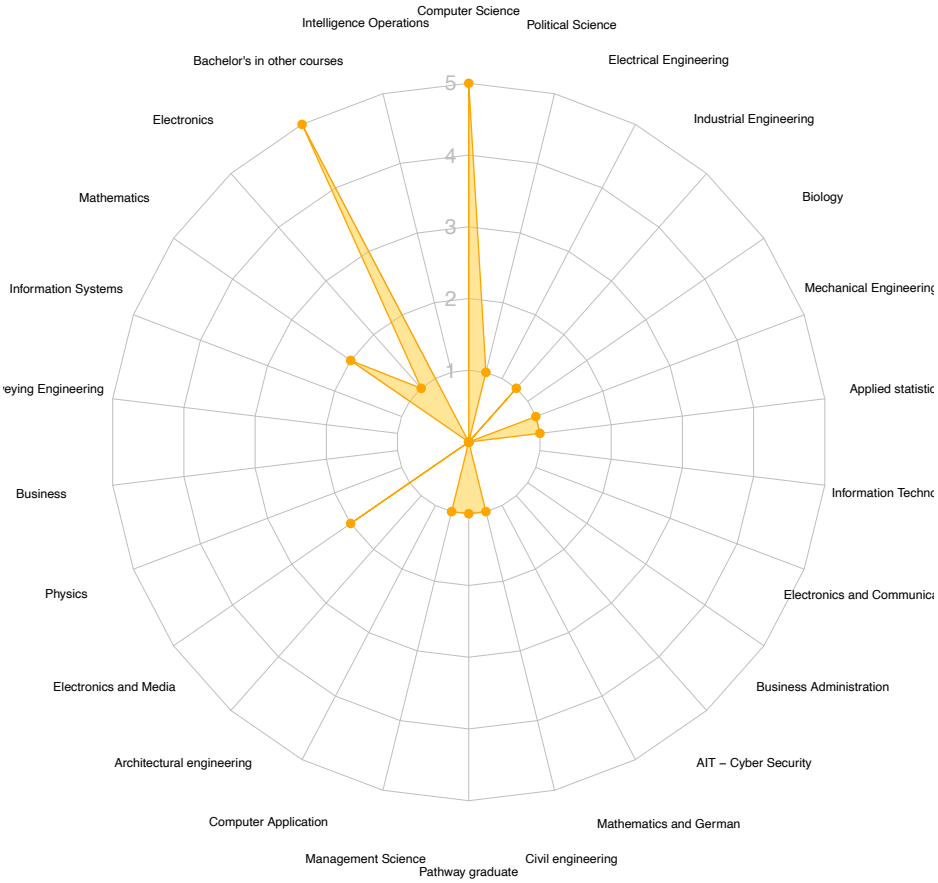


Section DL1

```
unique((unique(filter(data,data$Section=="DL")))$Undergraduate.Degree)
computerscienced=count(filter(computerscience,computerscience$Section=="DL"))
intelopsd=count(filter(intelops,intelops$Section=="DL"))
othersd=count(filter(others,others$Section=="DL"))
electronicsd=count(filter(electronics,electronics$Section=="DL"))
mathematicsd=count(filter(mathematics,mathematics$Section=="DL"))
isd=count(filter(is,is$Section=="DL"))
surveyengd=count(filter(surveyeng,surveyeng$Section=="DL"))
businessd=count(filter(business,business$Section=="DL"))
physicsd=count(filter(physics,physics$Section=="DL"))
elecmediad=count(filter(elecmedia,elecmedia$Section=="DL"))
archengd=count(filter(archeng,archeng$Section=="DL"))
compapplicationd=count(filter(compapplication,compapplication$Section=="DL"))
managementscienced=count(filter(managementscience,managementscience$Section==
"DL"))
pgraduated=count(filter(pgraduate,pgraduate$Section=="DL"))
civild=count(filter(civil,civil$Section=="DL"))
mathgermand=count(filter(mathgerman,mathgerman$Section=="DL"))
cybersecd=count(filter(cybersec,cybersec$Section=="DL"))
bad=count(filter(ba,ba$Section=="DL"))
eleccommd=count(filter(eleccomm,eleccomm$Section=="DL"))
itd=count(filter(it,it$Section=="DL"))
asd=count(filter(as,as$Section=="DL"))
mechd=count(filter(mech,mech$Section=="DL"))
biologyd=count(filter(biology,biology$Section=="DL"))
indengd=count(filter(indeng,indeng$Section=="DL"))
elecengd=count(filter(eleceng,eleceng$Section=="DL"))
politicalsd=count(filter(politicals,politicals$Section=="DL"))
degreed=as.data.frame(c(computerscienced,intelopsd,othersd,electronicsd,mathematic
sd,isd,surveyengd,businessd,physicsd,elecmediad,archengd,compapplicationd,manage
mentscienced,pgraduated,civild,mathgermand,cybersecd,bad,eleccommd,itd,asd,mech
d,biologyd,indengd,elecengd,politicalsd))
c=t(matrix(unlist(degreed)))
c=as.data.frame(c)
colnames(c)=c("Computer Science","Intelligence Operations","Bachelor's in other
courses","Electronics","Mathematics","Information Systems","Surveying
Engineering","Business","Physics","Electronics and Media","Architectural
```

```
engineering","Computer Application","Management Science","Pathway graduate","Civil engineering","Mathematics and German","AIT - Cyber Security","Business Administration","Electronics and Communication","Information Technology","Applied statistics","Mechanical Engineering","Biology","Industrial Engineering","Electrical Engineering","Political Science")
rownames(c)=("Section - DL1")
c=rbind(rep(5,26),rep(1,26),c)

radarchart( c  , axistype=1,
        pcol="orange" , pfcol=rgb(1,0.8,0.2,0.5) , plwd=1 ,
        cglcol="grey", cglty=1.1, axislabcol="grey", caxislabels=c(1:5), cglwd=0.4,
        vlcex=0.55,title=paste("Undergraduate degrees of students in AIT580 spring 2020: Section-DL1")
)
```

