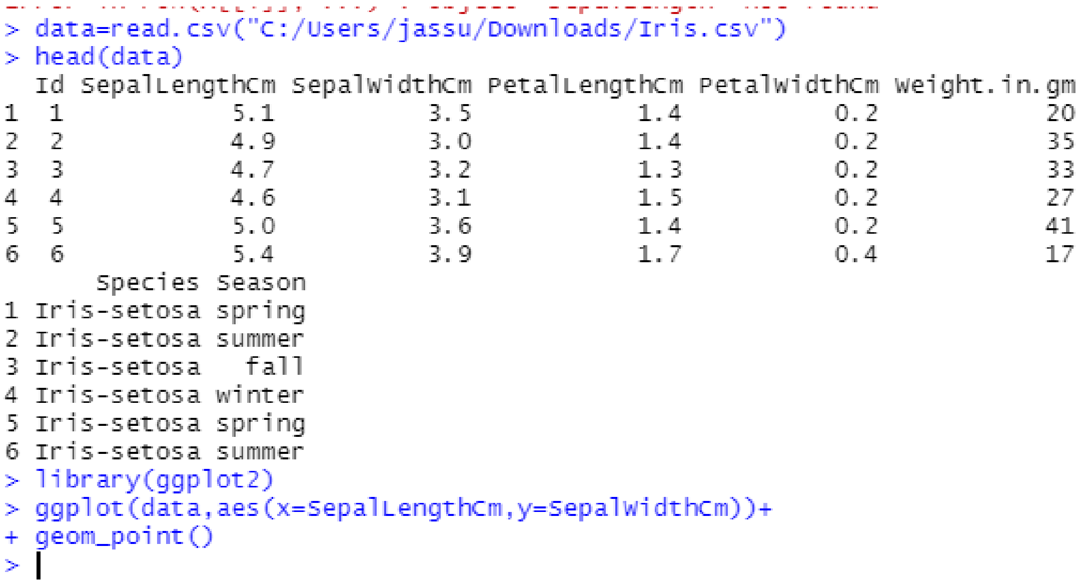
VIT-AP UNIVERSITY, ANDHRA PRADESH

CSE4027 – Data Analytics - Lab Sheet :7

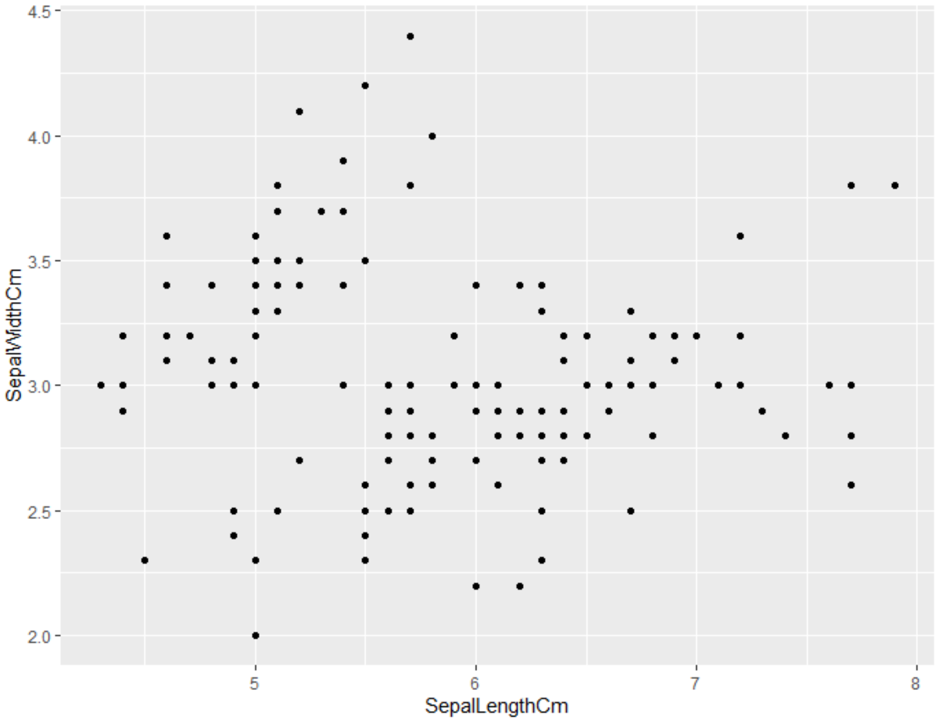
Academic year: 2022-2023 **Branch/ Class:** B.Tech
Semester: Fall **Date: 21/10/22**
Faculty Name: Dr Syed Khasim **School:** SCOPE
Student name: Majjiga Jaswanth **Reg. no.: 20BCD7171**

PART -1

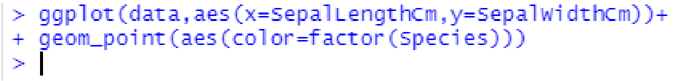
1. Draw Basic scatter plot between SepalLengthCm and SepalWidthCm.



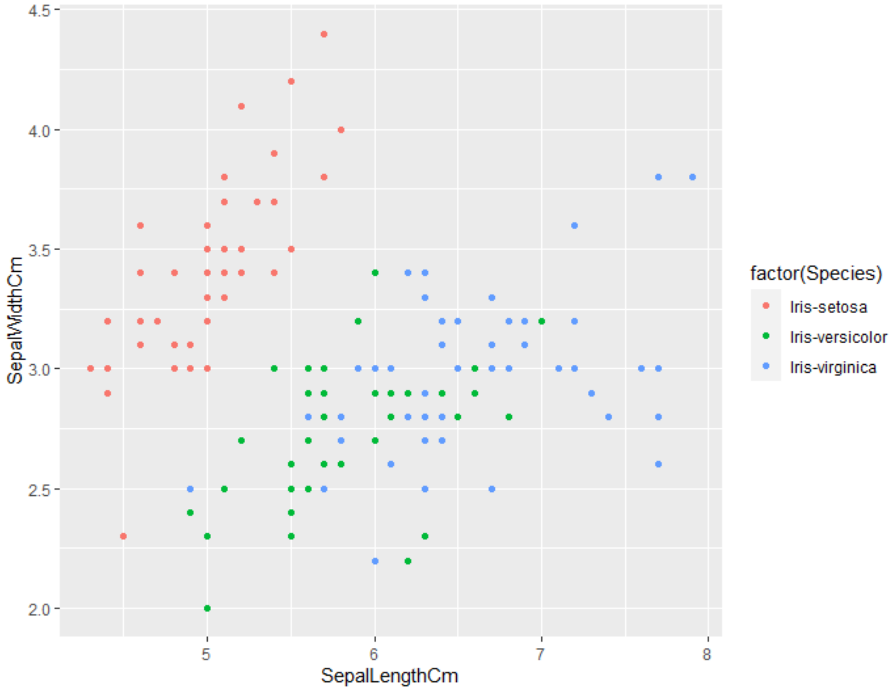
```
> data=read.csv("C:/Users/jassu/Downloads/Iris.csv")
> head(data)
  Id SepalLengthCm SepalWidthCm PetalLengthCm PetalWidthCm weight.in.gm
1  1           5.1          3.5           1.4          0.2           20
2  2           4.9          3.0           1.4          0.2           35
3  3           4.7          3.2           1.3          0.2           33
4  4           4.6          3.1           1.5          0.2           27
5  5           5.0          3.6           1.4          0.2           41
6  6           5.4          3.9           1.7          0.4           17
      Species Season
1 Iris-setosa spring
2 Iris-setosa summer
3 Iris-setosa   fall
4 Iris-setosa winter
5 Iris-setosa spring
6 Iris-setosa summer
> library(ggplot2)
> ggplot(data,aes(x=SepalLengthCm,y=SepalWidthCm))+
+ geom_point()
> |
```



