graph modifying functions

This section covers the following functions:

1. points()
2. lines()
3. text()
4. abline()
5. legend()

1. `points()`

◆ Purpose

The `points()` function in R is used to **add points to an existing plot**, typically used to overlay additional data on top of a base plot.

📦 Package

Base R (`graphics` package)

📄 Function Header

```
points(x, y = NULL, type = "p", pch = 1, col = par("col"), bg = NA,
       cex = 1, lwd = 1, lty = par("lty"), ...)
```

🔧 Parameters

Argument	Description	Accepted Values / Data Types
x	x-coordinates of points	Numeric vector or list
y	y-coordinates of points (optional if `x` is a list or matrix)	Numeric vector

| Argument	Description	Accepted Values / Data Types
type	Type of plot symbols/lines	`"p"` (points), `"l"` (lines), `"o"` (overplotted), `"b"` (both), `"c"` (lines excluding points), `"s"`, `"S"`, `"h"` (high-density), `"n"` (none)
pch	Plotting character or symbol type	Integers (0–25), or single characters like `"*"`
col	Color of points	Any valid R color name or hex code (e.g., `"red"`, `"#FF5733"`), see `colors()`
bg	Background color (for `pch` values 21–25)	Color name or hex code
cex	Point size (relative to default)	Numeric (e.g., `1.5` makes points 1.5× larger)
lwd	Line width (used with `type = "l"` or `type = "b"`)	Numeric
lty	Line type	`0` (blank), `1` (solid), `2` (dashed), `3` (dotted), `4` (dotdash), `5` (longdash), `6` (twodash)
...	Additional graphical parameters	See graphical parameters table below

🎨 Graphical Parameters

Parameter	Description	Accepted Values
xlim , ylim	Axis limits	Numeric vectors (length = 2)
main , sub	Main & subtitle text	Character strings
xlab , ylab	Axis labels	Character strings
asp	Aspect ratio	Numeric
axes	Whether to draw axes	Logical (`TRUE` / `FALSE`)
frame.plot	Whether to draw a box	Logical
ann	Annotate with axis labels & title	Logical

💡 Example Use Cases

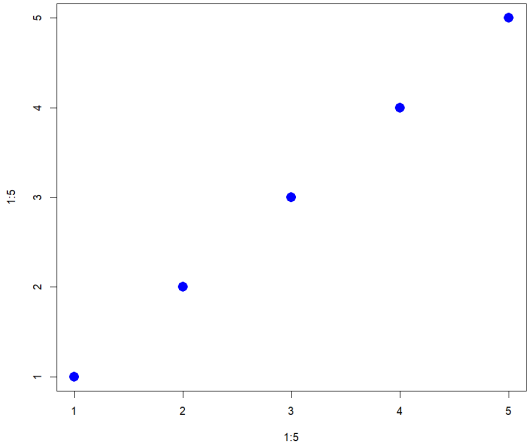### ➤ Basic Use

```
plot(1:5, 1:5, type = "n")   # Empty plot
points(1:5, 1:5)             # Adds points at coordinates
```

```r
plot(x, y, pch = 1)
points(x, y + 0.5, pch = 2, col = "red")
```
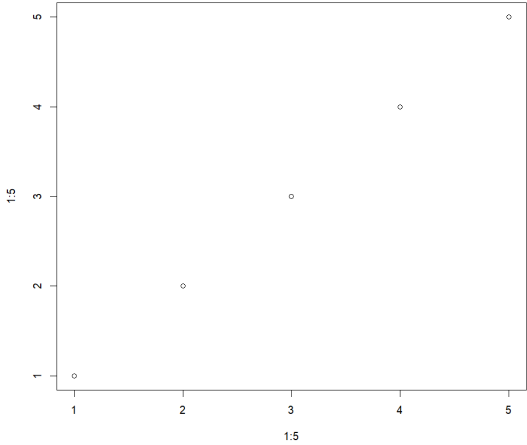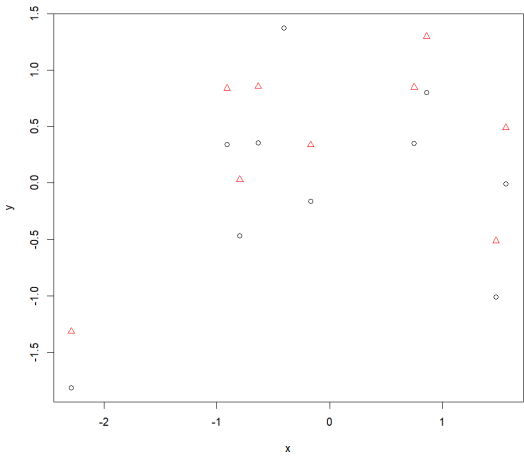