Undergraduate degrees of students in AIT580 spring 2020: Section–DL1



UndergraduateDegree_summary<-summary(data$Undergraduate.Degree)
UndergraduateDegree_summary

```
> UndergraduateDegree_summary<-summary(data$Undergraduate.Degree)
> UndergraduateDegree_summary
                                    AIT - Cyber Security
                              1                         1
           Applied statistics     Architectural engineering
                              1                         1
  Bachelor's in other courses                   Biology
                             14                         1
                       Business       Business Administration
                              2                         1
            Civil engineering          Computer Application
                              2                         1
                    Electronics Electronics and Communication
                              3                         2
         Electronics and Media        Industrial Engineering
                              2                         1
           Information Systems        Information Technology
                              2                         1
       Intelligence Operations            Management Science
                              1                         1
                    Mathematics        Mathematics and German
                              3                         1

      Intelligence Operations            Management Science
                              1                         1
                    Mathematics        Mathematics and German
                              3                         1
       Mechanical Engineering              Pathway graduate
                              1                         1
                      Physics             Political Science
                              1                         1
        Surveying Engineering
                              1
```

Expected graduation date

Section 001

ed1<-data$Expected.Graduation.date[data$Section=="1"]

ed1

ed2<-data$Expected.Graduation.date[data$Section=="4"]

ed2

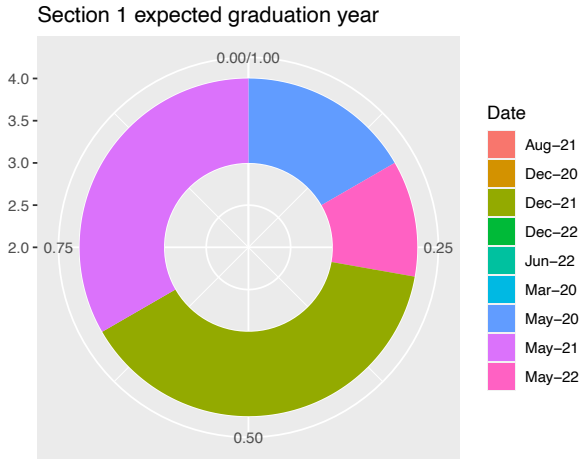
ed3<-data$Expected.Graduation.date[data$Section=="DL"]

ed3

#Plot for section 1

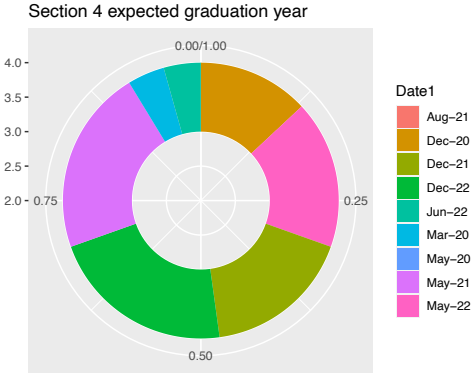
coun<-ed1[ed1=="May-20"]

```
length(coun)
coun1<-ed1[ed1=="Dec-20"]
length(coun1)
coun2<-ed1[ed1=="May-22"]
length(coun2)
coun3<-ed1[ed1=="Dec-21"]
length(coun3)
coun4<-ed1[ed1=="Dec-22"]
length(coun4)
coun5<-ed1[ed1=="May-21"]
length(coun5)
coun6<-ed1[ed1=="Mar-20"]
length(coun6)
coun7<-ed1[ed1=="Aug-21"]
length(coun7)
coun8<-ed1[ed1=="Jun-22"]
length(coun8)
df <- data.frame(Date=c("May-20","Dec-20","May-22","Dec-21","Dec-22","May-21","Mar-20","Aug-21","Jun-22"),count=c(3,0,2,7,0,6,0,0,0))
df
df$fraction = df$count / sum(df$count)
df$fraction
df$ymax = cumsum(df$fraction)
df$ymax
df$ymin = c(0, head(df$ymax, n=-1))
df$ymin
ggplot(df, aes(ymax=ymax, ymin=ymin, xmax=4, xmin=3, fill=Date)) +geom_rect() +ggtitle("Section 1 expected graduation year") +coord_polar(theta="y") +xlim(c(2, 4))
```



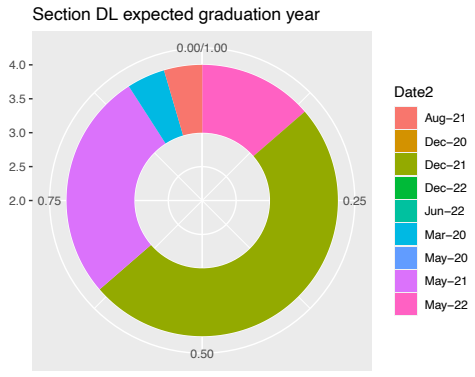
Section 004

```
cou<-ed2[ed2=="May-20"]
length(cou)
cou1<-ed2[ed2=="Dec-20"]
length(cou1)
cou2<-ed2[ed2=="May-22"]
length(cou2)
cou3<-ed2[ed2=="Dec-21"]
length(cou3)
cou4<-ed2[ed2=="Dec-22"]
length(cou4)
cou5<-ed2[ed2=="May-21"]
length(cou5)
cou6<-ed2[ed2=="Mar-20"]
length(cou6)
cou7<-ed2[ed2=="Aug-21"]
length(cou7)
cou8<-ed2[ed2=="Jun-22"]
length(cou8)
df1 <- data.frame(Date1=c("May-20","Dec-20","May-22","Dec-21","Dec-22","May-21","Mar-20","Aug-21","Jun-22"),count1=c(0,3,4,4,5,5,1,0,1))
df1
df1$fraction = df1$count1 / sum(df1$count1)
df1$fraction
df1$ymax = cumsum(df1$fraction)
df1$ymax
df1$ymin = c(0, head(df1$ymax, n=-1))
df1$ymin
ggplot(df1, aes(ymax=ymax, ymin=ymin, xmax=4, xmin=3, fill=Date1)) +geom_rect() +ggtitle("Section 4 expected graduation year") +coord_polar(theta="y") +xlim(c(2, 4))
```



Section DL1

```
co<-ed3[ed3=="May-20"]
length(co)
co1<-ed3[ed3=="Dec-20"]
length(co1)
co2<-ed3[ed3=="May-22"]
length(co2)
co3<-ed3[ed3=="Dec-21"]
length(co3)
co4<-ed3[ed3=="Dec-22"]
length(co4)
co5<-ed3[ed3=="May-21"]
length(co5)
co6<-ed3[ed3=="Mar-20"]
length(co6)
co7<-ed3[ed3=="Aug-21"]
length(co7)
co8<-ed3[ed3=="Jun-22"]
length(co8)
df2 <- data.frame(Date2=c("May-20","Dec-20","May-22","Dec-21","Dec-22","May-21","Mar-20","Aug-21","Jun-22"),count2=c(0,0,3,11,0,6,1,1,0))
df2
df2$fraction = df2$count2 / sum(df2$count2)
df2$fraction
df2$ymax = cumsum(df2$fraction)
df2$ymax
df2$ymin = c(0, head(df2$ymax, n=-1))
df2$ymin
ggplot(df2, aes(ymax=ymax, ymin=ymin, xmax=4, xmin=3, fill=Date2)) +geom_rect() +ggtitle("Section DL expected graduation year") +coord_polar(theta="y") +xlim(c(2, 4))
```