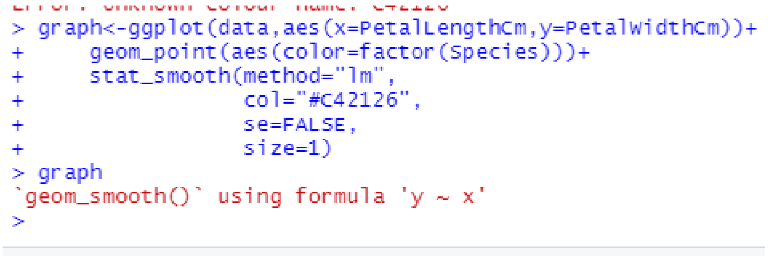
2. Visualize Scatter plot with color between groupSepalLengthCm and SepalWidthCm and group by Species.



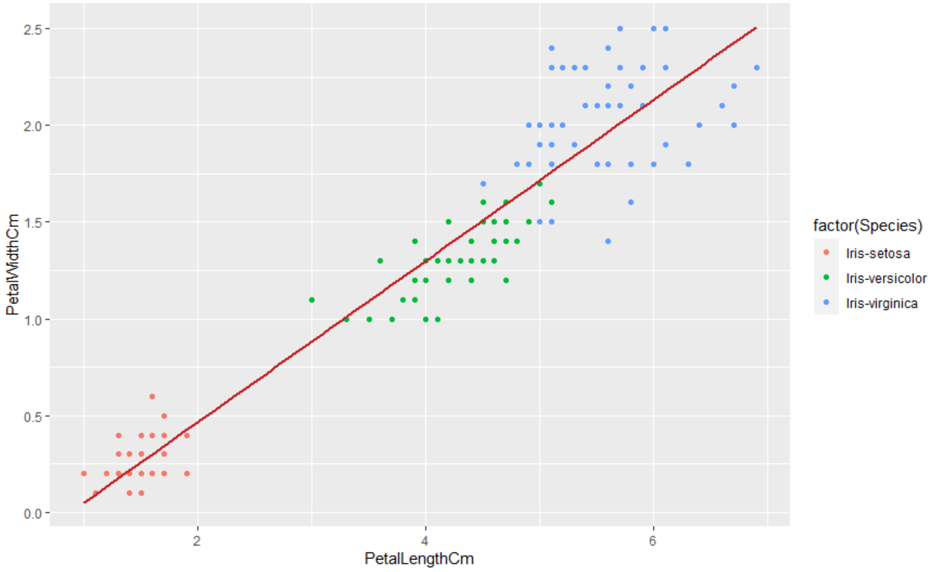
```
> ggplot(data,aes(x=SepalLengthCm,y=SepalWidthCm))+
+ geom_point(aes(color=factor(Species)))
> |
```



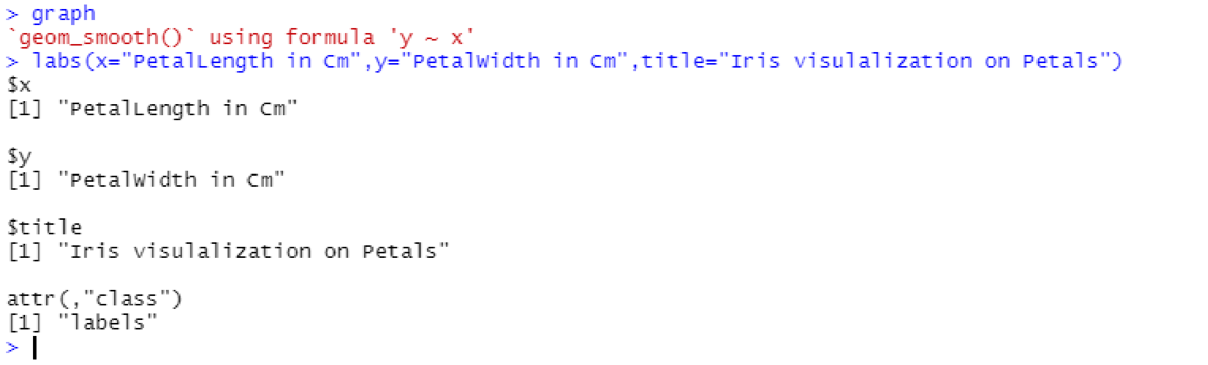
3. Visualize Scatter plot with added fitted values between PetalLengthCm andPetalWidthCm and use Linear regression for fitted line.