➤ Custom Symbols and Color

```r
plot(1:5, 1:5, type = "n")
points(1:5, 1:5, pch = 16, col = "blue", cex = 2)
```


➤ Adding Points to Existing Scatter Plot

```r
x <- rnorm(10)
y <- rnorm(10)
```

2. `lines()`

◆ Purpose

The `lines()` function adds **line segments** to an existing plot. It is commonly used to draw lines through or between points already plotted or to overlay new data series.

📦 Package

Base R (`graphics` package)

📑 Function Header

```r
lines(x, y = NULL, type = "l", lty = par("lty"), lwd = par("lwd"),
      col = par("col"), ...)
```

🔧 Parameters

Argument	Description	Accepted Values / Data Types
x	x-coordinates	Numeric vector or list (or time/date object)

| Argument	Description	Accepted Values / Data Types
`y`	y-coordinates (optional if `x` is a list or matrix)	Numeric vector
`type`	Type of plot	`"l"` (default; lines), `"s"`, `"S"`, `"h"`, `"p"`, `"o"`, `"b"`, `"c"` (like in `plot()`)
`lty`	Line type	`0` (blank), `1` (solid), `2` (dashed), `3` (dotted), `4` (dotdash), `5` (longdash), `6` (twodash)
`lwd`	Line width	Numeric (e.g., `2` for twice the default width)
`col`	Line color	Any valid R color name or hex code
`...`	Additional graphical parameters	See graphical parameters table below

🎨 Graphical Parameters

Parameter	Description	Accepted Values
`xlim`, `ylim`	Axis limits	Numeric vectors (length = 2)
`main`, `sub`	Main & subtitle text	Character strings
`xlab`, `ylab`	Axis labels	Character strings
`asp`	Aspect ratio	Numeric
`axes`	Whether to draw axes	Logical (`TRUE` / `FALSE`)
`frame.plot`	Whether to draw a box	Logical
`ann`	Annotate with axis labels & title	Logical

💡 Example Use Cases
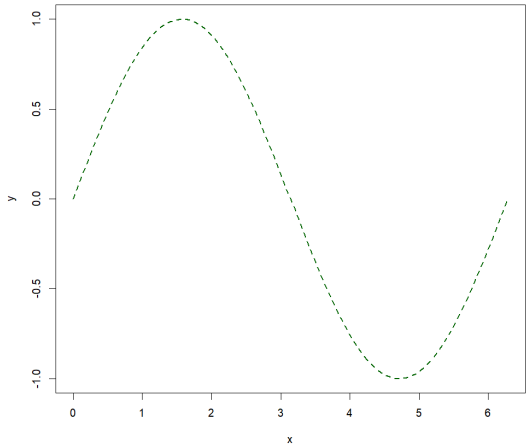
➤ Adding a Line to a Plot

```
x <- 1:10
y <- x^2
plot(x, y, type = "n")   # Empty plot with correct limits
lines(x, y)              # Draws a curve
```
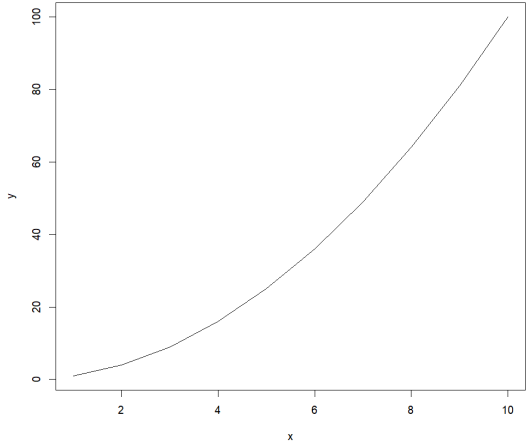

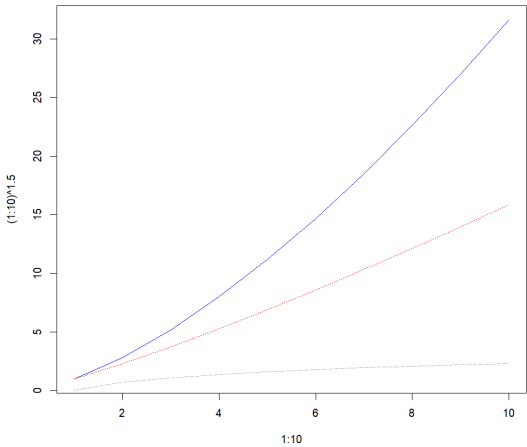### ➤ Styled Line with Width and Color

```
x <- seq(0, 2*pi, length.out = 100)
y <- sin(x)
plot(x, y, type = "n")
lines(x, y, col = "darkgreen", lty = 2, lwd = 2)
```