ExpectedGraduationDate_summary<-summary(data$Expected.Graduation.date)
ExpectedGraduationDate_summary

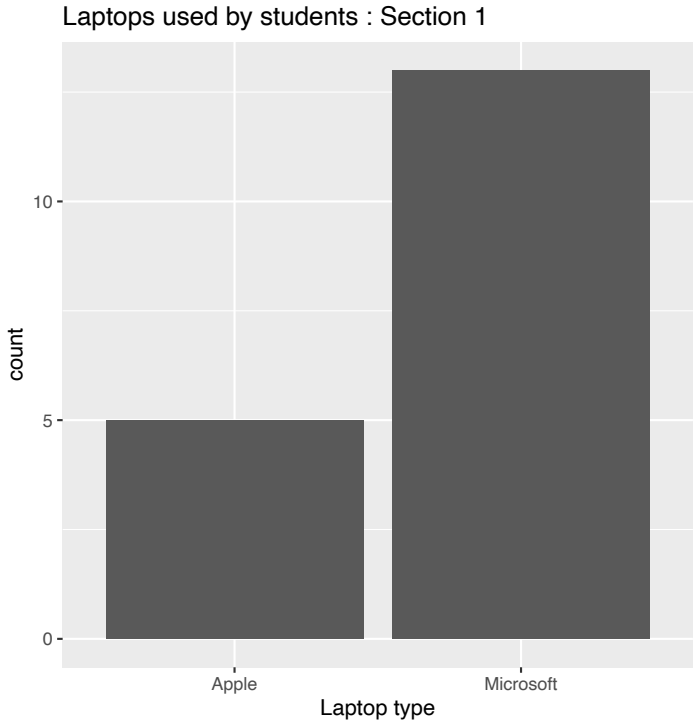
```
> ExpectedGraduationDate_summary<-summary(data$Expected.Graduation.date)
> ExpectedGraduationDate_summary
20-Dec 20-Mar 20-May 21-Aug 21-Dec 21-May 22-Dec 22-Jun 22-May
     3      2      3      1     22     17      5      1      9
```

Laptop type

data$Laptop.type = mapvalues(data$Laptop.type, from = c(1, 2), to = c("Microsoft", "Apple"))
s1=filter(data,data$Section==1)
s4=filter(data,data$Section==4)
sd=filter(data,data$Section=="DL")

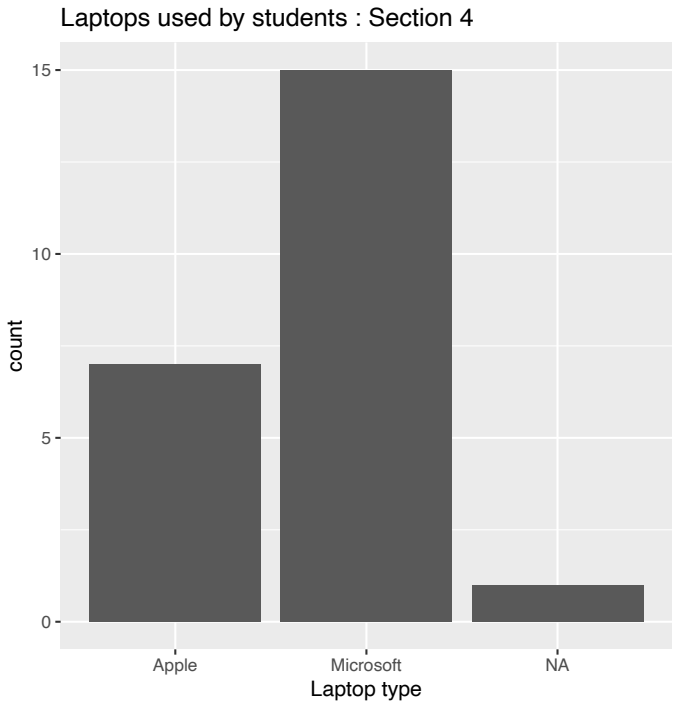
Section 001

ggplot(s1, aes(x = s1$Laptop.type)) + geom_bar() + ggtitle("Laptops used by students : Section 1")+xlab("Laptop type")



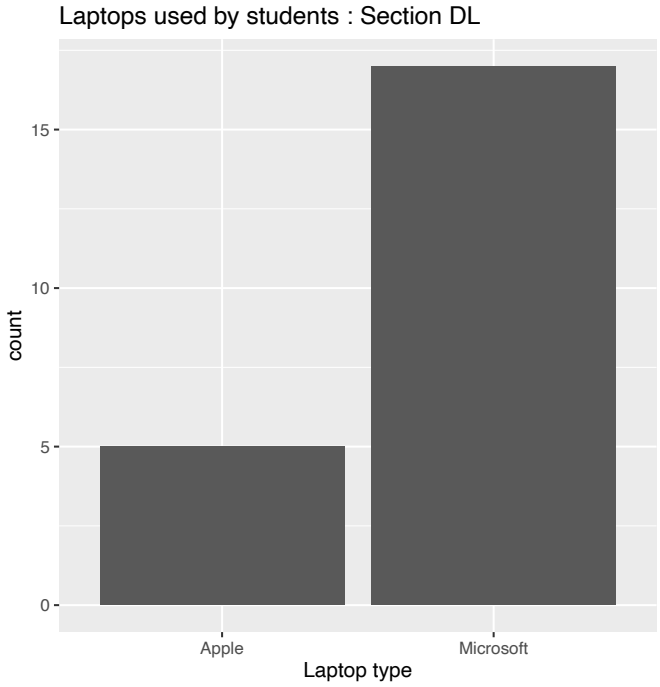
Section 004

ggplot(s4, aes(x = s4$Laptop.type)) + geom_bar() + ggtitle("Laptops used by students : Section 4")+xlab("Laptop type")



Section DL1

ggplot(sd, aes(x = sd$Laptop.type)) + geom_bar() + ggtitle("Laptops used by students : Section DL")+xlab("Laptop type")