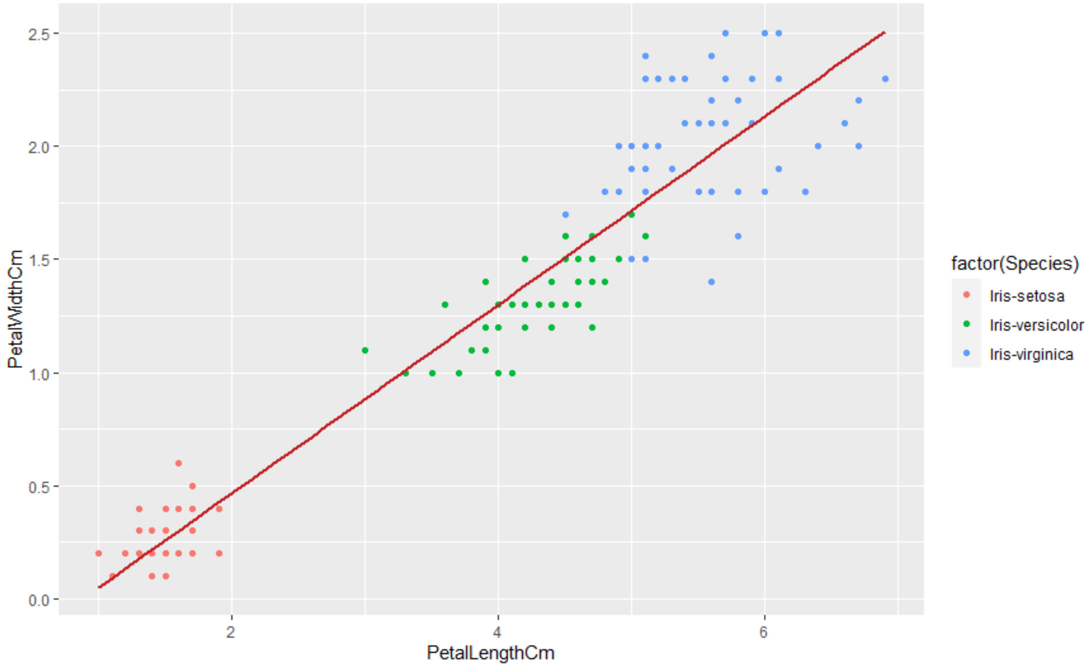
```
Error: Unknown colour name: C42126
> graph<-ggplot(data,aes(x=PetalLengthCm,y=PetalWidthCm))+
+     geom_point(aes(color=factor(Species)))+
+     stat_smooth(method="lm",
+                 col="#C42126",
+                 se=FALSE,
+                 size=1)
> graph
`geom_smooth()` using formula 'y ~ x'
>
```



4. Add the following information to the above drawn graph
    a. Add a title as “Iris visualization on Petal”
    b. Rename x-axis as “PetalLength in Cm” and y-axis as “PetalWidth in Cm”

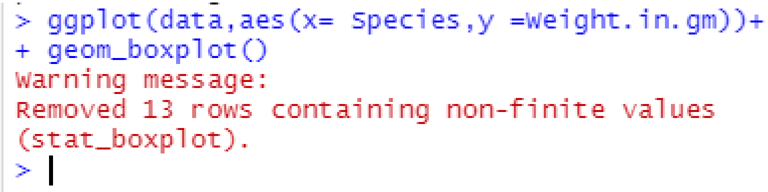


```
> graph
`geom_smooth()` using formula 'y ~ x'
> labs(x="PetalLength in Cm",y="PetalWidth in Cm",title="Iris visulalization on Petals")
$x
[1] "PetalLength in Cm"

$y
[1] "PetalWidth in Cm"

$title
[1] "Iris visulalization on Petals"

attr(,"class")
[1] "labels"
> 
```

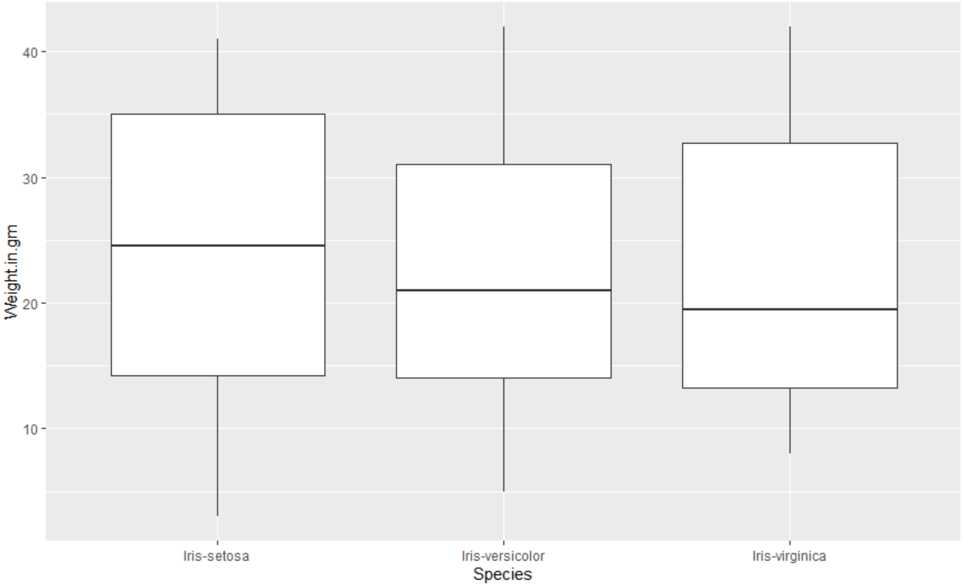


Part-2

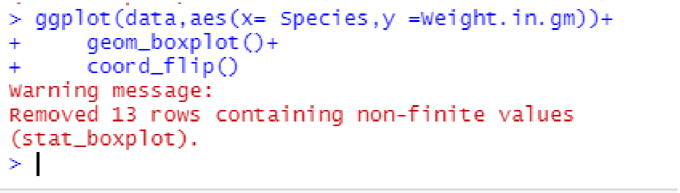
1. Visualize the Basic box plot on species wise weightdata



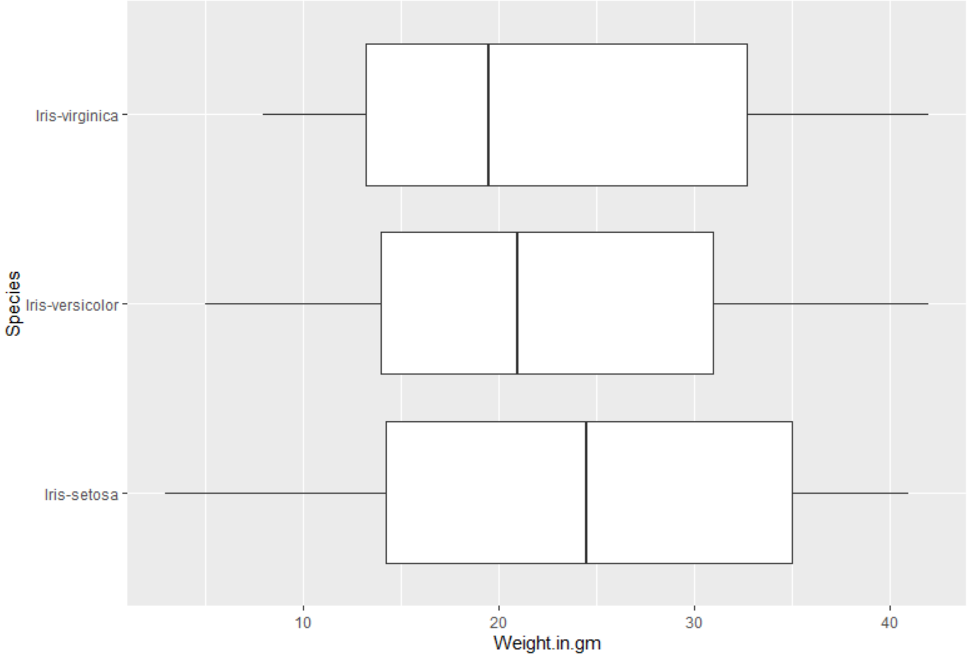
```
> ggplot(data,aes(x= Species,y =Weight.in.gm))+
+ geom_boxplot()
Warning message:
Removed 13 rows containing non-finite values
(stat_boxplot).
> 
```