➤ Overlay Multiple Lines

```
plot(1:10, (1:10)^1.5, type = "l", col = "blue")
lines(1:10, (1:10)^1.2, col = "red", lty = 3)
```

```
lines(1:10, log(1:10), col = "darkgray", lty = 4)
```


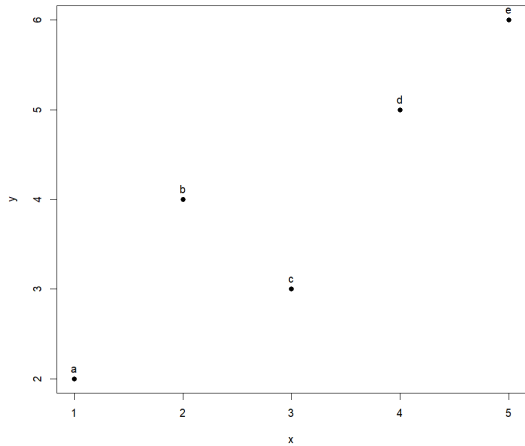| Argument	Description	Accepted Values / Data Types
`labels`	Text labels to be placed	Character vector or expression
`adj`	Adjustment of text (x, y justification)	Numeric vector of length 1 or 2 (0 = left/bottom, 1 = right/top)
`pos`	Positioning relative to coordinates	`1` (below), `2` (left), `3` (above), `4` (right); overrides `adj`
`offset`	Distance of text from coordinates (used with `pos`)	Numeric (e.g., `0.5`)
`vfont`	Vector font for text	Character vector of length 2: e.g., `c("sans serif", "plain")`
`cex`	Text size (relative to default)	Numeric
`col`	Text color	Color name or hex code
`font`	Font face/style	`1` (plain), `2` (bold), `3` (italic), `4` (bold italic), `5` (symbol)
`...`	Additional graphical parameters	See graphical parameters table below

🎨 Graphical Parameters

Parameter	Description	Accepted Values
`xpd`	Clipping of text outside plot region	Logical (`TRUE` , `FALSE`)
`family`	Font family	`"serif"`, `"sans"`, `"mono"`, etc.
`srt`	Text rotation in degrees	Numeric (e.g., `45`)
`xlim`, `ylim`	Axis limits	Numeric vectors (length = 2)
`main`, `sub`	Main & subtitle text	Character strings
`xlab`, `ylab`	Axis labels	Character strings

💡 Example Use Cases

### ➤ Labeling Points in a Scatter Plot

```
x <- 1:5
y <- c(2, 4, 3, 5, 6)
plot(x, y, pch = 19)
text(x, y, labels = letters[1:5], pos = 3)
```

3. `text()`

◆ Purpose

The `text()` function adds **text labels** to an existing plot. It is typically used for annotating points, labeling axes, or placing custom text at specific coordinates.

📦 Package

Base R (`graphics` package)

📃 Function Header

```
text(x, y = NULL, labels = seq_along(x), adj = NULL, pos = NULL,
     offset = 0.5, vfont = NULL, cex = 1, col = NULL, font = NULL, ...)
```

🔧 Parameters

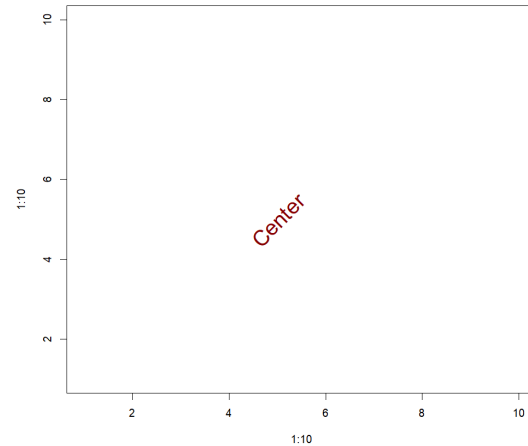| Argument	Description	Accepted Values / Data Types
x	x-coordinates of text	Numeric vector
y	y-coordinates of text	Numeric vector