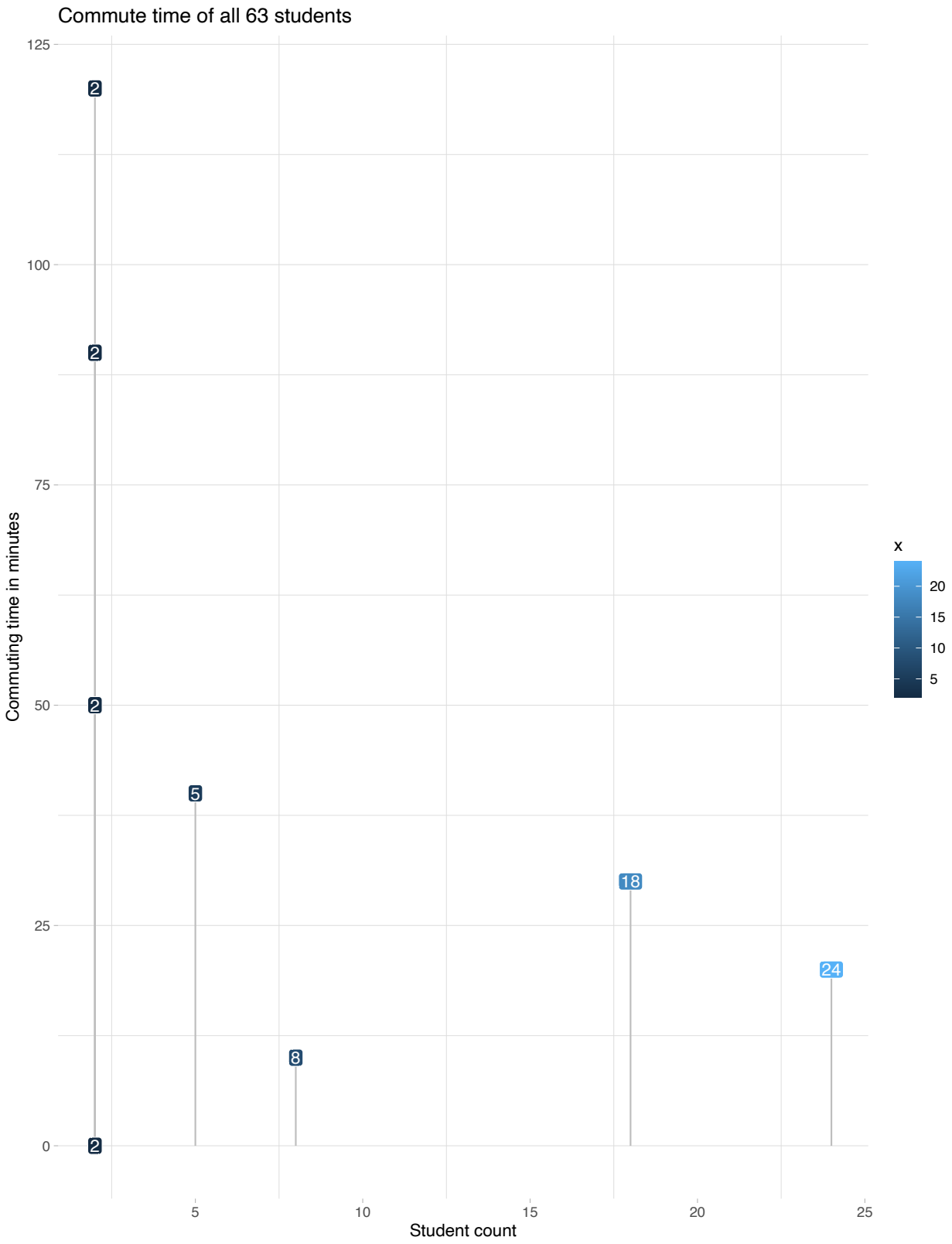
```
Laptoptype_summary<-summary(data$Laptop.type)
Laptoptype_summary
```

```
> Laptoptype_summary<-summary(data$Laptop.type)
> Laptoptype_summary
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
  1.000   1.000   1.000   1.274   2.000   2.000       1
```

Commuting time

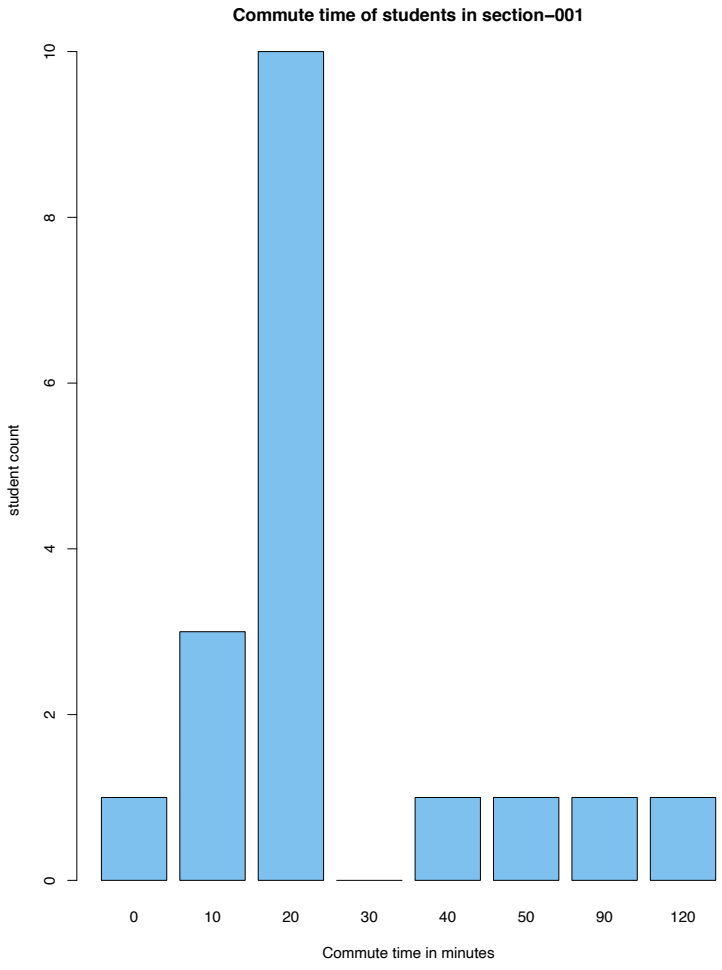
For all sections

```
ze=count(filter(data,data$Commuting.time..minutes...to.class=="0"))
te=count(filter(data,data$Commuting.time..minutes...to.class=="10"))
tw=count(filter(data,data$Commuting.time..minutes...to.class=="20"))
th=count(filter(data,data$Commuting.time..minutes...to.class=="30"))
fo=count(filter(data,data$Commuting.time..minutes...to.class=="40"))
fi=count(filter(data,data$Commuting.time..minutes...to.class=="50"))
ni=count(filter(data,data$Commuting.time..minutes...to.class=="90"))
ot=count(filter(data,data$Commuting.time..minutes...to.class=="120"))
commute=unlist(c(ze,te,tw,th,fo,fi,ni,ot))
ct=data.frame(x=commute ,y=c(0,10,20,30,40,50,90,120))
ggplot(ct, aes(x=x, y=y)) +
 geom_segment( aes(x=x, xend=x, y=0, yend=y), color="grey") +
 geom_label(aes(x=x, fill=x, label=x),label.padding=unit(1.5,"pt"),color="white")+
 theme_light() +
 theme(
  panel.grid.major.x = element_blank(),
  panel.border = element_blank(),
  axis.ticks.x = element_line()
 ) +
 xlab("Student count") +
 ylab("Commuting time in minutes")+
 ggtitle("Commute time of all 63 students")
```


