```
        # R-Project - Comcast Telecom Consumer Complaints

# DESCRIPTION
#Comcast is an American global telecommunication company. The firm has been
providing terrible customer service. They continue to fall short despite
repeated promises to improve. Only last month (October 2016) the authority
fined them a $2.3 million, after receiving over 1000 consumer complaints.
#The existing database will serve as a repository of public customer
complaints filed against Comcast.
#It will help to pin down what is wrong with Comcast's customer service.


# Installing Packages
install.packages('ggplot2')
install.packages('ggpubr')
install.packages('dplyr')
install.packages('lubridate')
install.packages('stringi')
install.packages('data.table')
install.packages('plyr')

# Adding Libraries
library(ggplot2)
library(ggpubr)
library(dplyr)
library(lubridate)
library(stringi)
library(data.table)
library(plyr)

# Analysis Task
# Task01-Import data into R environment.
comcast_data<- read.csv("D:\\Data Science\\R
Programming\\Projects\\Project2\\Comcast Telecom Complaints
data.csv",header=TRUE)
comcast_data

summary(comcast_data)
View(comcast_data)

# Extracting monthly and daily count
comcast_data$Date<-dmy(comcast_data$Date)
comcast_data$datemonth<-months(as.Date(comcast_data$Date))


comcast_data$datemonth

Months_counts <- table(comcast_data$datemonth)

#Months_counts
months_counts_df<-as.data.frame(Months_counts)
names(months_counts_df)[1]<-"Months"
names(months_counts_df)[2]<-"Count"
```

```
#months_counts_df
comcast_data$date_of_each_month<-day(as.Date(comcast_data$Date))

#comcast_data$date_of_each_month
Date_counts <- table(comcast_data$date_of_each_month)

#Date_counts
Date_counts_df<-as.data.frame(Date_counts)

names(Date_counts_df)[1]<-"date_of_each_month"
names(Date_counts_df)[2]<-"Count"
#Date_counts_df


#Task02-Provide the trend chart for the number of complaints at monthly and
daily granularity levels.

#Trend Chart for Monthly count
ggplot(data=months_counts_df, aes(x=Months, y=Count, label=Count, group=1))
+
  geom_line(color="green")+
  geom_point(size = 0.8)+geom_text()+
  labs(title = "Monthly Ticket Count",x= "Months",y ="No. of Tickets")

#Trend Chart for Daily count
ggplot(data=Date_counts_df, aes(x=date_of_each_month, y=Count,label=Count,
group=1)) +
  geom_line(color="blue")+
  geom_point(size = 0.8)+
  geom_text()+labs(title = "Daily Ticket Count",x= "Dates of Each Month",y
="No. of Tickets")

#Task03-Provide a table with the frequency of complaint types.
comcast_data$Customer.Complaint
complaints_count<-table(comcast_data$Customer.Complaint)


network_tickets<- data.frame(contains(comcast_data$Customer.Complaint,match
= 'network',ignore.case = T))
internet_tickets<-
data.frame(contains(comcast_data$Customer.Complaint,match =
'internet',ignore.case = T))
billing_tickets<- data.frame(contains(comcast_data$Customer.Complaint,match
= 'bill',ignore.case = T))
datacap_tickets<- data.frame(contains(comcast_data$Customer.Complaint,match
= 'data cap',ignore.case = T))
customerservice_tickets<-
data.frame(contains(comcast_data$Customer.Complaint,match = 'customer
service',ignore.case = T))

nrow(network_tickets)
nrow(internet_tickets)
nrow(billing_tickets)
nrow(datacap_tickets)
```

```r
nrow(customerservice_tickets)

#Task04-Which complaint types are maximum i.e., around internet, network
issues, or across any other domains.
#Create a new categorical variable with value as Open and Closed. Open &
Pending is to be categorized as Open and Closed & Solved is to be
categorized as Closed.