➤ Custom Font, Size, and Color

```
plot(1, 1, type = "n", xlim = c(0, 2), ylim = c(0, 2))
text(1, 1, "Hello, world!", col = "blue", cex = 1.5, font = 2)
```
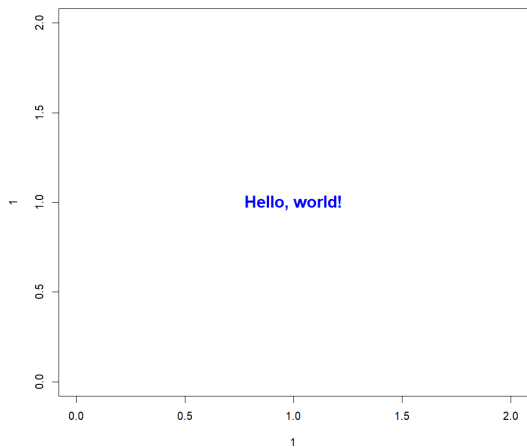

➤ Rotated and Positioned Text

```
plot(1:10, 1:10, type = "n")
text(5, 5, "Center", srt = 45, col = "darkred", cex = 2)
```

4. `abline()`

◆ Purpose

The `abline()` function is used to **add straight lines** to an existing plot. It can draw:

- Regression lines (`y = a + b*x`)
- Horizontal lines (`h = y`)
- Vertical lines (`v = x`)

📦 Package

Base R (`graphics` package)

📄 Function Header

```
abline(a = NULL, b = NULL, h = NULL, v = NULL,
       reg = NULL, col = par("col"), lty = par("lty"),
       lwd = par("lwd"), ...)
```

🔧 Parameters

Argument	Description	Accepted Values / Data Types
`a`	Intercept of the line (`y = a + b*x`)	Numeric
`b`	Slope of the line (`y = a + b*x`)	Numeric
`h`	Horizontal line(s) at y	Numeric or numeric vector
`v`	Vertical line(s) at x	Numeric or numeric vector
`reg`	A model object (e.g., from `lm()`)	Object of class `lm`
`col`	Line color	Color name or hex code
`lty`	Line type	`0` (blank), `1` (solid), `2` (dashed), `3` (dotted), `4` (dotdash), `5` (longdash), `6` (twodash)
`lwd`	Line width	Numeric
`...`	Additional graphical parameters	See below

**Note:** You must specify one of `a` and `b`, `h`, `v`, or `reg`. These are mutually exclusive options.

🎨 Graphical Parameters

| Parameter	Description	Accepted Values
`xpd`	Clipping of line outside plot	Logical
`xlim`, `ylim`	Axis limits	Numeric vectors (length = 2)
`main`, `sub`	Main & subtitle text	Character strings
`asp`	Aspect ratio	Numeric
`xaxs`, `yaxs`	Axis interval expansion	`"r"` (regular), `"i"` (internal)

💡 Example Use Cases

➤ Add Regression Line (`y = a + b*x`)

```
plot(1:10, (1:10)^1.2)
abline(a = 0, b = 1, col = "blue", lty = 2)
```
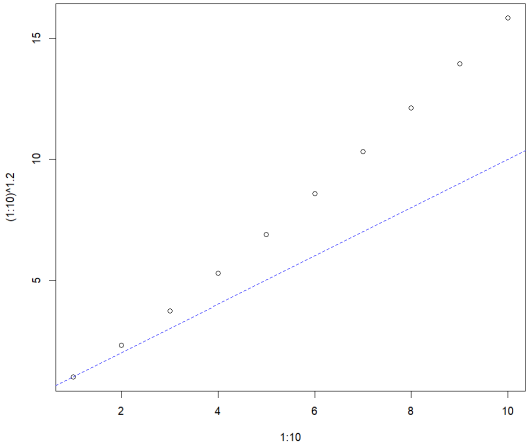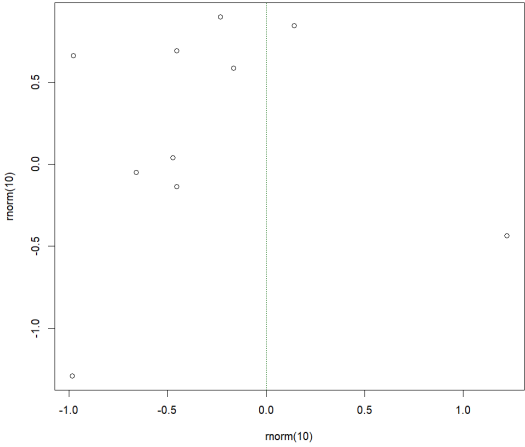