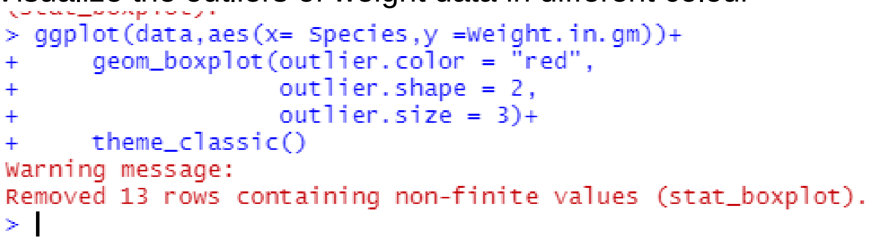
2. Change side of the graph which you have plotted in question 1



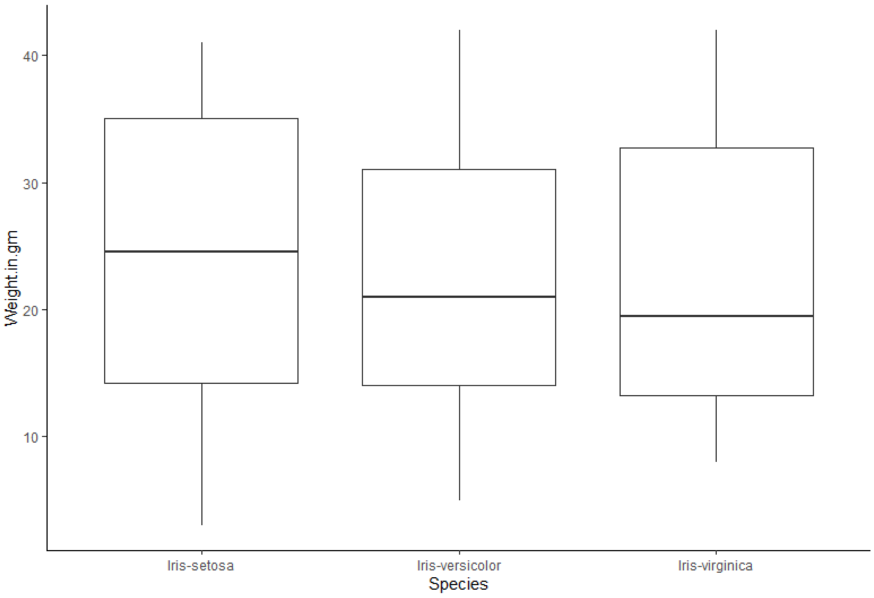
```
> ggplot(data,aes(x= Species,y =Weight.in.gm))+
+     geom_boxplot()+
+     coord_flip()
Warning message:
Removed 13 rows containing non-finite values
(stat_boxplot).
> 
```



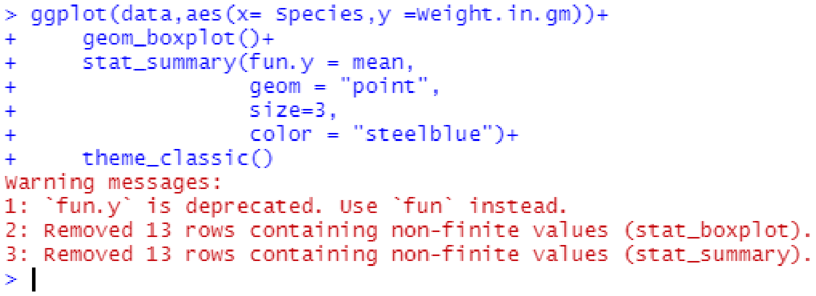
3. Visualize the outliers of weight data in different colour



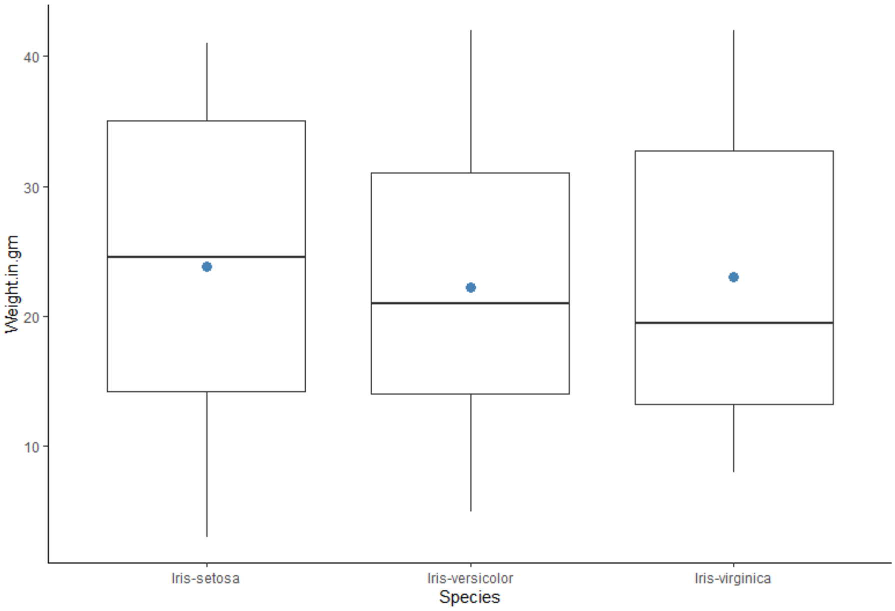
```
> ggplot(data,aes(x= Species,y =Weight.in.gm))+
+     geom_boxplot(outlier.color = "red",
+                  outlier.shape = 2,
+                  outlier.size = 3)+
+     theme_classic()
Warning message:
Removed 13 rows containing non-finite values (stat_boxplot).
> 
```



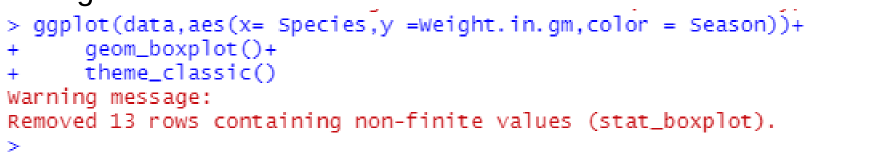
4. Add the summary statistic on the box plot drawn in question 1.



