Hands - on start to wolfram Mathematica

Second Edition

By Cliff Hastings, Kelvin Mischo and Micheal Morrison

Jana Rasras
June 2023

chapter 1| Introduction

```
In[•]:= 711 / 3
Out[•]= 237

In[•]:= 718 / 3
Out[•]= 718/3

        N[718 / 3, 5] (*Approx answer to 718 divided by 3 rounded to 5 digits*)

        239.33333333333333333`5.

In[•]:= 718/3
        718 / 3
Out[•]= 718/3

In[•]:= a = 5 (* assign a value to a name*)
Out[•]= 5

In[•]:= 3 a + 1
Out[•]= 16

In[•]:= Clear[a] (*clear the varible defention of a, which will make a undefined*)
```

```
Expand[(a + 5) (a + 9)] (*Expand the algebric experssion*)
```

Out[•]= $45 + 14\,a + a^2$

```
In[•]:= Solve[2 x - 7 == 0]
```

Out[•]= $\left\{\left\{x \to \frac{7}{2}\right\}\right\}$

In[•]:=

solve[2x−7]

Result

```
Reduce[-7 + 2*x == 0, x]
```

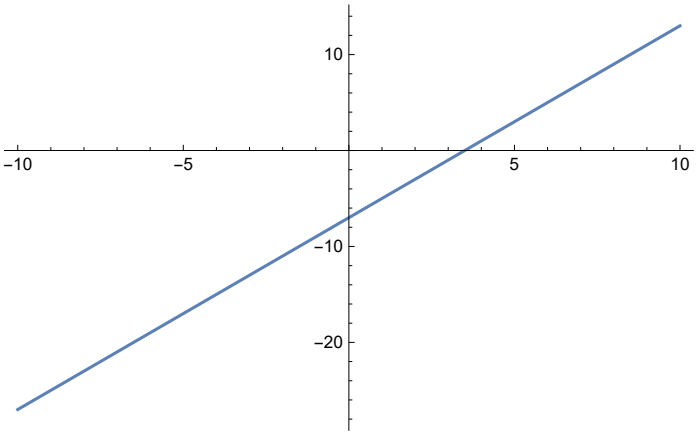
Out[•]= $x == \frac{7}{2}$

```
In[•]:= Solve[{2 x - 7 == 0, 3 x - 2 y == 0}, {x, y}]
```

Out[•]= $\left\{\left\{x \to \frac{7}{2}, y \to \frac{21}{4}\right\}\right\}$

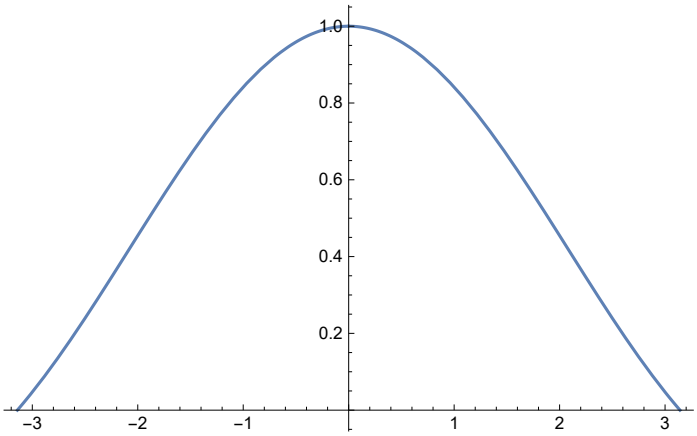
```
Plot[2 x - 7, {x, -10, 10}] (* plOT the equation, where x goes from -10 to 10*)
```

Out[•]=




```
In[•]:= Plot[Sin[x] / x, {x, -Pi, Pi}]
```

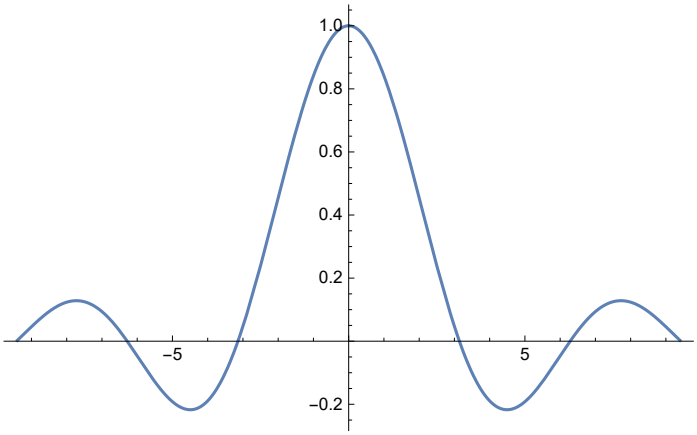
Out[•]=



In[ ]:=
plot sin(x) / x

```
Plot[Sin[x]/x, {x, -9.4, 9.4}]
```

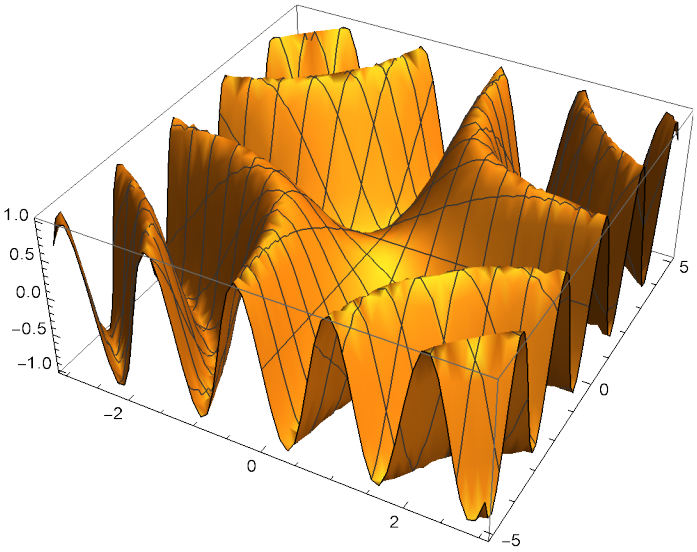
Out[ ]=



In[ ]:=

```
Plot3D[Sin[x*y], {x, -3, 3}, {y, -5, 5}]
```

Out[ ]=



In[ ]:=

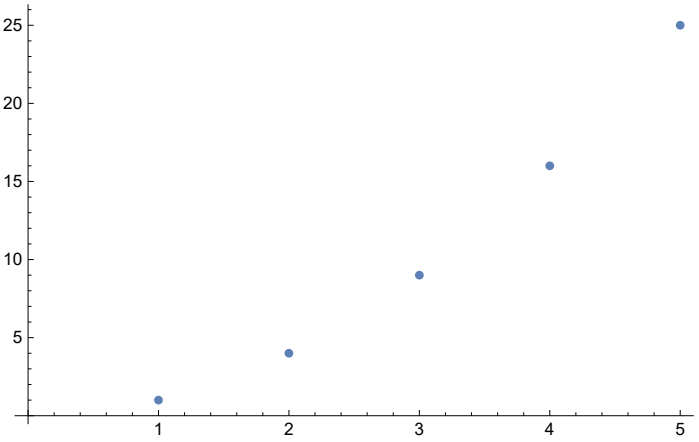
```
Table[i^2, {i, 1, 5}]
```

Out[ ]=

```
{1, 4, 9, 16, 25}
```

In[•]:=
```
ListPlot[ Table[i^2, {i, 1, 5}]]
```

Out[•]=




```
Integrate[Sin[x], x] (*Integrate with respect to x*)
```

Out[•]=
```
-Cos[x]
```

```
mat1 = {{1, 2, 3}, {3, 5, 7}, {4, 6, 8}};
```

In[•]:=
```
Det[{{1, 2, 3}, {3, 5, 7}, {4, 6, 8}}]
```

Out[•]=
```
0
```

In[•]:=
```
Clear[mat1]
```

Chapter 2| A sample Project in Mathematica

Out[•]=
```
78.9 yr
```

```
data = DeleteCases[Table[{i, CountryData[i, "LifeExpectancy"]}, {i, CountryData[All]}],
   {_, _Missing}]; (*dELETE missing values*)
```