➤ Add Horizontal Line at y = 5

```
plot(1:10, rnorm(10))
abline(h = 0, col = "red", lwd = 2)
```
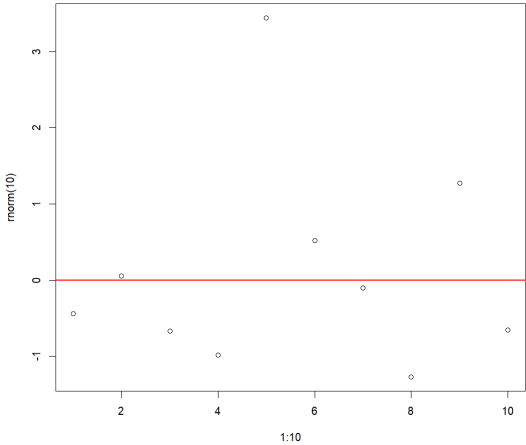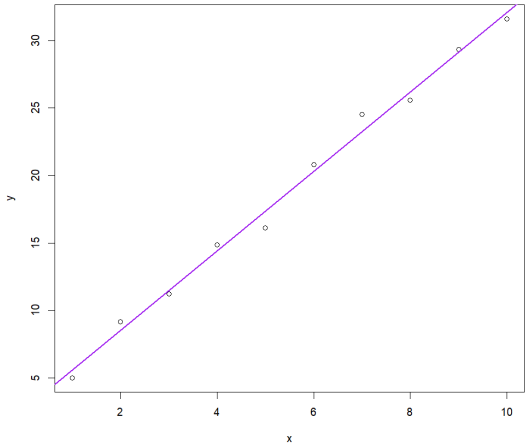

➤ Add Vertical Line at x = 5

```
plot(rnorm(10), rnorm(10))
abline(v = 0, col = "darkgreen", lty = 3)
```

➤ Add Regression Line from `lm` Model

```r
x <- 1:10
y <- 2 + 3*x + rnorm(10)
fit <- lm(y ~ x)
plot(x, y)
abline(fit, col = "purple", lwd = 2)
```


5. `legend()`

◆ Purpose

The `legend()` function adds a **legend** to an existing plot. It is typically used to label different data series or graphical elements like points, lines, and polygons, helping to clarify which colors or symbols represent which data.

📦 Package

Base R (`graphics` package)

📄 Function Header

```r
legend(x, y = NULL, legend, fill = NULL, col = NULL, border = NULL,
       lty = NULL, lwd = NULL, pch = NULL, pt.cex = NULL, pt.bg = NULL,
       text.col = NULL, text.font = NULL, box.col = NULL, box.lty = NULL,
       box.lwd = NULL, bg = NULL, title = NULL, title.col = NULL, ...)
```

🔧 Parameters

Argument	Description	Accepted Values / Data Types
x	x-coordinate or position for the legend box	Numeric, character (`"topright"`, `"bottomleft"`, etc.)
y	y-coordinate for the legend box (optional if `x` is a position string)	Numeric
legend	Character vector with legend labels	Character vector
fill	Fill colors for legend items (e.g., for rectangles or boxes)	Color names or hex codes
col	Line or point colors	Color names or hex codes
border	Border color for symbols or boxes in the legend	Color names or hex codes
lty	Line types for legend items	`0` (blank), `1` (solid), `2` (dashed), `3` (dotted), `4` (dotdash), `5` (longdash), `6` (twodash)
lwd	Line width for legend items	Numeric
pch	Plotting characters (symbols) for legend items	Numeric or character (e.g., `16` for filled circle)
pt.cex	Symbol size (relative to default)	Numeric
pt.bg	Background color of points in the legend	Color names or hex codes
text.col	Text color	Color names or hex codes

| Argument	Description	Accepted Values / Data Types
`text.font`	Font style for legend text	Numeric (`1` = plain, `2` = bold, `3` = italic, `4` = bold italic)
`box.col`	Background color of legend box	Color names or hex codes
`box.lty`	Line type of the legend box	`0` (no box), `1` (solid), etc.
`box.lwd`	Line width of the legend box	Numeric
`bg`	Background color of the legend area	Color names or hex codes
`title`	Title text for the legend	Character string
`title.col`	Title text color	Color names or hex codes
`...`	Additional graphical parameters	See below