Commute time of all 63 students
125
100
75
50
25
0
2
2
2
5
8
18
24
Commuting time in minutes
5
10
15
20
25
Student count
x
20
15
10
5

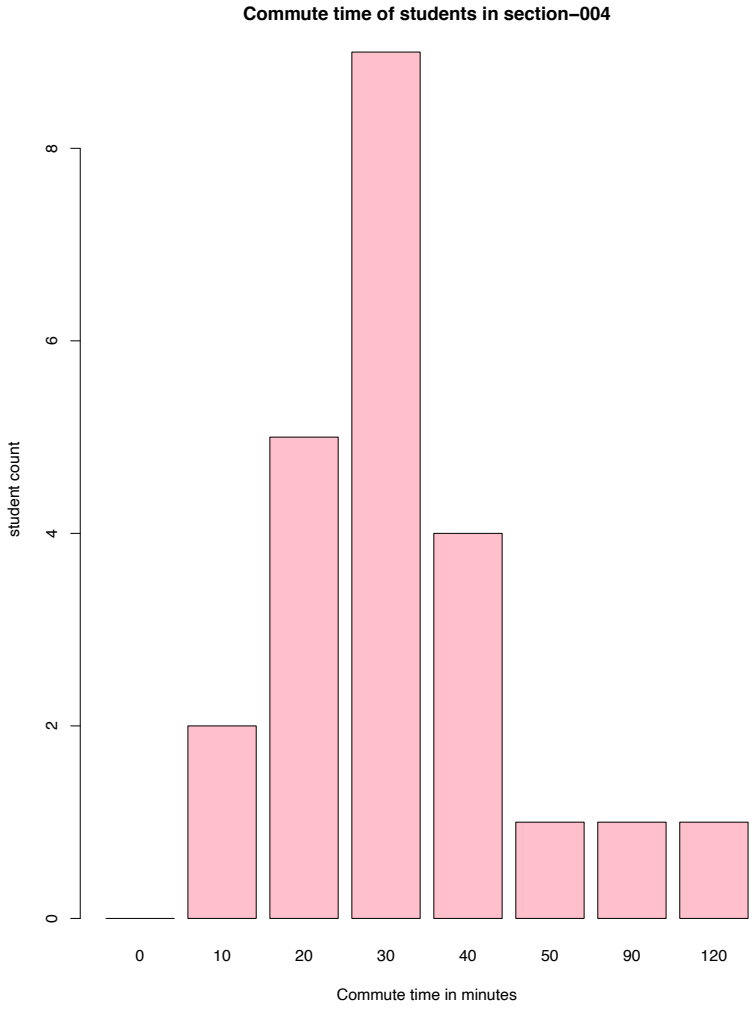
Section 001

```
s1=filter(data,data$Section==1)
ze1=count(filter(s1,s1$Commuting.time..minutes...to.class==0))
te1=count(filter(s1,s1$Commuting.time..minutes...to.class==10))
tw1=count(filter(s1,s1$Commuting.time..minutes...to.class==20))
th1=count(filter(s1,s1$Commuting.time..minutes...to.class==30))
fo1=count(filter(s1,s1$Commuting.time..minutes...to.class==40))
fi1=count(filter(s1,s1$Commuting.time..minutes...to.class==50))
ni1=count(filter(s1,s1$Commuting.time..minutes...to.class==90))
ot1=count(filter(s1,s1$Commuting.time..minutes...to.class==120))
commute1=t(data.matrix(unlist(c(ze1,te1,tw1,th1,fo1,fi1,ni1,ot1))))
ct1=data.frame(x=commute1 ,y=c(0,10,20,30,40,50,90,120))
colnames(commute1)=c("0","10","20","30","40","50","90","120")
barplot(commute1,xlab="Commute time in minutes",
      ylab="student count",
      main="Commute time of students in section-001",col = "skyblue2")
```