In[•]:= **Short[data]**

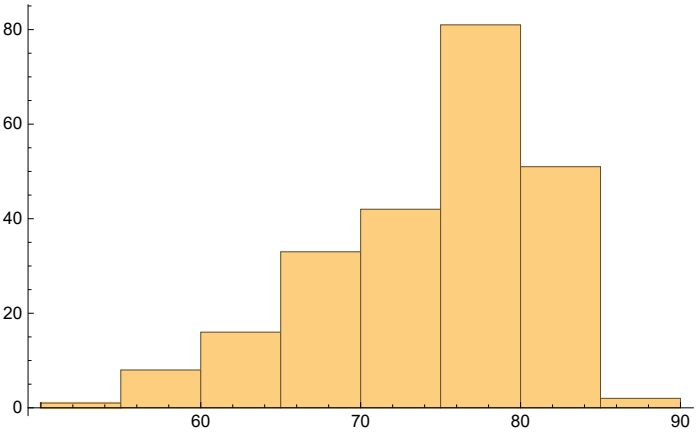
Out[•]//Short= {{Afghanistan, 53.25 yr}, {Albania, 79.23 yr}, {Algeria, 77.79 yr},
{American Samoa, 75.06 yr}, {Andorra, 83.23 yr}, {Angola, 61.71 yr},
{Anguilla, 82 yr}, {Antigua and Barbuda, 77.55 yr}, {Argentina, 78.07 yr},
{Armenia, 75.86 yr}, {Aruba, 77.76 yr}, {Australia, 82.89 yr},
{Austria, 82.07 yr}, {Azerbaijan, 73.88 yr}, {Bahamas, 75.87 yr},
{Bahrain, 79.67 yr}, {Bangladesh, 74.43 yr}, {Barbados, 78.31 yr},
{Belarus, 74.01 yr}, {Belgium, 81.65 yr}, {Belize, 75.56 yr},
{Benin, 61.82 yr}, {Bermuda, 81.83 yr}, {Bhutan, 71.5 yr},
{Bolivia, 70.7 yr}, {Bosnia and Herzegovina, 77.74 yr}, {Botswana, 65.24 yr},
{Brazil, 74.98 yr}, {British Virgin Islands, 79.44 yr}, {Brunei, 78.14 yr},
{Bulgaria, 75.3 yr}, <<173>>, {Syria, 74.01 yr}, {Taiwan, 80.95 yr},
{Tajikistan, 69.06 yr}, {Tanzania, 69.9 yr}, {Thailand, 77.41 yr},
{Togo, 70.99 yr}, {Tonga, 77.29 yr}, {Trinidad and Tobago, 74.92 yr},
{Tunisia, 76.57 yr}, {Turkey, 75.96 yr}, {Turkmenistan, 71.54 yr},
{Turks and Caicos Islands, 80.6 yr}, {Tuvalu, 68.07 yr}, {Uganda, 68.58 yr},
{Ukraine, 73.18 yr}, {United Arab Emirates, 79.37 yr}, {United Kingdom, 81.3 yr},
{United States, 80.43 yr}, {United States Virgin Islands, 80.05 yr},
{Uruguay, 78.19 yr}, {Uzbekistan, 75.03 yr}, {Vanuatu, 74.87 yr},
{Venezuela, 72.22 yr}, {Vietnam, 75.25 yr}, {Wallis and Futuna Islands, 80.45 yr},
{West Bank, 76.12 yr}, {Western Sahara, 63.8 yr},
{Yemen, 67.18 yr}, {Zambia, 65.92 yr}, {Zimbabwe, 62.83 yr}}

```
Short[SortBy[data, Last]] (*sort from least to greatest*)
(*Short  represent only the results*)
```

Out[•]//Short= {{Afghanistan, 53.25 yr}, {Central African Republic, 55.07 yr},
<<1>>
{{Somalia, 55.32 yr}, {Mozambique, 56.49 yr}, {South Sudan, 58.6 yr},
<<1>>
{{Chad, 58.73 yr}, {Lesotho, 58.9 yr}, {Eswatini, 59.13 yr},
<<1>>
{{Niger, 59.7 yr}, {Sierra Leone, 60.19 yr}, {Nigeria, 60.87 yr},
<<1>>
{{Democratic Republic of the Congo, 61.43 yr}, {Republic of the Congo, 61.69 yr},
<<1>>
{{Angola, 61.71 yr}, {Ivory Coast, 61.8 yr}, {Benin, 61.82 yr}, {Mali, 62.01 yr},
<<1>>
{{Cameroon, 62.79 yr}, {Zimbabwe, 62.83 yr}, {Burkina Faso, 63.06 yr},
<<1>>
{{Guinea-Bissau, 63.26 yr}, {Guinea, 63.53 yr}, {Western Sahara, 63.8 yr},
<<1>>
{{Senegal, 63.83 yr}, {Mauritania, 64.86 yr}, {Djibouti, 65 yr},
<<1>>
{{South Africa, 65.04 yr}, {Liberia, 65.1 yr}, {Botswana, 65.24 yr},
<<1>>
{{Rwanda, 65.48 yr}, {Haiti, 65.61 yr}, <<173>>, {Bermuda, 81.83 yr},
<<1>>
{{Cayman Islands, 81.84 yr}, {Isle of Man, 81.84 yr}, {Netherlands, 81.95 yr},
<<1>>
{{Anguilla, 82 yr}, {Austria, 82.07 yr}, {Spain, 82.21 yr},
<<1>>
{{New Zealand, 82.33 yr}, {Norway, 82.35 yr}, {Liechtenstein, 82.36 yr},
<<1>>
{{France, 82.39 yr}, {Jersey, 82.43 yr}, {Sweden, 82.6 yr}, {Italy, 82.67 yr},
<<1>>
{{Luxembourg, 82.78 yr}, {South Korea, 82.78 yr}, {Australia, 82.89 yr},
<<1>>
{{Malta, 83 yr}, {Guernsey, 83.03 yr}, {Switzerland, 83.03 yr}, {Israel, 83.15 yr},
<<1>>
{{Andorra, 83.23 yr}, {Hong Kong, 83.41 yr}, {Iceland, 83.45 yr},
<<1>>
{{Canada, 83.62 yr}, {San Marino, 83.68 yr}, {Japan, 84.65 yr},
<<1>>
{{Macau, 84.81 yr}, {Singapore, 86.19 yr}, {Monaco, 89.4 yr}}