🎨 Graphical Parameters

| Parameter	Description	Accepted Values
`xpd`	Clipping behavior	Logical
`xlim`, `ylim`	Axis limits	Numeric vectors (length = 2)
`asp`	Aspect ratio	Numeric
`main`, `sub`	Main & subtitle text	Character strings
`xlab`, `ylab`	Axis labels	Character strings

💡 Example Use Cases

➤ Simple Legend

```
plot(1:10, rnorm(10), col = "red", pch = 16)
legend("topright", legend = c("Data points"), col = "red", pch = 16)
```
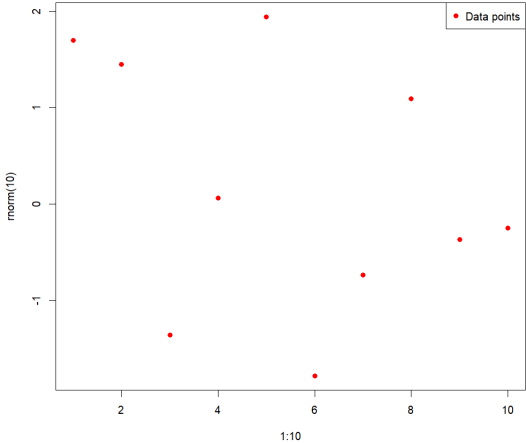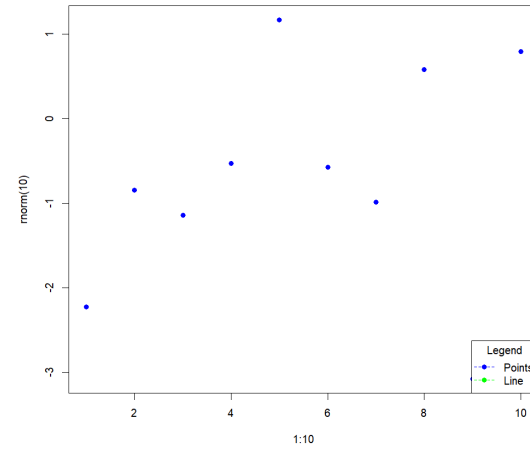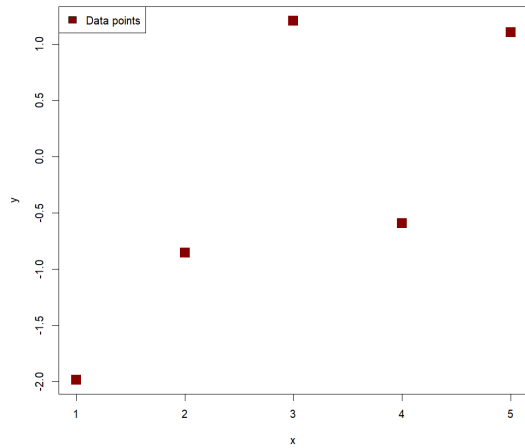

➤ Legend with Lines and Points

```
x <- 1:5
y1 <- x^2
y2 <- x^3
plot(x, y1, type = "l", col = "blue", lwd = 2)
lines(x, y2, col = "green", lwd = 2)
legend("topright", legend = c("y = x^2", "y = x^3"), col = c("blue", "green"), lwd = 2)
```
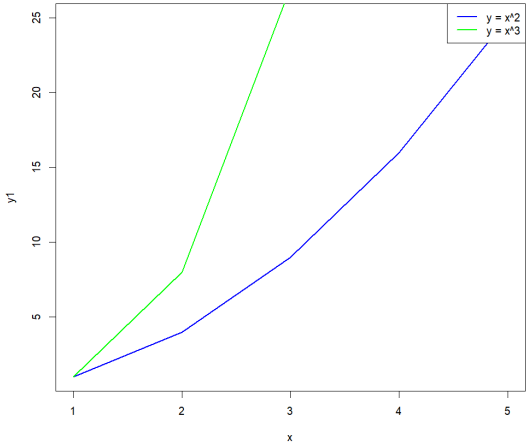

➤ Customized Legend with Fill Color

```r
x <- 1:5
y <- rnorm(5)
plot(x, y, pch = 15, col = "darkred", cex = 2)
legend("topleft", legend = c("Data points"), fill = "darkred", border = "black")
```


➤ Multi-Element Legend with Titles

```r
plot(1:10, rnorm(10), col = "blue", pch = 16)
lines(1:10, 2 + 1.5*(1:10), col = "green", lty = 2)
legend("bottomright", legend = c("Points", "Line"), col = c("blue", "green"), pch = 16,
lty = 2, title = "Legend")
```