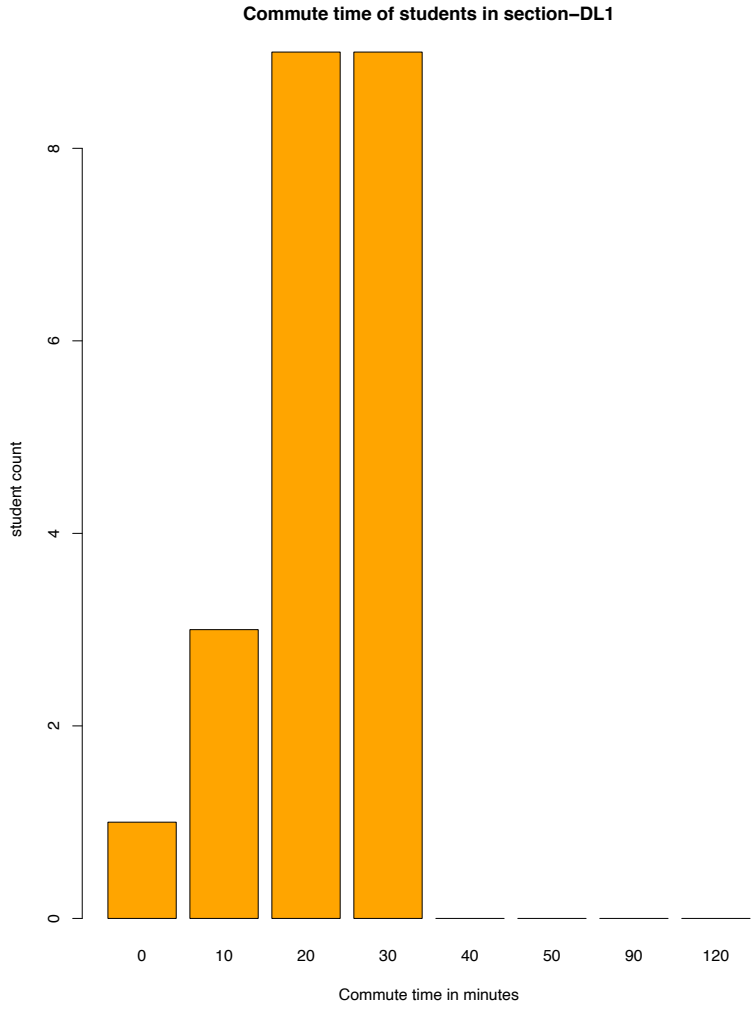
Section 004

```
s4=filter(data,data$Section==4)
ze4=count(filter(s4,s4$Commuting.time..minutes...to.class==0))
te4=count(filter(s4,s4$Commuting.time..minutes...to.class==10))
tw4=count(filter(s4,s4$Commuting.time..minutes...to.class==20))
th4=count(filter(s4,s4$Commuting.time..minutes...to.class==30))
fo4=count(filter(s4,s4$Commuting.time..minutes...to.class==40))
fi4=count(filter(s4,s4$Commuting.time..minutes...to.class==50))
ni4=count(filter(s4,s4$Commuting.time..minutes...to.class==90))
ot4=count(filter(s4,s4$Commuting.time..minutes...to.class==120))
commute4=t(data.matrix(unlist(c(ze4,te4,tw4,th4,fo4,fi4,ni4,ot4))))
colnames(commute4)=c("0","10","20","30","40","50","90","120")
barplot(commute4,xlab="Commute time in minutes",
      ylab="student count",
      main="Commute time of students in section-004",col = "pink")
```



Section DL1

```
sd=filter(data,data$Section=="DL")
zed=count(filter(sd,sd$Commuting.time..minutes...to.class==0))
ted=count(filter(sd,sd$Commuting.time..minutes...to.class==10))
twd=count(filter(sd,sd$Commuting.time..minutes...to.class==20))
thd=count(filter(sd,sd$Commuting.time..minutes...to.class==30))
fod=count(filter(sd,sd$Commuting.time..minutes...to.class==40))
fid=count(filter(sd,sd$Commuting.time..minutes...to.class==50))
nid=count(filter(sd,sd$Commuting.time..minutes...to.class==90))
otd=count(filter(sd,sd$Commuting.time..minutes...to.class==120))
commuted=t(data.matrix(unlist(c(zed,ted,twd,thd,fod,fid,nid,otd))))
colnames(commuted)=c("0","10","20","30","40","50","90","120")
barplot(commuted,xlab="Commute time in minutes",
      ylab="student count",
      main="Commute time of students in section-DL1",col = "orange")
```



commutingtime_summary<-summary(data$Commuting.time..minutes...to.class)
commutingtime_summary

```
> commutingtime_summary<-summary(data$Commuting.time..minutes...to.class)
> commutingtime_summary
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00   20.00   20.00   28.89   30.00  120.00
```

Employment status

data$Employement.status <- mapvalues(data$Employement.status, from = c(1.0, 2.0, 3.0), to = c("Yes, Full Time", "Working, but not Full Time", "Not Working while attending Mason"))

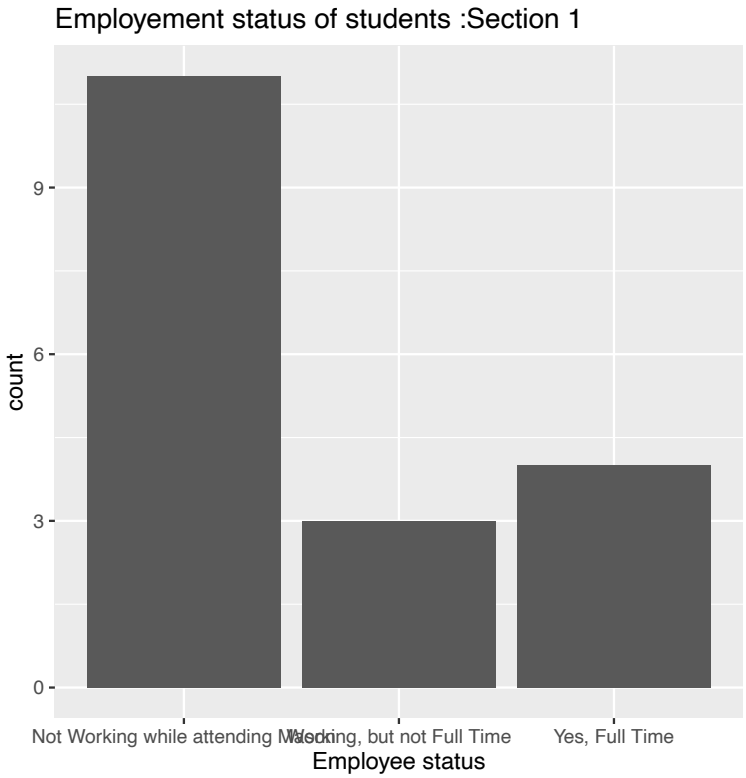
s1=filter(data,data$Section==1)

s4=filter(data,data$Section==4)

sd=filter(data,data$Section=="DL")

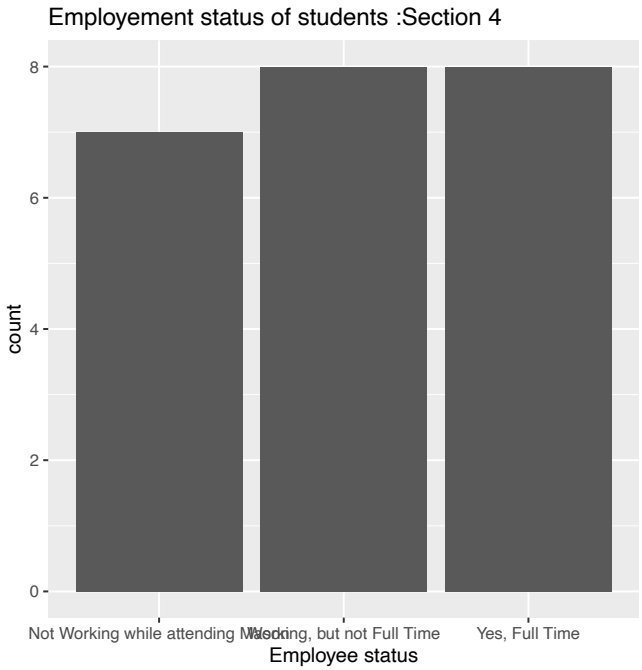
Section 001

ggplot(s1, aes(x = Employement.status)) + geom_bar() + ggtitle("Employement status of students :Section 1")+xlab("Employee status")