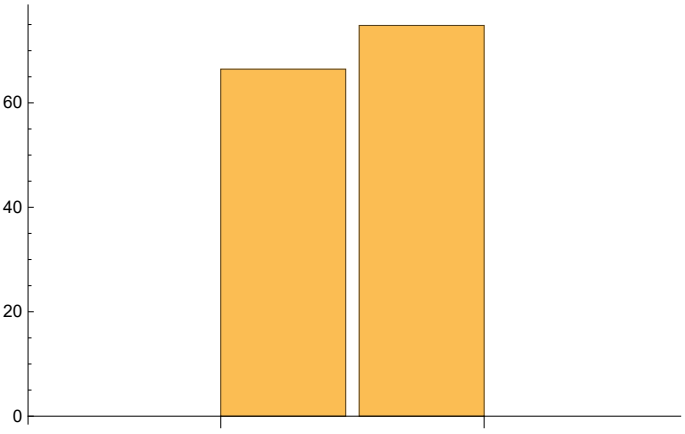
```
Histogram[data[[All, 2]] ] (* ESC [[ ESC*)
```

Out[•]=




```
In[•]:= dataAfrica = DeleteCases[Table[{i, CountryData[i, "LifeExpectancy"]},
          {i, CountryData["Africa"]}], {_, _Missing}];

In[•]:= Mean[dataAfrica[[All, 2]]]

Out[•]= 66.4587 yr

In[•]:= dataAsia = DeleteCases[Table[
          {i, CountryData[i, "LifeExpectancy"]}, {i, CountryData["Asia"]}], {_, _Missing}];

In[•]:= Mean[dataAsia[[All, 2]]]

Out[•]= 74.8257 yr

In[•]:= BarChart[{Mean[dataAfrica[[All, 2]]], Mean[dataAsia[[All, 2]]]}]
```

Out[•]=



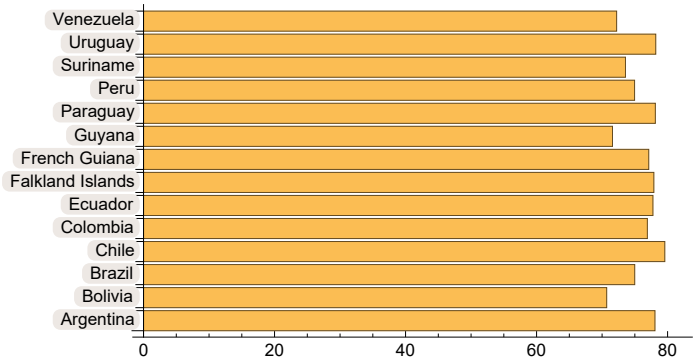

```
In[•]:= dataSouthAmerica = DeleteCases[Table[{i, CountryData[i, "LifeExpectancy"]},
          {i, CountryData["SouthAmerica"]}], {_, _Missing}];
```

In[ ]:=
```
Take[dataSouthAmerica, 3]
```

Out[ ]=
```
{{Argentina, 78.07 yr}, {Bolivia, 70.7 yr}, {Brazil, 74.98 yr}}
```

In[ ]:=
```
BarChart[dataSouthAmerica[[All, 2]],
 ChartLabels → dataSouthAmerica[[All, 1]], BarOrigin → Left]
```

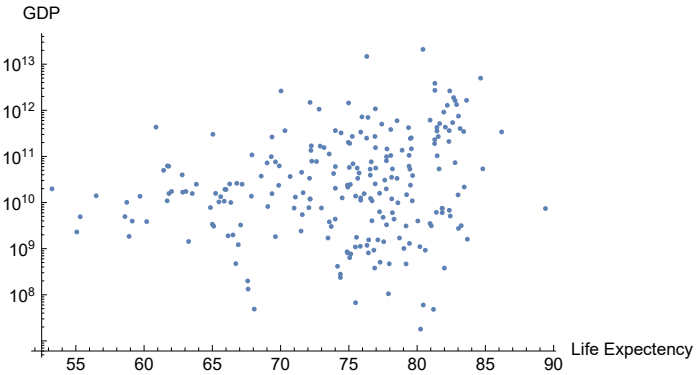
Out[ ]=



In[ ]:=
```
data = Table[Tooltip[{CountryData[i, "LifeExpectancy"], CountryData[i, "GDP"]},
    CountryData[i, "Name"]], {i, CountryData[]}];
```

In[ ]:=
```
ListLogPlot[data, AxesLabel → {"Life Expectency", "GDP"}]
```

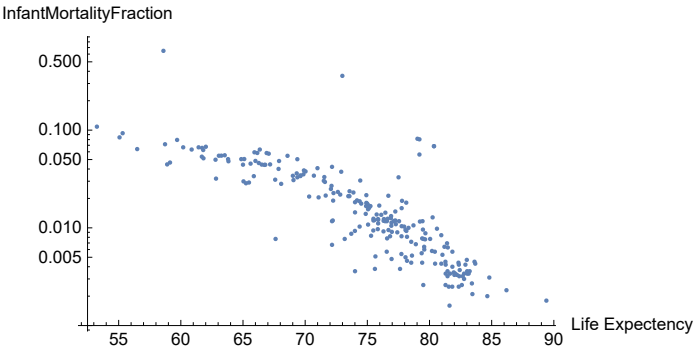
Out[ ]=



In[ ]:=
```
data = Table[
   Tooltip[{CountryData[i, "LifeExpectancy"], CountryData[i, "InfantMortalityFraction"]},
    CountryData[i, "Name"]], {i, CountryData[]}];
```

In[ ]:=
```
ListLogPlot[data, AxesLabel → {"Life Expectency", "InfantMortalityFraction"}]
```

Out[ ]=



antMortalityFraction
Life Expecte

In[ ]:=
```
Manipulate[
 plotFn[Table[Tooltip[{CountryData[i, "LifeExpectancy"], CountryData[i, prop]},
    CountryData[i, "Name"]], {i, CountryData[All]}],
  AxesLabel → {"Life Expetency", " prop"}],
 {prop, {"InfantMortalityFraction", "GDP", "LiteracyFraction"}},
 {{PlotFn, ListLogPlot}, {ListPlot, ListLogPlot, ListLogLogPlot}}, SaveDefinitions → True]
```