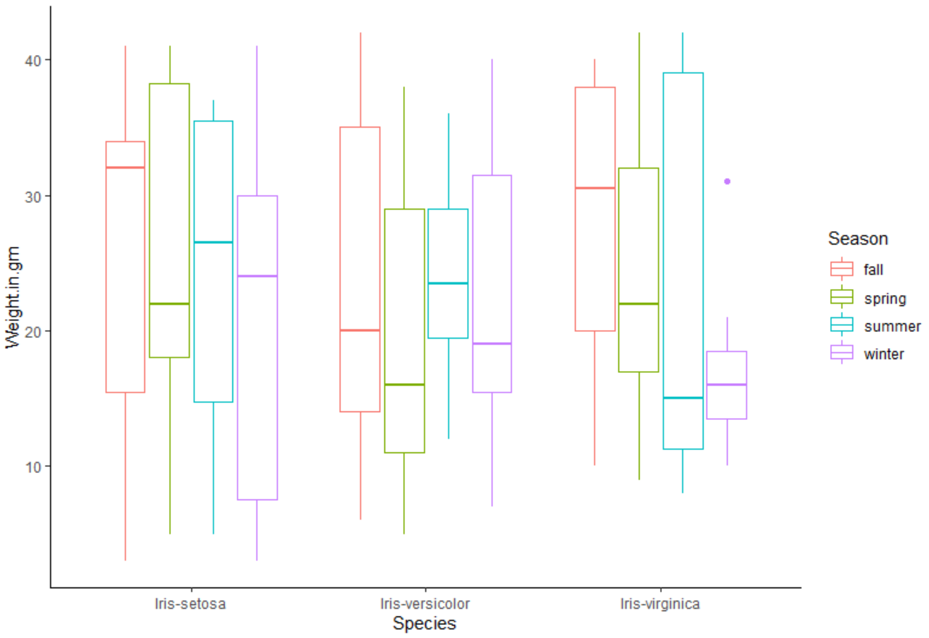
```
> ggplot(data,aes(x= Species,y =Weight.in.gm))+
+     geom_boxplot()+
+     stat_summary(fun.y = mean,
+                  geom = "point",
+                  size=3,
+                  color = "steelblue")+
+     theme_classic()
Warning messages:
1: `fun.y` is deprecated. Use `fun` instead.
2: Removed 13 rows containing non-finite values (stat_boxplot).
3: Removed 13 rows containing non-finite values (stat_summary).
> 
```



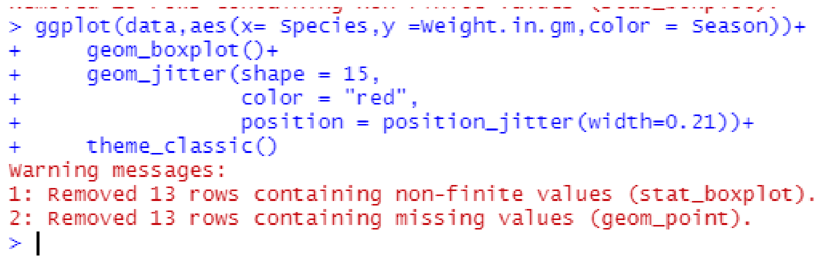
5. Change the colour of the box based on season



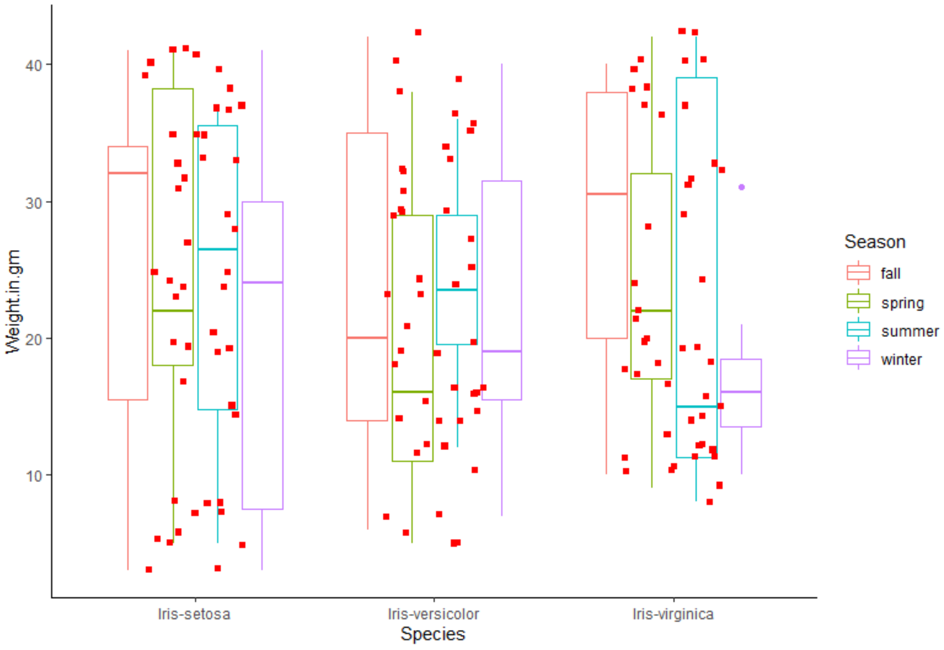
```
> ggplot(data,aes(x= Species,y =Weight.in.gm,color = Season))+
+     geom_boxplot()+
+     theme_classic()
Warning message:
Removed 13 rows containing non-finite values (stat_boxplot).
>
```



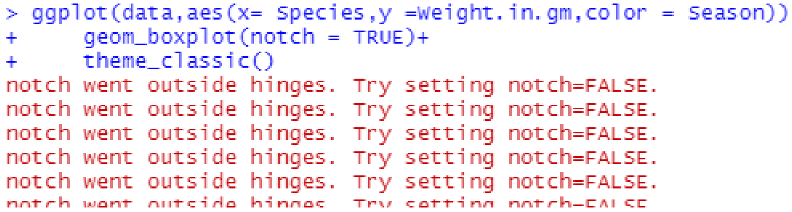
6. Visualize the Box Plot with Jittered Dots