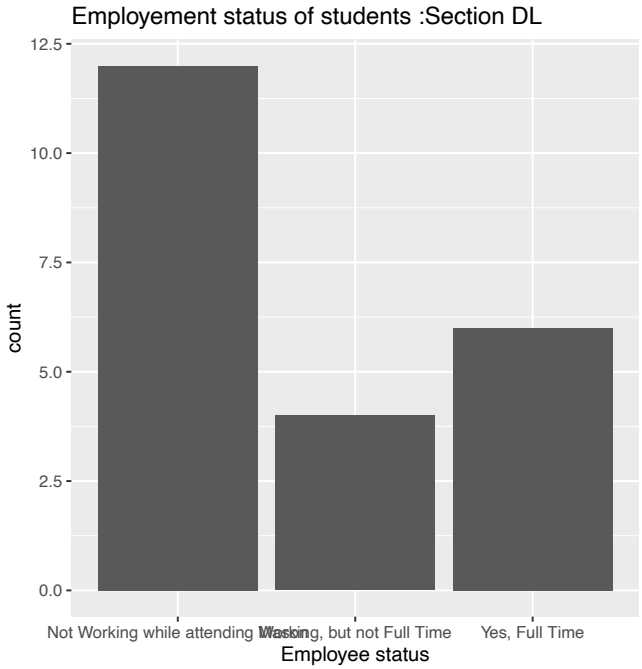
Section 004

ggplot(s4, aes(x = Employement.status)) + geom_bar() + ggtitle("Employement status of students :Section 4")+xlab("Employee status")



Section DL1

ggplot(sd, aes(x = Employement.status)) + geom_bar() + ggtitle("Employement status of students :Section DL")+xlab("Employee status")



Employmentstatus_summary<-summary(data$Employement.status)

Employmentstatus_summary

```
> Employmentstatus_summary<-summary(data$Employement.status)
> Employmentstatus_summary
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1.00    1.00    2.00    2.19    3.00    3.00
```

Programming skills

data$programming.skill.in.Python <- mapvalues(data$programming.skill.in.Python, from = c(1.0, 2.0, 3.0, 4.0, 5.0), to = c("Little/none", "Some familiarity", "Average user", "Frequent use for projects", "Fluent/expert"))

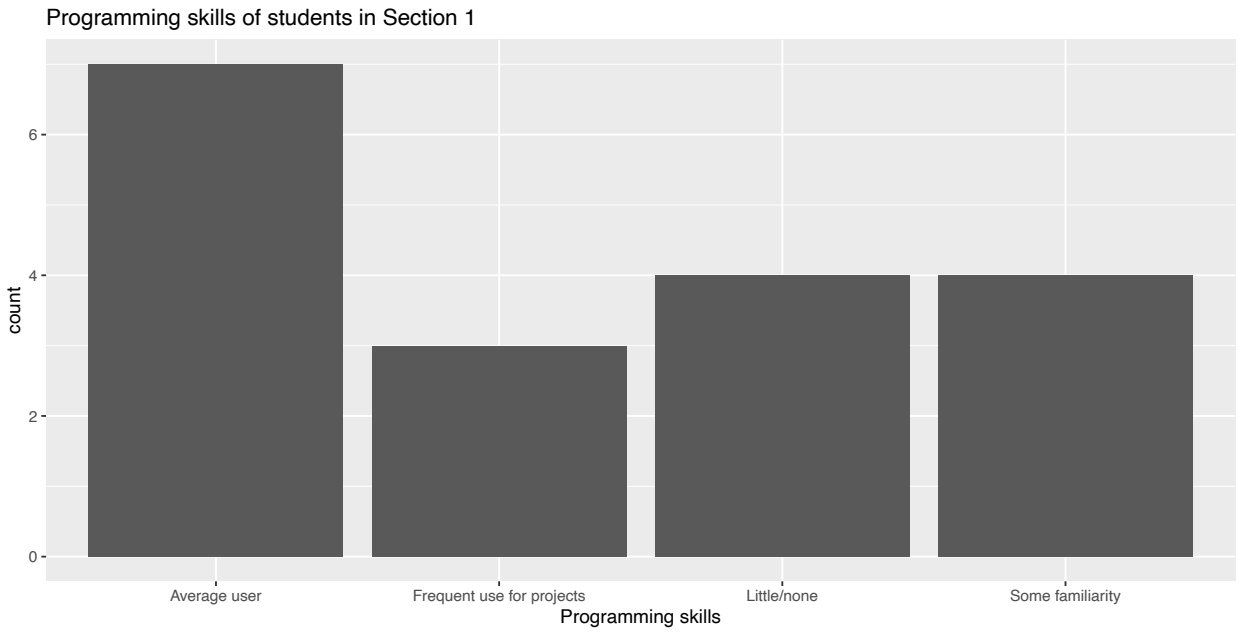
s1=filter(data,data$Section==1)

s4=filter(data,data$Section==4)

sd=filter(data,data$Section=="DL")