prop: InfantMortalityFraction | GDP | LiteracyFraction

PlotFn: ListPlot | ListLogPlot | ListLogLogPlot

```
plotFn[{{{53.25 yr, 0.1085 people/person},
   {Missing[NotAvailable], Missing[NotAvailable]},
   {79.23 yr, 0.0116 people/person}, {77.79 yr, 0.0189 people/person},
   {75.06 yr, 0.0108 people/person}, {83.23 yr, 0.0036 people/person},
   {61.71 yr, 0.0658 people/person}, {82 yr, 0.0033 people/person},
   {77.55 yr, 0.0117 people/person}, {78.07 yr, 0.0095 people/person},
   {75.86 yr, 0.0123 people/person}, {77.76 yr, 0.0104 people/person},
   {82.89 yr, 0.0042 people/person}, {82.07 yr, 0.0034 people/person},
   {73.88 yr, 0.023 people/person}, {75.87 yr, 0.0111 people/person},
   {79.67 yr, 0.0088 people/person}, {74.43 yr, 0.0305 people/person},
   {78.31 yr, 0.01 people/person}, {74.01 yr, 0.0036 people/person},
   {81.65 yr, 0.0034 people/person}, {75.56 yr, 0.012 people/person},
   {61.82 yr, 0.0515 people/person}, {81.83 yr, 0.0025 people/person},
   {71.5 yr, 0.0303 people/person}, {70.7 yr, 0.0342 people/person},
   {Missing[NotAvailable], Missing[NotAvailable]},
   {77.74 yr, 0.0054 people/person}, {65.24 yr, 0.0286 people/person},
   {Missing[NotAvailable], Missing[NotAvailable]},
   {74.98 yr, 0.0169 people/person},
   {Missing[NotAvailable], Missing[NotAvailable]},
   {79.44 yr, 0.0117 people/person}, {78.14 yr, 0.0093 people/person},
   {75.3 yr, 0.0083 people/person}, {63.06 yr, 0.0547 people/person},
   {67.07 yr, 0.0574 people/person}, {66.27 yr, 0.0461 people/person},
   {62.79 yr, 0.0498 people/person}, {83.62 yr, 0.0045 people/person},
   {73.47 yr, 0.0211 people/person}, {81.84 yr, 0.0057 people/person},
   {55.07 yr, 0.0843 people/person}, {58.73 yr, 0.0717 people/person},
   {79.57 yr, 0.0064 people/person}, {76.31 yr, 0.0118 people/person},
   {Missing[NotAvailable], Missing[NotAvailable]},
   {Missing[NotAvailable], Missing[NotAvailable]}
```

```
{Missing[NotAvailable], Missing[NotAvailable]},
{76.91 yr, 0.0132 people/person},
{66.9 yr, 0.0583 people/person}, {76.89 yr, 0.0126 people/person},
{79.41 yr, 0.0078 people/person}, {76.97 yr, 0.0091 people/person},
{79.41 yr, 0.0044 people/person}, {79.19 yr, 0.0805 people/person},
{79.51 yr, 0.0077 people/person}, {79.5 yr, 0.0026 people/person},
{61.43 yr, 0.0667 people/person}, {81.45 yr, 0.0032 people/person},
{65 yr, 0.0443 people/person}, {77.96 yr, 0.0103 people/person},
{72.28 yr, 0.0227 people/person}, {69.62 yr, 0.0339 people/person},
{77.76 yr, 0.0159 people/person}, {74.01 yr, 0.0183 people/person},
{75.11 yr, 0.0163 people/person}, {66.35 yr, 0.0633 people/person},
{66.51 yr, 0.0444 people/person}, {77.63 yr, 0.0038 people/person},
{67.9 yr, 0.0483 people/person}, {77.9 yr, Missing[NotAvailable]},
{81.04 yr, 0.0053 people/person}, {74 yr, 0.0093 people/person},
{81.55 yr, 0.0025 people/person}, {82.39 yr, 0.0032 people/person},
{77.121 yr, 0.01176 people/person}, {78.19 yr, 0.0046 people/person},
{Missing[NotAvailable], Missing[NotAvailable]},
{69.37 yr, 0.0329 people/person}, {66.15 yr, 0.0584 people/person},
{75.14 yr, 0.016 people/person}, {77.25 yr, 0.0147 people/person},
{81.3 yr, 0.0034 people/person}, {69.01 yr, 0.0341 people/person},
{80.2 yr, 0.0058 people/person}, {81.28 yr, 0.0045 people/person},
{73.71 yr, 0.0087 people/person}, {75.48 yr, 0.0094 people/person},
{80.947 yr, 0.00841 people/person}, {77.25 yr, 0.0119 people/person},
{72.63 yr, 0.0233 people/person}, {83.03 yr, 0.0034 people/person},
{63.53 yr, 0.0553 people/person}, {63.26 yr, 0.0548 people/person},
{71.59 yr, 0.0295 people/person}, {65.61 yr, 0.0454 people/person},
{74.9 yr, 0.0167 people/person}, {83.41 yr, 0.0027 people/person},
{76.95 yr, 0.0048 people/person}, {83.45 yr, 0.0021 people/person},
{70.03 yr, 0.0378 people/person}, {72.82 yr, 0.0219 people/person},
{75.06 yr, 0.0155 people/person}, {72.9 yr, 0.0375 people/person},
{81.45 yr, 0.0036 people/person}, {81.84 yr, 0.004 people/person},
{83.15 yr, 0.0034 people/person}, {82.67 yr, 0.0032 people/person},
{61.8 yr, 0.0626 people/person}, {75.49 yr, 0.0124 people/person},
{84.65 yr, 0.002 people/person}, {82.43 yr, 0.0037 people/person},
{75.75 yr, 0.0137 people/person}, {72.25 yr, 0.019 people/person},
{69.32 yr, 0.0361 people/person}, {67.59 yr, 0.0311 people/person},
{72.99 yr, 0.3593 people/person}, {78.9 yr, 0.0068 people/person},
{72.07 yr, 0.025 people/person}, {66 yr, 0.0484 people/person},
```