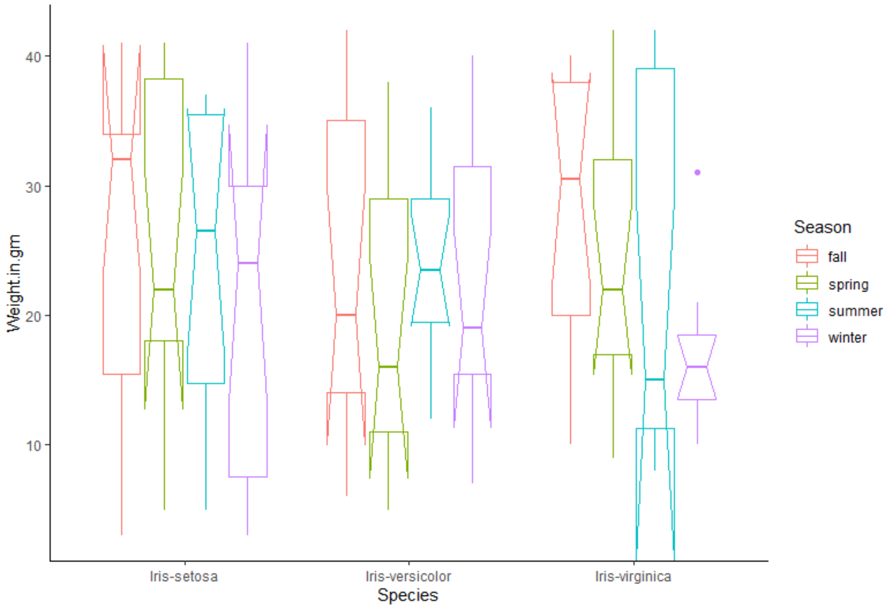
```
> ggplot(data,aes(x= Species,y =Weight.in.gm,color = Season))+
+     geom_boxplot()+
+     geom_jitter(shape = 15,
+                 color = "red",
+                 position = position_jitter(width=0.21))+
+     theme_classic()
Warning messages:
1: Removed 13 rows containing non-finite values (stat_boxplot).
2: Removed 13 rows containing missing values (geom_point).
> |
```



7. Plot Notched Box Plot

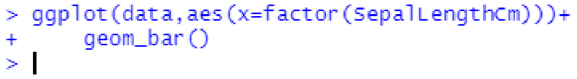


```
> ggplot(data,aes(x= Species,y =Weight.in.gm,color = Season))
+     geom_boxplot(notch = TRUE)+
+     theme_classic()
notch went outside hinges. Try setting notch=FALSE.
notch went outside hinges. Try setting notch=FALSE.
notch went outside hinges. Try setting notch=FALSE.
notch went outside hinges. Try setting notch=FALSE.
notch went outside hinges. Try setting notch=FALSE.
notch went outside hinges. Try setting notch=FALSE.
```

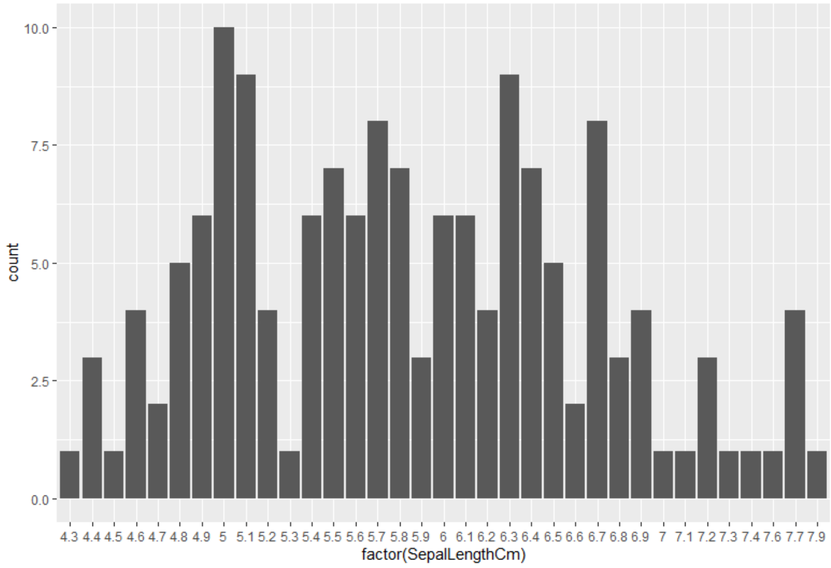


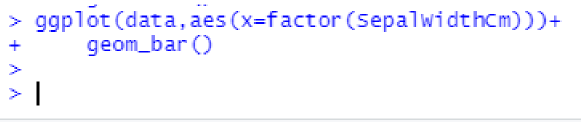
Part-3

1. Show individual geombar plot on factors of SepalLengthCm, SepalWidthCm, SepalLengthCm and SepalWidthCm

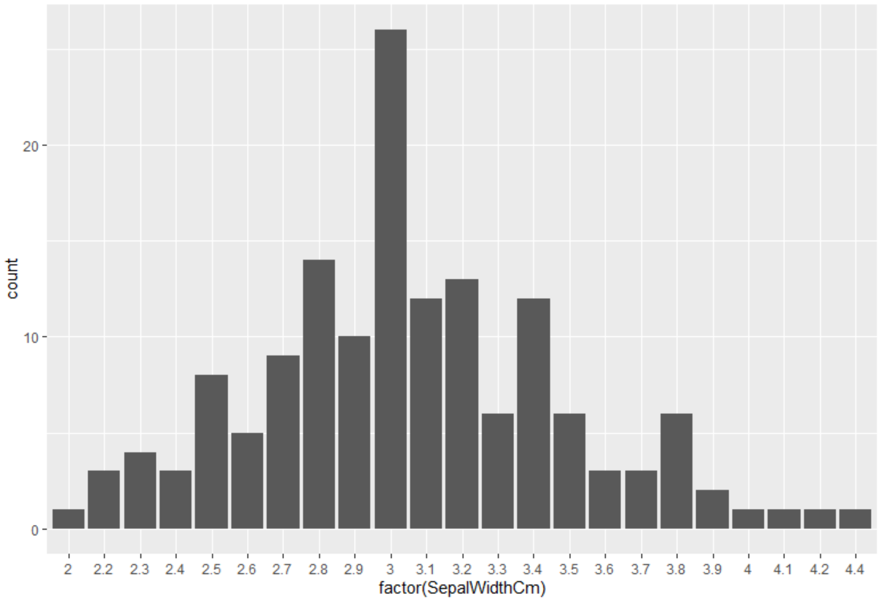


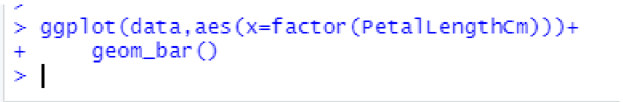
```
> ggplot(data,aes(x=factor(SepalLengthCm)))+
+     geom_bar()
> |
```



