**Pie Chart**

Syntax

The basic syntax for creating a pie-chart using the R is −

pie(x, labels, radius, main, col, clockwise)

Following is the description of the parameters used −

* **x** is a vector containing the numeric values used in the pie chart.
* **labels** is used to give description to the slices.
* **radius** indicates the radius of the circle of the pie chart.(value between −1 and +1).
* **main** indicates the title of the chart.
* **col** indicates the color palette.
* **clockwise** is a logical value indicating if the slices are drawn clockwise or anti clockwise.

Example

A very simple pie-chart is created using just the input vector and labels. The below script will create and save the pie chart in the current R working directory.

# Create data for the graph.

x <- c(21, 62, 10, 53)

labels <- c("London", "New York", "Singapore", "Mumbai")

# Give the chart file a name.

png(file = "city.jpg")

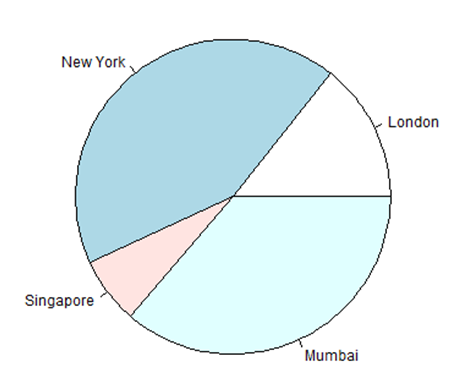
# Plot the chart.

pie(x,labels)

# Save the file.

dev.off()

When we execute the above code, it produces the following result −



Pie Chart Title and Colors

We can expand the features of the chart by adding more parameters to the function. We will use parameter **main** to add a title to the chart and another parameter is **col** which will make use of rainbow colour pallet while drawing the chart. The length of the pallet should be same as the number of values we have for the chart. Hence we use length(x).

Example

The below script will create and save the pie chart in the current R working directory.

# Create data for the graph.

x <- c(21, 62, 10, 53)

labels <- c("London", "New York", "Singapore", "Mumbai")

# Give the chart file a name.

png(file = "city\_title\_colours.jpg")

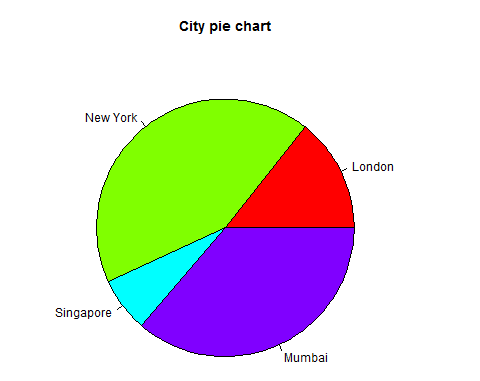
# Plot the chart with title and rainbow color pallet.

pie(x, labels, main = "City pie chart", col = rainbow(length(x)))

# Save the file.

dev.off()

When we execute the above code, it produces the following result −



Slice Percentages and Chart Legend

We can add slice percentage and a chart legend by creating additional chart variables.

# Create data for the graph.

x <- c(21, 62, 10,53)

labels <- c("London","New York","Singapore","Mumbai")

piepercent<- round(100\*x/sum(x), 1)

# Give the chart file a name.

png(file = "city\_percentage\_legends.jpg")

# Plot the chart.

pie(x, labels = piepercent, main = "City pie chart",col = rainbow(length(x)))

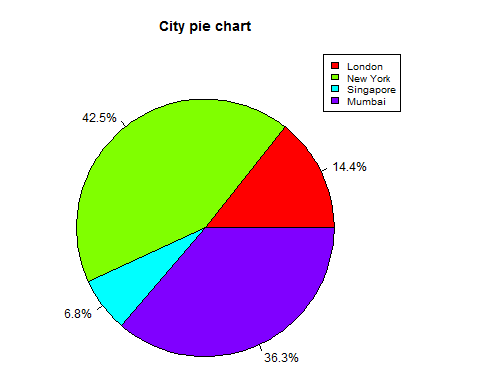
legend("topright", c("London","New York","Singapore","Mumbai"), cex = 0.8,

fill = rainbow(length(x)))

# Save the file.

dev.off()

When we execute the above code, it produces the following result −



3D Pie Chart

A pie chart with 3 dimensions can be drawn using additional packages. The package **plotrix** has a function called **pie3D()** that is used for this.