Out[•]=

```
{ 75.65 yr , 0.0051 people/person }, { 78.53 yr , 0.0072 people/person },
{ 58.9 yr , 0.0446 people/person }, { 65.1 yr , 0.0506 people/person },
{ 76.93 yr , 0.0105 people/person }, { 82.36 yr , 0.0042 people/person },
{ 75.61 yr , 0.0038 people/person }, { 82.78 yr , 0.0034 people/person },
{ 84.81 yr , 0.0031 people/person }, { 76.59 yr , 0.0078 people/person },
{ 67.86 yr , 0.0401 people/person }, { 72.16 yr , 0.0421 people/person },
{ 75.87 yr , 0.0121 people/person }, { 76.69 yr , 0.0213 people/person },
{ 62.01 yr , 0.0676 people/person }, { 83 yr , 0.0047 people/person },
{ 74.38 yr , 0.0187 people/person }, { 81.41 yr , 0.00695 people/person },
{ 64.86 yr , 0.0505 people/person }, { 76.7 yr , 0.0095 people/person },
{ 79.19 yr , 0.05629 people/person }, { 76.94 yr , 0.0113 people/person },
{ 74.17 yr , 0.0191 people/person }, { 72.16 yr , 0.0117 people/person },
{ 89.4 yr , 0.0018 people/person }, { 71.08 yr , 0.0205 people/person },
{ 77.51 yr , 0.0329 people/person }, { 75.49 yr , 0.0119 people/person },
{ 73.56 yr , 0.0211 people/person }, { 56.49 yr , 0.064 people/person },
{ 69.62 yr , 0.0344 people/person }, { 65.87 yr , 0.0338 people/person },
{ 67.62 yr , 0.0077 people/person }, { 72.12 yr , 0.0269 people/person },
{ 81.95 yr , 0.0035 people/person }, { 78.59 yr , 0.0052 people/person },
{ 82.33 yr , 0.0044 people/person }, { 74.51 yr , 0.0177 people/person },
{ 59.7 yr , 0.0794 people/person }, { 60.87 yr , 0.0633 people/person },
{Missing[NotAvailable], Missing[NotAvailable]},
{Missing[NotAvailable], Missing[NotAvailable]},
{ 76.33 yr , 0.0123 people/person }, { 71.65 yr , 0.0214 people/person },
{ 82.35 yr , 0.0025 people/person }, { 76.64 yr , 0.0124 people/person },
{ 69.37 yr , 0.0504 people/person }, { 74.38 yr , 0.0103 people/person },
{ 79.47 yr , 0.0096 people/person }, { 69.86 yr , 0.0353 people/person },
{ 78.13 yr , 0.0181 people/person }, { 74.96 yr , 0.0178 people/person },
{ 70.32 yr , 0.0209 people/person }, {Missing[NotAvailable],
 Missing[NotAvailable]}, { 78.53 yr , 0.0044 people/person },
{ 81.29 yr , 0.0026 people/person }, { 81.47 yr , 0.0063 people/person },
{ 79.58 yr , 0.006 people/person }, { 61.69 yr , 0.0535 people/person },
{ 79.646 yr , 0.00763 people/person }, { 76.3 yr , 0.0092 people/person },
{ 72.16 yr , 0.0067 people/person }, { 65.48 yr , 0.0291 people/person },
{ 80.36 yr , 0.0685 people/person }, { 80.25 yr , 0.0128 people/person },
{ 76.83 yr , 0.0082 people/person }, { 78.71 yr , 0.0106 people/person },
{ 80.36 yr , 0.0685 people/person }, { 81.2 yr , 0.0064 people/person },
{ 76.43 yr , 0.0117 people/person }, { 74.92 yr , 0.018 people/person },
{ 83.68 yr , 0.0043 people/person }, { 66.72 yr , 0.0441 people/person },
```

```
    { 76.4 yr , 0.0121 people/person }, { 63.83 yr , 0.048 people/person },
    { 76.56 yr , 0.0057 people/person }, { 75.84 yr , 0.0097 people/person },
    { 60.19 yr , 0.0667 people/person }, { 86.19 yr , 0.0023 people/person },
    { 79.03 yr , 0.0816 people/person }, { 78.07 yr , 0.005 people/person },
    { 81.61 yr , 0.0016 people/person }, { 76.45 yr , 0.0143 people/person },
    { 55.32 yr , 0.093 people/person }, { 65.04 yr , 0.0299 people/person },
    {Missing[NotAvailable], Missing[NotAvailable]},
    { 82.78 yr , 0.003 people/person }, { 58.6 yr , 0.6477 people/person },
    { 82.21 yr , 0.0033 people/person }, { 77.75 yr , 0.0082 people/person },
    { 66.79 yr , 0.0442 people/person }, { 73.57 yr , 0.0237 people/person },
    {Missing[NotAvailable], Missing[NotAvailable]},
    { 59.13 yr , 0.0466 people/person },
    { 82.6 yr , 0.0026 people/person }, { 83.03 yr , 0.0036 people/person },
    { 74.01 yr , 0.0144 people/person }, { 80.95 yr , 0.0043 people/person },
    { 69.06 yr , 0.0308 people/person }, { 69.9 yr , 0.0387 people/person },
    { 77.41 yr , 0.009 people/person }, { 70.99 yr , 0.0408 people/person },
    {Missing[NotAvailable], Missing[NotAvailable]},
    { 77.29 yr , 0.0109 people/person }, { 74.92 yr , 0.0216 people/person },
    { 76.57 yr , 0.0117 people/person }, { 75.96 yr , 0.0169 people/person },
    { 71.54 yr , 0.0331 people/person }, { 80.6 yr , 0.0098 people/person },
    { 68.07 yr , 0.0282 people/person }, { 68.58 yr , 0.0546 people/person },
    { 73.18 yr , 0.0077 people/person }, { 79.37 yr , 0.0055 people/person },
    { 81.3 yr , 0.0042 people/person }, { 80.43 yr , 0.0057 people/person },
    {Missing[NotAvailable], Missing[NotAvailable]},
    { 80.05 yr , 0.0077 people/person }, { 78.19 yr , 0.0081 people/person },
    { 75.03 yr , 0.0174 people/person }, { 74.87 yr , 0.0139 people/person },
    {Missing[NotAvailable], Missing[NotAvailable]},
    { 72.22 yr , 0.0119 people/person }, { 75.25 yr , 0.0167 people/person },
    { 80.45 yr , 0.0043 people/person }, { 76.12 yr , 0.0136 people/person },
    { 63.8 yr , 0.0505 people/person }, { 67.18 yr , 0.0446 people/person },
    { 65.92 yr , 0.0593 people/person }, { 62.83 yr , 0.0319 people/person }},
  AxesLabel → {Life Expetency,  prop}]
```

In[ ]:=

```
Clear[data]
```

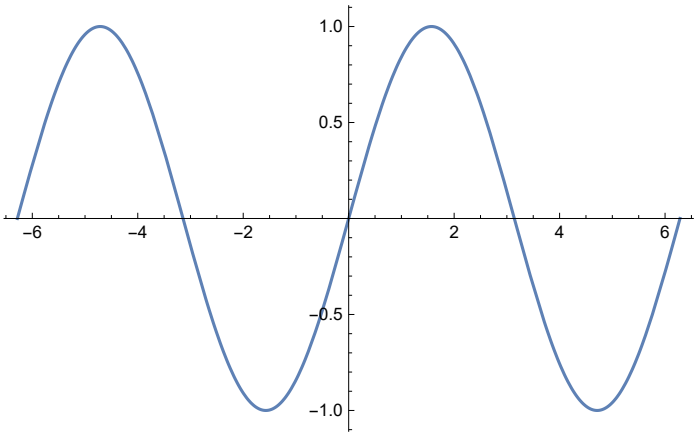
Chapter3| Input and Output

In[ ]:=
```
2^100
```

Out[ ]= 1 267 650 600 228 229 401 496 703 205 376

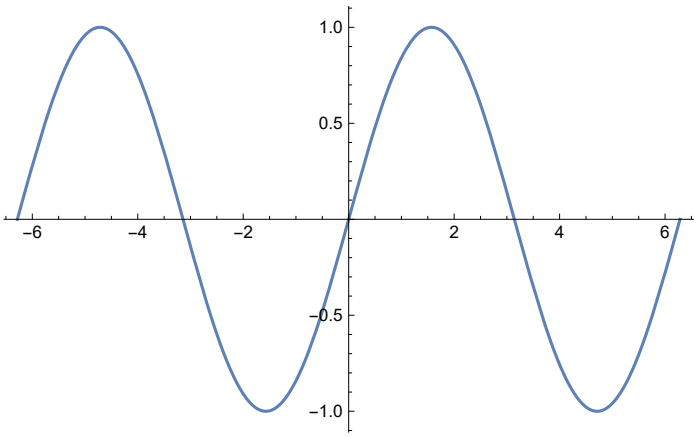
In[ ]:=
```
plot sin(x)
Plot[Sin[x], {x, -2*Pi, 2*Pi}]
```