#Provide state wise status of complaints in a stacked bar chart. Use the
categorized variable from Q3. Provide insights on:


#Complaint Type that has maximum Tickets
if(nrow(network_tickets) > nrow(internet_tickets)){
  print("Network Issues")
} else if(nrow(internet_tickets) > nrow(billing_tickets)){
  print("Internet Issues")
} else if(nrow(billing_tickets) > nrow(datacap_tickets)){
  print("Billing Issues")
} else if(nrow(datacap_tickets) > nrow(customerservice_tickets)){
  print("Data cap Issues")
} else {
  print("Customer Service Issues")
}


#Open and Pending Statuses are considered as "Open"
my_data <- as_tibble(comcast_data$Status)
Open_complaints <- my_data %>% filter(value=="Open" | value=="Pending")
comcast_data<-subset(comcast_data,select=-c(ComplaintStatus))

comcast_data$Status<-gsub('Pending', 'Open', comcast_data$Status)

Open_complaints<-(comcast_data$Status == "Open")
comcast_data$Complaint_Status[Open_complaints]<-"Open"

#Solved and Closed Statuses are considered as "Closed"
comcast_data$Status<-gsub('Solved','Closed', comcast_data$Status)


Closed_complaints<-(comcast_data$Status == "Closed")
comcast_data$Complaint_Status[Closed_complaints]<-"Closed"


#Stacked Bar chart for Open and Closed Complaints
comcast_data<- group_by(comcast_data,State,Complaint_Status)
chart_data<- dplyr::summarise(comcast_data,Count = n())

ggplot(as.data.frame(chart_data) ,mapping = aes(State,Count))+
  geom_col(aes(fill = Complaint_Status),width = 0.95)+
  theme(axis.text.x = element_text(angle = 90),
        axis.title.y = element_text(size = 15),
        axis.title.x = element_text(size = 15),
        title = element_text(size = 16,colour = "#0073C2FF"),
        plot.title = element_text(hjust =  0.5))+
  labs(title = "Ticket Status Stacked Bar Chart",
       x = "States",y = "No of Tickets",
       fill= "Status")


#Task05-Which state has the maximum complaints
chart_data%>% filter(Complaint_Status=="Open")-> Open_complaints

max(Open_complaints$Count)
(Open_complaints)[1]
Open_complaints[Open_complaints$Count == max(Open_complaints$Count),c(1,3)]

#Complaints which were received through the Internet and customer care
calls.
#Task06-Which state has the highest percentage of unresolved complaints
#        -Provide the percentage of complaints resolved till date, which
were received through the
#        -Internet and customer care calls.

Resolved_data<-group_by(comcast_data,Complaint_Status)
Total_resolved<-dplyr::summarise(Resolved_data ,percentage =
(n()/nrow(Resolved_data)))
#Total_resolved
Resolved_data1 <- group_by(comcast_data,Received.Via,Complaint_Status)
Category_resloved<-dplyr::summarise(Resolved_data1,percentage =
(n()/nrow(Resolved_data)))
#Category_resloved


#Pie Chart for Category wise Ticket Status
#**************************************************

par(mfrow = c(1,2))
total<-ggplot(data=Total_resolved,
              aes(x= "",y =percentage,fill = Complaint_Status))+
  geom_bar(stat = "identity",width = 1)+
  coord_polar("y",start = 0)+
  geom_text(aes(label = paste0(round(percentage*100),"%")),
            position = position_stack(vjust = 0.5))+
  labs(title = "Pie Chart based on Ticket Status",x = NULL,y = NULL,fill =
NULL)+
  theme_classic()+theme(axis.line = element_blank(),
                        axis.text = element_blank(),
                        axis.ticks = element_blank())


category<-ggplot(data=Category_resloved,
                 aes(x= "",y =percentage,fill = Complaint_Status))+
  geom_bar(stat = "identity",width = 1)+
  coord_polar("y",start = 0)+
  geom_text(aes(label = paste0(Received.Via,"",round(percentage*100),"%")),
            position = position_stack(vjust = 0.5))+
  labs(title = "Pie Chart for Category wise Ticket Status",x = NULL,y =
NULL,fill = NULL)+
  theme_classic()+theme(axis.line = element_blank(),
                        axis.text = element_blank(),
                        axis.ticks = element_blank())

ggarrange(total,category,nrow = 1, ncol = 2)
```