# Get the library.

library(plotrix)

# Create data for the graph.

x <- c(21, 62, 10,53)

lbl <- c("London","New York","Singapore","Mumbai")

# Give the chart file a name.

png(file = "3d\_pie\_chart.jpg")

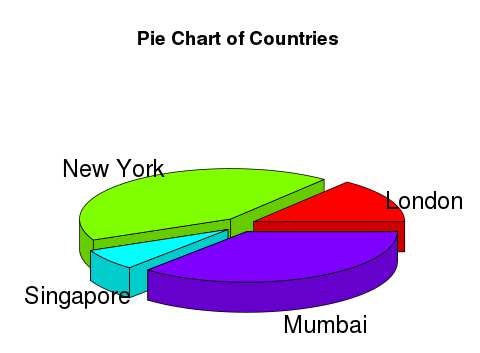
# Plot the chart.

pie3D(x,labels = lbl,explode = 0.1, main = "Pie Chart of Countries ")

# Save the file.

dev.off()

When we execute the above code, it produces the following result −



Bar Chart

Syntax

The basic syntax to create a bar-chart in R is −

barplot(H, xlab, ylab, main, names.arg, col)

Following is the description of the parameters used −

* **H** is a vector or matrix containing numeric values used in bar chart.
* **xlab** is the label for x axis.
* **ylab** is the label for y axis.
* **main** is the title of the bar chart.
* **names.arg** is a vector of names appearing under each bar.
* **col** is used to give colors to the bars in the graph.

Example

A simple bar chart is created using just the input vector and the name of each bar.

The below script will create and save the bar chart in the current R working directory.

# Create the data for the chart.

H <- c(7,12,28,3,41)

# Give the chart file a name.

png(file = "barchart.png")

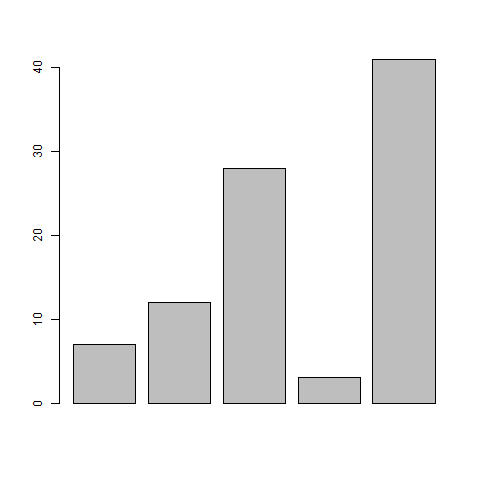
# Plot the bar chart.

barplot(H)

# Save the file.

dev.off()

When we execute the above code, it produces the following result −



Bar Chart Labels, Title and Colors

The features of the bar chart can be expanded by adding more parameters. The **main** parameter is used to add **title**. The **col** parameter is used to add colors to the bars. The **args.name** is a vector having same number of values as the input vector to describe the meaning of each bar.

Example

The following script will create and save the bar chart in the current R working directory.

# Create the data for the chart.

H <- c(7,12,28,3,41)

M <- c("Mar","Apr","May","Jun","Jul")

# Give the chart file a name.

png(file = "barchart\_months\_revenue.png")

# Plot the bar chart.

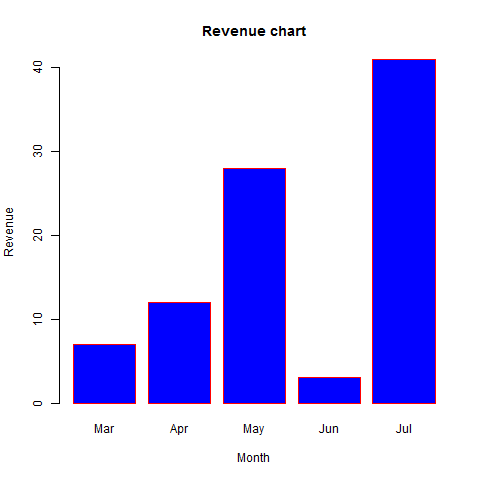
barplot(H,names.arg = M,xlab = "Month",ylab = "Revenue",col = "blue",

main = "Revenue chart",border = "red")

# Save the file.

dev.off()

When we execute the above code, it produces the following result −



Group Bar Chart and Stacked Bar Chart

We can create bar chart with groups of bars and stacks in each bar by using a matrix as input values.

More than two variables are represented as a matrix which is used to create the group bar chart and stacked bar chart.