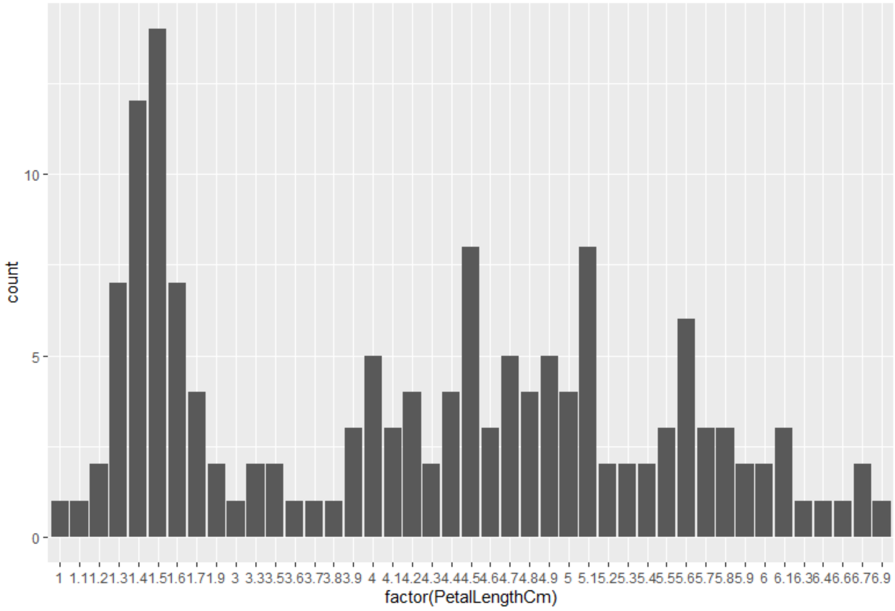


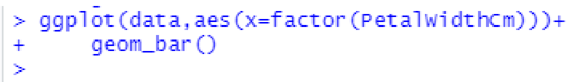
```
> ggplot(data,aes(x=factor(SepalWidthCm))+
+     geom_bar()
>
> |
```



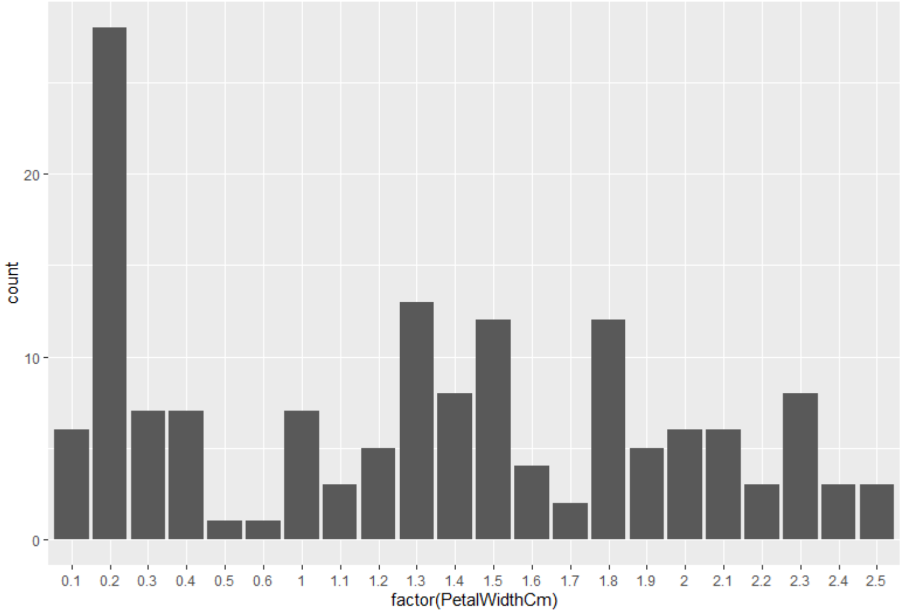


```
> ggplot(data,aes(x=factor(PetalLengthCm)))+
+     geom_bar()
> 
```

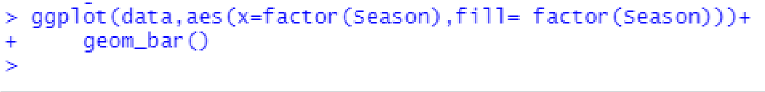




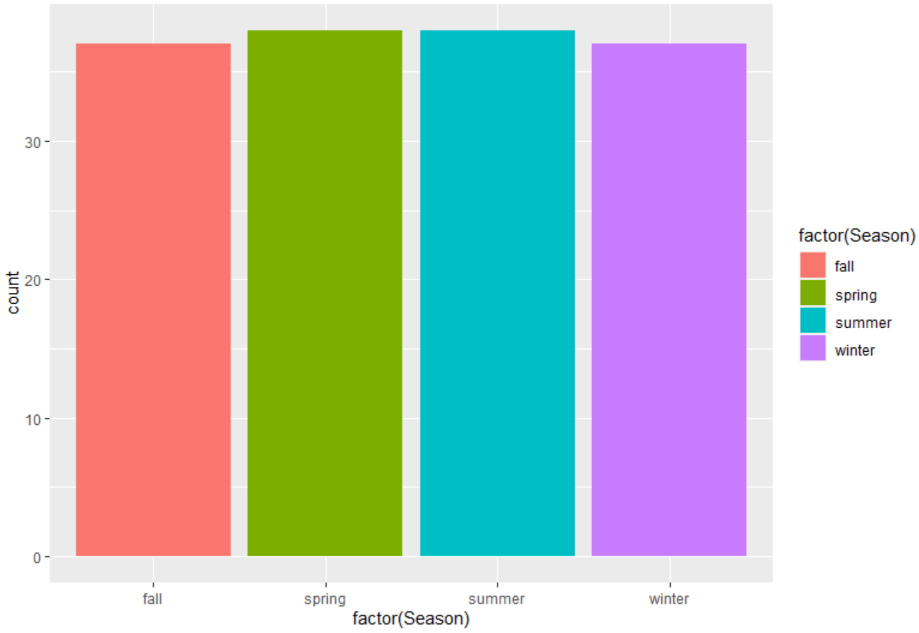
```
> ggplot(data,aes(x=factor(PetalWidthCm))+
+     geom_bar()
>
```



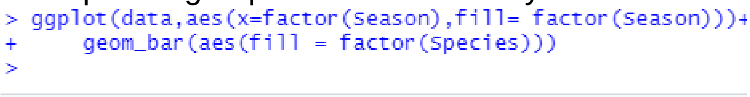
2. Visualize the colour geombar plot on factors of season data