Out[ ]=



In[ ]:=
```
plot the sin of x
Plot[Sin[x], {x, -2*Pi, 2*Pi}]
```

Out[ ]=

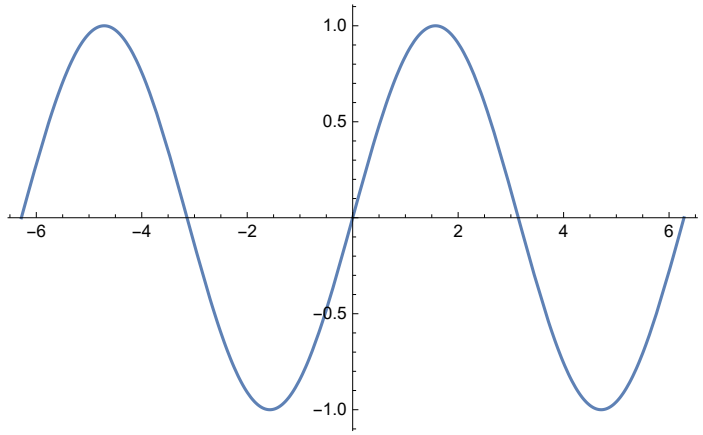


In[ ]:=

graph of sin x

```
Plot[Sin[x], {x, -2*Pi, 2*Pi}]
```

Out[ ]=



In[ ]:=

integral of 1/(x^3 +1)

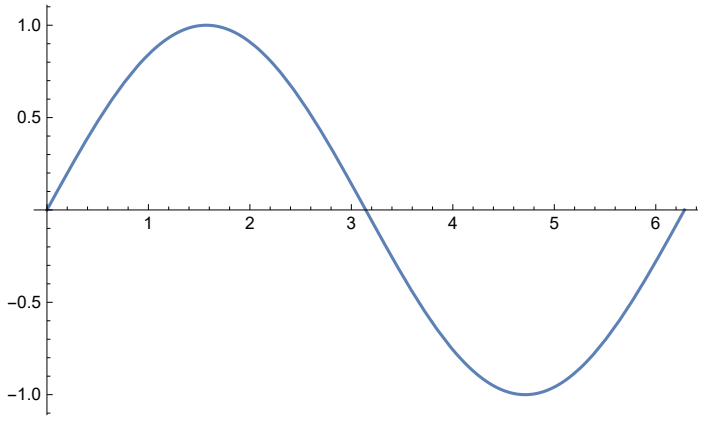
```
Integrate[1 / (x^3 + 1), x]
```

Out[ ]=

$$\frac{\text{ArcTan}\left[\frac{-1+2x}{\sqrt{3}}\right]}{\sqrt{3}} + \frac{1}{3}\text{Log}[1+x] - \frac{1}{6}\text{Log}[1-x+x^2]$$

```
Plot[Sin[x], {x, 0, 2 π}] (* Plot[function, {var, min, max}] *)
```

Out[ ]=



In[ ]:=

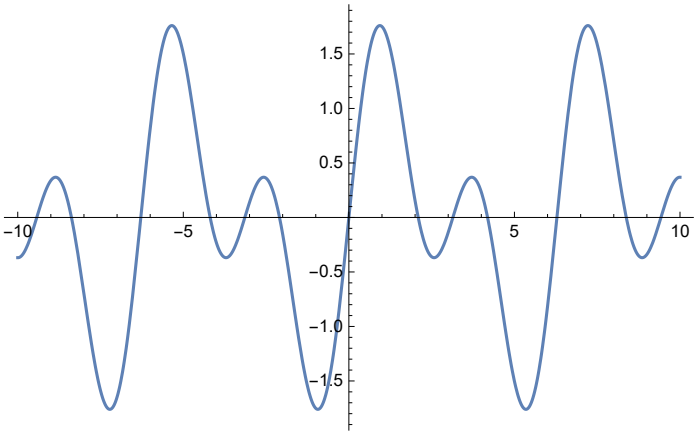
```
Expand[(a + b) ^10]
```

Out[ ]=

$$a^{10} + 10\,a^9\,b + 45\,a^8\,b^2 + 120\,a^7\,b^3 + 210\,a^6\,b^4 + 252\,a^5\,b^5 + 210\,a^4\,b^6 + 120\,a^3\,b^7 + 45\,a^2\,b^8 + 10\,a\,b^9 + b^{10}$$

In[ ]:=
```
Plot[Sin[x] + Sin[2 x], {x, -10, 10}]
```

Out[ ]=



In[ ]:=
```
Solve[-10 + x^2 == 0, x]
```

Out[ ]=
$$\{\{x \to -\sqrt{10}\}, \{x \to \sqrt{10}\}\}$$

In[ ]:=
```
first 25 prime numbers »
Values
Prime[Range[1, 25]]
```

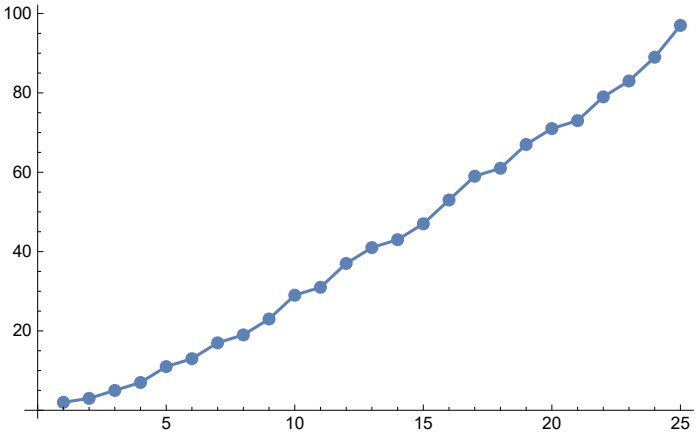
Out[ ]=
```
{2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97}
```

```
Prime[Range[1, 25]]
```

Out[ ]=
```
{2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97}
```

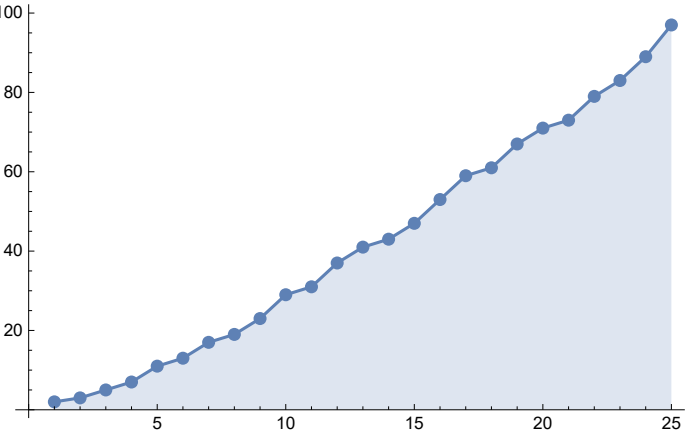
In[ ]:=
```
ListLinePlot[Prime[Range[1, 25]], Mesh → All]
```