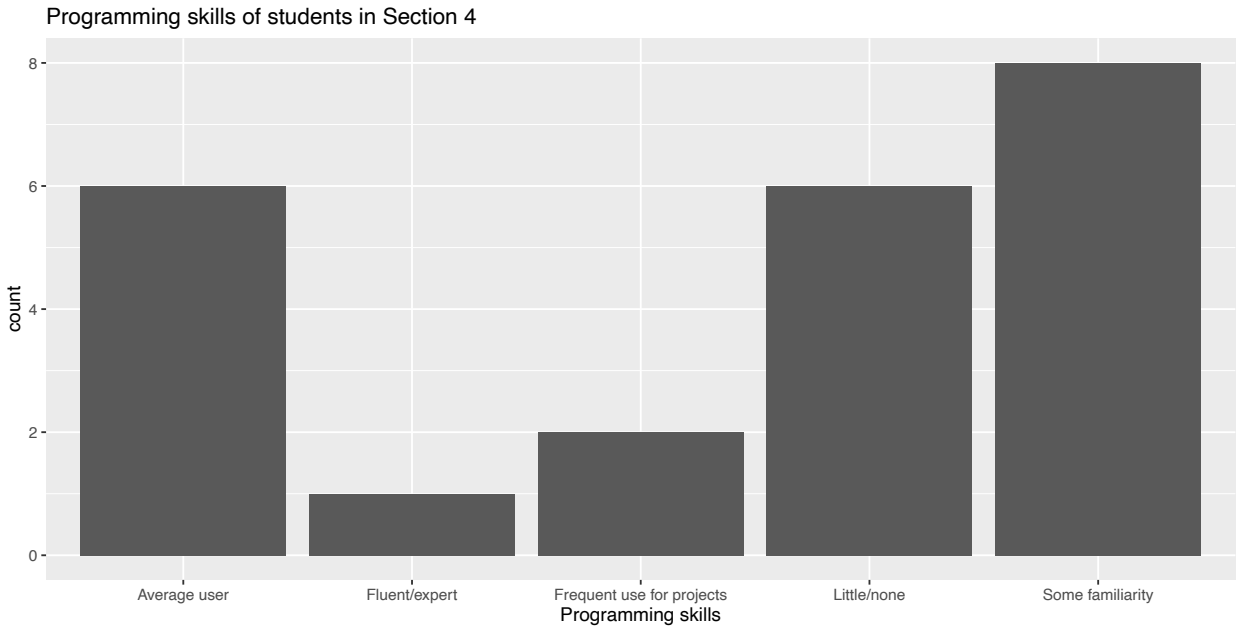
Section 001

ggplot(s1, aes(x = programming.skill.in.Python)) + geom_bar() + ggtitle("Programming skills of students in section1")+xlab("Programming skills")



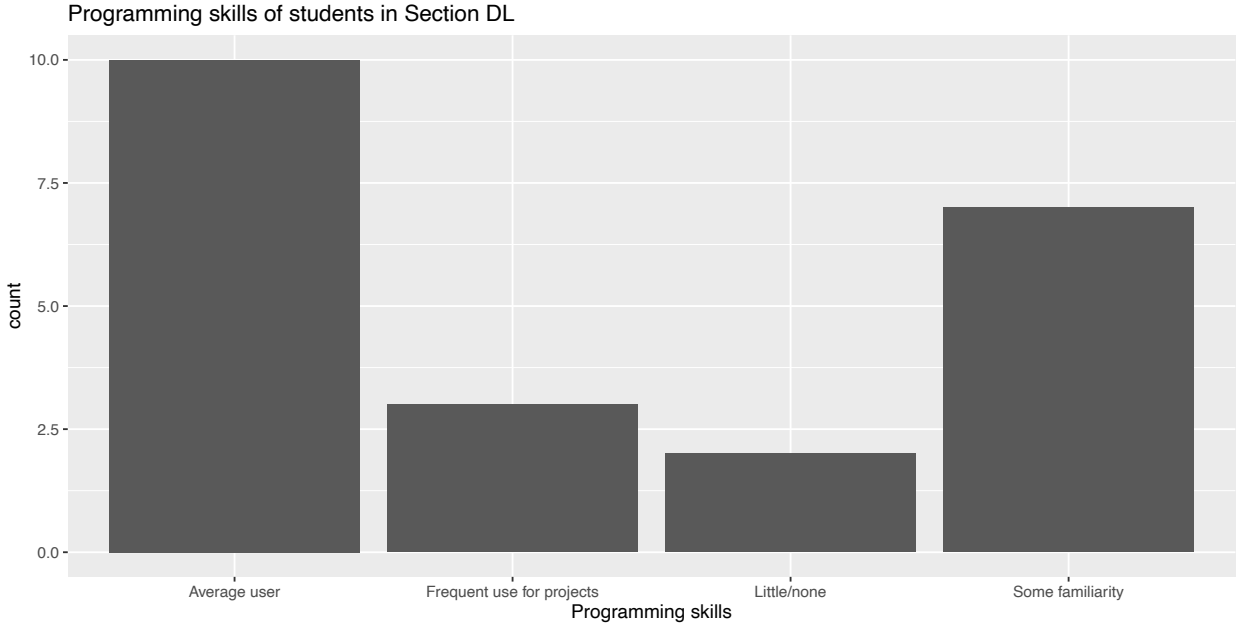
Section 004

ggplot(s4, aes(x = programming.skill.in.Python)) + geom_bar() + ggtitle("Programming skills of students in section4")+xlab("Programming skills")



Section DL1

ggplot(sd, aes(x = programming.skill.in.Python)) + geom_bar() + ggtitle("Programming skills of students in sectionDL")+xlab("Programming skills")



Programmingskill_summary<-summary(data$programming.skill.in.Python)
Programmingskill_summary

```
> Programmingskill_summary<-summary(data$programming.skill.in.Python)
> Programmingskill_summary
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   2.000   3.000   2.476   3.000   5.000
```

c. Discuss some suggestions for how to improve the online learning experience for this experiment.

The following points help to improve the online learning experience more effectively.

Result-oriented evaluations play a key role in the online learning platform. This essentially analyzes the fields where students were unable to perform, so that the students can perform well on their weaknesses based on the statistical results.

Gathering the experience of the students with online mentoring on satisfactory levels with the advising and mentoring have been getting in each area is necessary. This information can be used as a baseline to adjust and improve current and future actions. Apart from the course's routine learning, online workshops need to be conducted which are related to career skills with respect to their specializations so that most of the students get benefitted in every aspect. So that the overall quality of academic advising and guidance can be improved by conducting a survey. Peer interactions would lack in online education when compared to traditional classroom teaching. Frequent group discussion activities should be arranged, and presentations need to be conducted on a timely basis which would concentrate not only on the curriculum but also on the team-building qualities of an individual which is very much needed in the current day.

Another hurdle that few students fail to resolve is when there is an absence of a facilitator at a given place and time over the period of one semester. So some students experience a learning curve when they first acclimated by asking questions in different forums, rather than in a conventional face-to-face, classroom setting so professors should always be highly interactive with the students who lack motivation, whether intrinsic or extrinsic can easily lose sight of their main goal and they should also be accountable to any of the queries with respect to students.

Team

Sno.	Name	G#	Email	Phone
1	Sai Chaitanya Sadasivuni	G01241462	ssadasiv@gmu.edu	+15716978469
2	Vinuthna Chillakuru	G01229488	vchillak@gmu.edu	+15716978469
3	Deepthi Simha Akula	G01221203	pakula2@gmu.edu	+15716978469