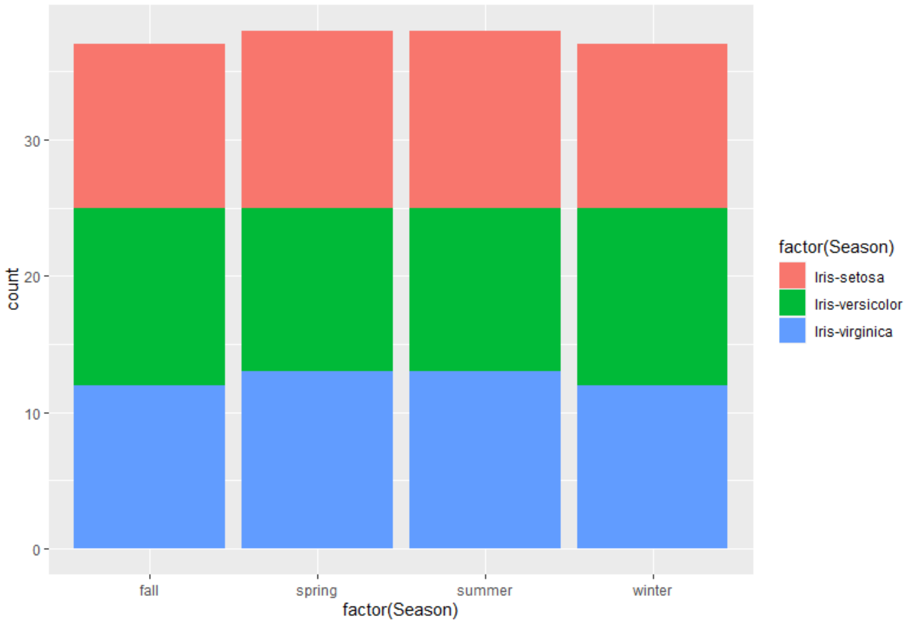
```
> ggplot(data,aes(x=factor(Season),fill= factor(Season)))+
+     geom_bar()
>
```



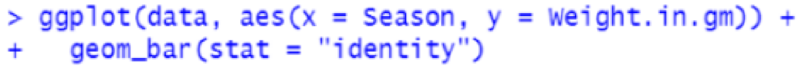
3. Add species group in the bars which you have drawn in question 2.



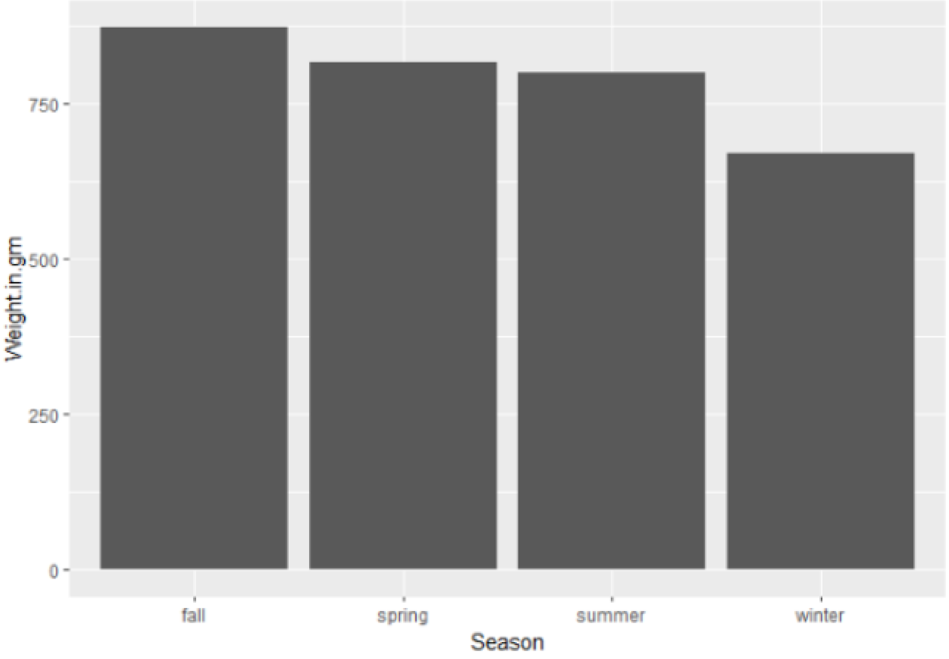
```
> ggplot(data,aes(x=factor(Season),fill= factor(Season)))+
+     geom_bar(aes(fill = factor(Species)))
>
```



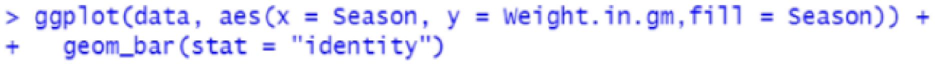
4. Create a basic histogram with season and weight data



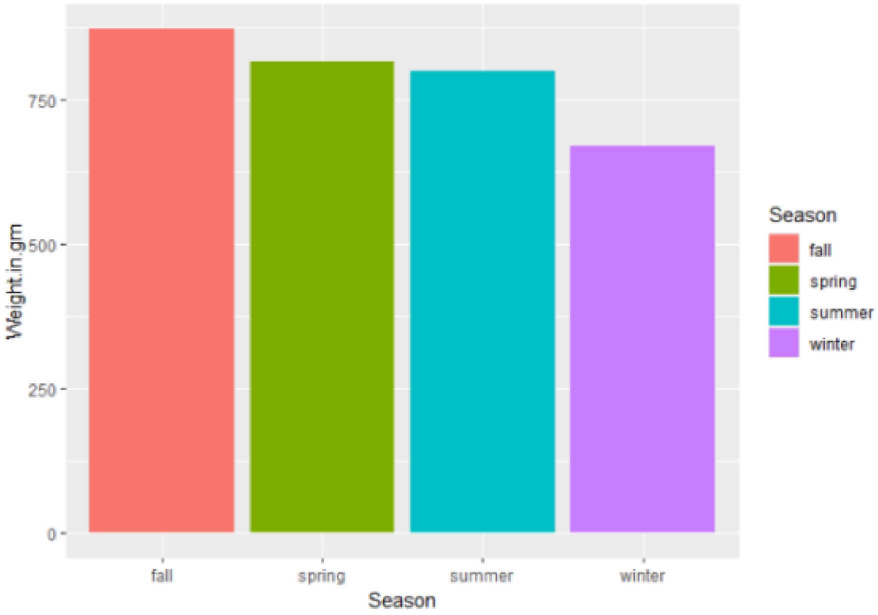
```
> ggplot(data, aes(x = Season, y = Weight.in.gm)) +
+   geom_bar(stat = "identity")
```



5. Change the colour and add labels to the graph which you drawn in question 4



```
> ggplot(data, aes(x = Season, y = Weight.in.gm,fill = Season)) +
+   geom_bar(stat = "identity")
```


Weight.in.gm
750
500
250
0
fall
spring
summer
winter
Season
Season
fall
spring
summer
winter