Out[ ]=



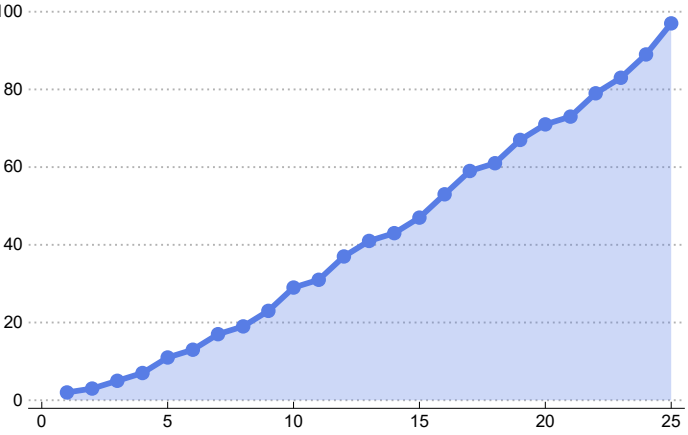
In[ ]:=
```
ListLinePlot[Prime[Range[1, 25]], Filling → Automatic, Mesh → All]
```

Out[ ]=



In[ ]:=
```
ListLinePlot[Prime[Range[1, 25]], PlotTheme → "Business", Filling → Automatic, Mesh → All]
```

Out[ ]=



In[ ]:=
```
N[1234/5678]  (*approximate an exact quantity*)
```

Out[ ]=
```
0.21733
```

In[ ]:=
```
∫_0^1 Sin[x] dx
```

Out[ ]=
```
1 - Cos[1]
```

In[ ]:=
```
N[%] (*% shorthand for the last output*)
```

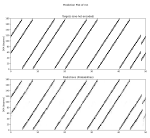
Out[ ]=
```
0.459698
```

```
∫_0^1 Sin[x] dx // N (* // POSTfix *)
```

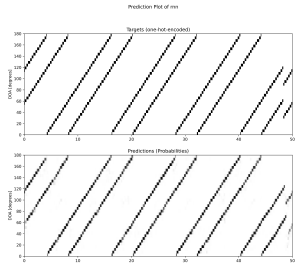
Chapter4 | Word processing and typesetting

Chapter 5| Presenting with Slide Shows

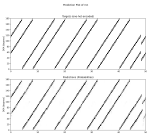
File -> New -> Slide show

In[ ]:=
```
Column[{  , "Prediction Plot"}]
```

Out[ ]=

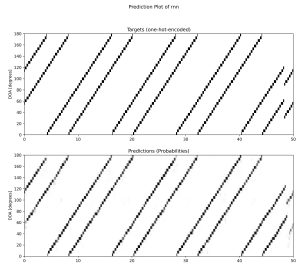


Prediction Plot

```
Column[{  , "Prediction Plot"}, Center, 3]
```

```
(* 3 determine the verticle spacing between elements *)
```

Out[ ]=

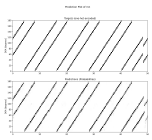


Prediction Plot

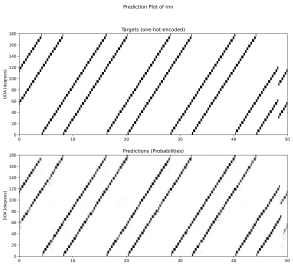
```
Grid[{            (* Grid: 2D layoyt,
  takes a list of lists; the first sublist becomes the first row*)
   {a, b, c},
   {d, e, f},
   {g, h, i}
 }]
```

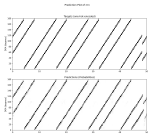
Out[ ]=

a	b	c
d	e	f
g	h	i

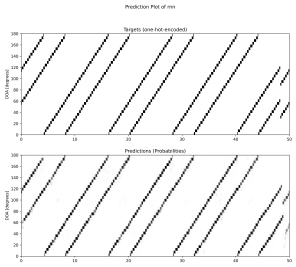
In[ ]:=
```
Grid[{
  {"Image",  },
  {"Color space", "BW"},
  {"Dimention", "{1080, 720}"}
 }, Alignment → Left, Frame → True, Dividers → All]
```

Out[ ]=

Image	
Color space	BW
Dimention	{1080, 720}

In[ ]:=
```
Style[Grid[{
  {Style["Image", Bold, Red],  },
  {"Color space", "BW"},
  {"Dimention", "{1080, 720}"}
 }, Alignment → Left, Frame → True, Dividers → All], FontFamily → "Times"]
```

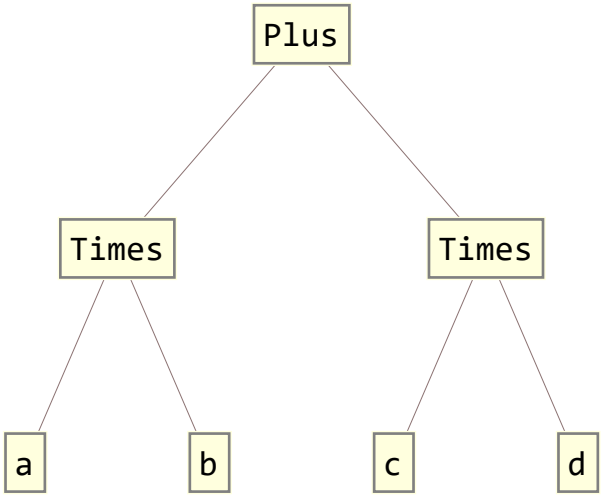
Out[ ]=

Image	
Color space	BW
Dimention	{1080, 720}

Chapter 6| Fundamentals of the Wolfram Language

```
In[•]:= FullForm[a b + c d]
Out[•]//FullForm= Plus[Times[a, b], Times[c, d]]
```

```
In[•]:= TreeForm[a b + c d]
Out[•]//TreeForm=
```




```
In[•]:= Table[{i, i^2}, {i, 1, 10, 1}]
Out[•]= {{1, 1}, {2, 4}, {3, 9}, {4, 16}, {5, 25}, {6, 36}, {7, 49}, {8, 64}, {9, 81}, {10, 100}}
```

```
In[•]:= π Squared is N[π^2]
Out[•]= 31.0063 is Squared
```

```
In[•]:= StringJoin[" The first part ", " and the second part"]
Out[•]=  The first part  and the second part
```

```
In[•]:= ToString[N[π^2]]
Out[•]= 9.8696
```

```
In[•]:= FullForm[%]
Out[•]//FullForm= "9.8696"
```

```
In[•]:= StringJoin["π Squared is", ToString[N[π^2]]]
Out[•]= π Squared is9.8696
```

```
10 m
―― 5 s  (* units *)
 s
```

Out[•]= 50 m

In[•]:=
```
Quantity[2, "Feet"] + Quantity[3, "Meters"]
```

Out[•]= $\frac{1504}{127}$ ft

In[•]:=
```
UnitConvert[%, "Meters"]
```

Out[•]= $\frac{2256}{625}$ m

In[•]:=
```
Quantity[15, "days"] + Quantity[9, "weeks"]
```

Out[•]= 78 days

In[•]:=
```
Today
```

Out[•]= Day: Fri 16 Jun 2023

In[•]:=
```
DatePlus[Today, Quantity[9, "weeks"]]
```

Out[•]= Day: Fri 18 Aug 2023

```
f[x_] := x^2 (* Function; left : defintion, right: values *)
```