# Create the input vectors.

colors <- c("green","orange","brown")

months <- c("Mar","Apr","May","Jun","Jul")

regions <- c("East","West","North")

# Create the matrix of the values.

Values <- matrix(c(2,9,3,11,9,4,8,7,3,12,5,2,8,10,11),nrow = 3,ncol = 5,byrow = TRUE)

# Give the chart file a name.

png(file = "barchart\_stacked.png")

# Create the bar chart.

barplot(Values,main = "total revenue",names.arg = months,xlab = "month",ylab = "revenue",

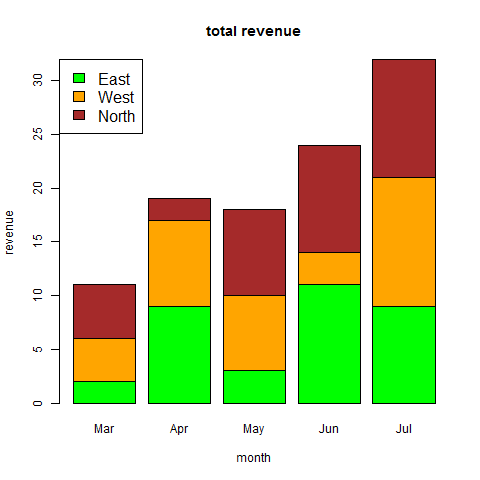
col = colors)

# Add the legend to the chart.

legend("topleft", regions, cex = 1.3, fill = colors)

# Save the file.

dev.off()



Line Graphs

Syntax

The basic syntax to create a line chart in R is −

plot(v,type,col,xlab,ylab)

Following is the description of the parameters used −

* **v** is a vector containing the numeric values.
* **type** takes the value "p" to draw only the points, "l" to draw only the lines and "o" to draw both points and lines.
* **xlab** is the label for x axis.
* **ylab** is the label for y axis.
* **main** is the Title of the chart.
* **col** is used to give colors to both the points and lines.

Example

A simple line chart is created using the input vector and the type parameter as "O". The below script will create and save a line chart in the current R working directory.

# Create the data for the chart.

v <- c(7,12,28,3,41)

# Give the chart file a name.

png(file = "line\_chart.jpg")

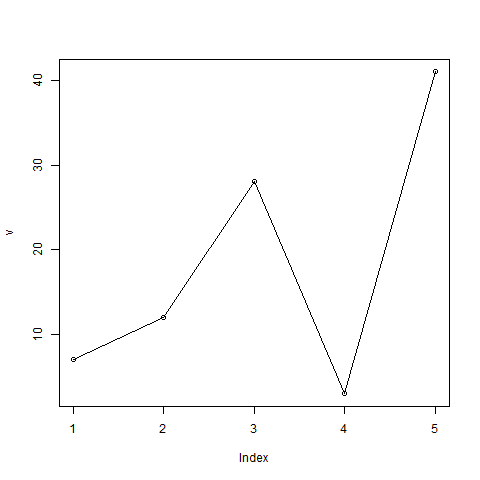
# Plot the bar chart.

plot(v,type = "o")

# Save the file.

dev.off()

When we execute the above code, it produces the following result −



Line Chart Title, Color and Labels

The features of the line chart can be expanded by using additional parameters. We add color to the points and lines, give a title to the chart and add labels to the axes.

Example

# Create the data for the chart.

v <- c(7,12,28,3,41)

# Give the chart file a name.

png(file = "line\_chart\_label\_colored.jpg")

# Plot the bar chart.

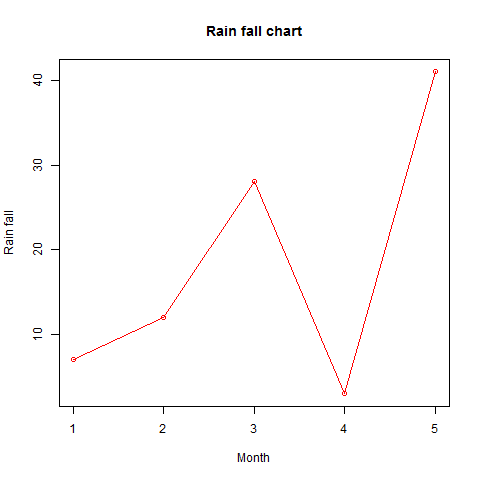
plot(v,type = "o", col = "red", xlab = "Month", ylab = "Rain fall",

main = "Rain fall chart")

# Save the file.

dev.off()

When we execute the above code, it produces the following result −



Multiple Lines in a Line Chart

More than one line can be drawn on the same chart by using the **lines()**function.

After the first line is plotted, the lines() function can use an additional vector as input to draw the second line in the chart,

# Create the data for the chart.

v <- c(7,12,28,3,41)

t <- c(14,7,6,19,3)

# Give the chart file a name.

png(file = "line\_chart\_2\_lines.jpg")

# Plot the bar chart.

plot(v,type = "o",col = "red", xlab = "Month", ylab = "Rain fall",

main = "Rain fall chart")

lines(t, type = "o", col = "blue")

# Save the file.

dev.off()

When we execute the above code, it produces the following result −