In[•]:=
```
(* f[x_] function name followed by square brackets
  it has to denoted with underscore char _
  := delayed assingment *)
```

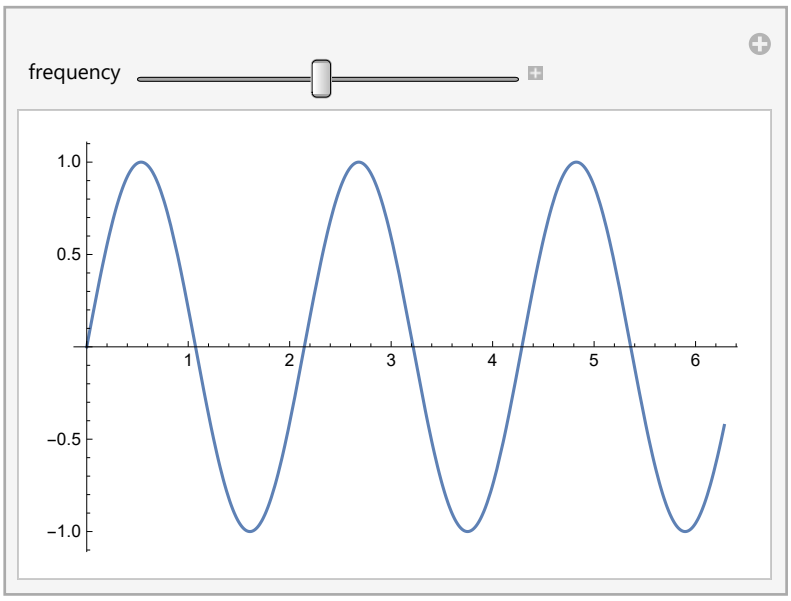
```
Clear[x, f]
```

Chapter 7| Creating Interactive Models with single command

```
(* Manipulate[
 Expression to manipulate,
 Parameter specfication]
*)
```

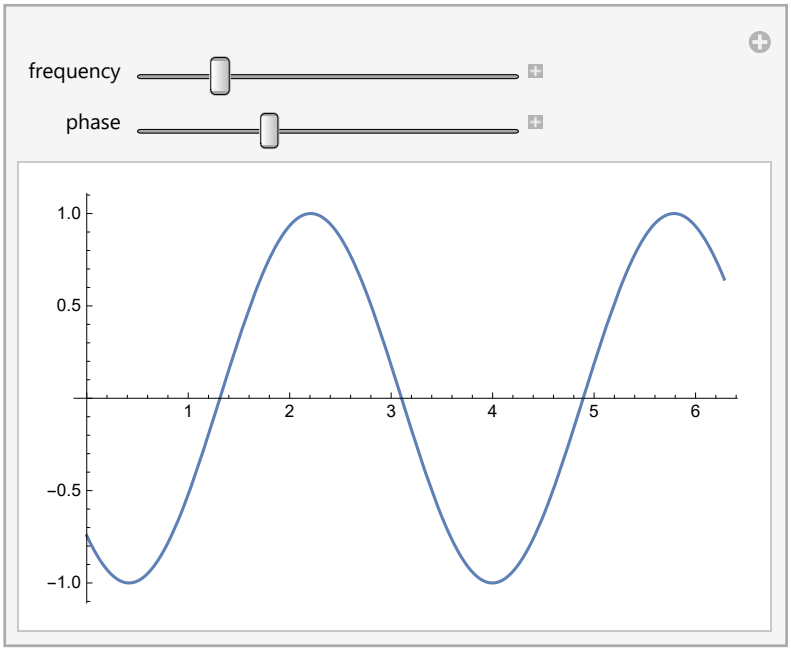
In[ ]:=
```
Manipulate[
 Plot[Sin[frequency * x], {x, 0, 2 π}],
 {frequency, 1, 5}]
```

Out[ ]=



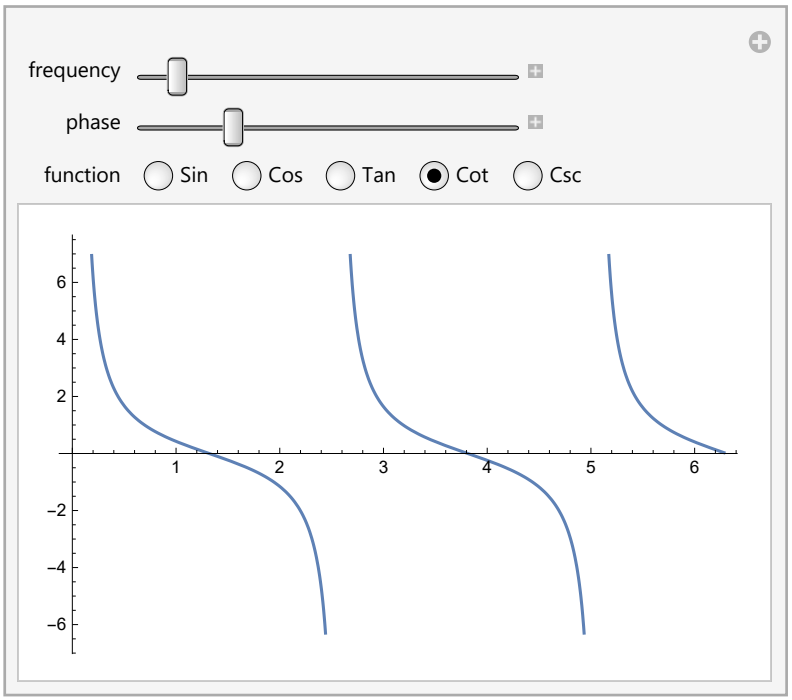

```
Manipulate[                                        (* Multiple controls *)
 Plot[Sin[frequency * x + phase], {x, 0, 2 π}],
 {frequency, 1, 5}, {phase, 1, 10}]
```

Out[ ]=



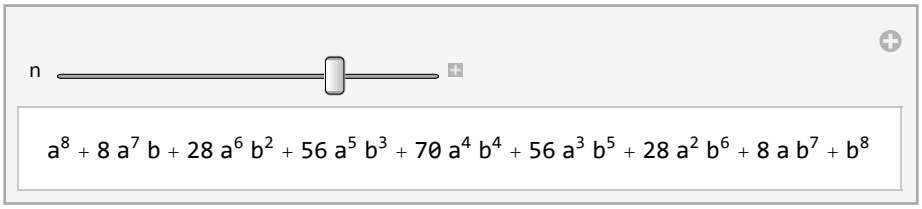
In[ ]:=
```
Manipulate[                                          (* Multiple functions *)
 Plot[function[frequency * x + phase], {x, 0, 2 π}],
 {frequency, 1, 5}, {phase, 1, 10},
 {function, {Sin, Cos, Tan, Cot , Csc}, ControlType → RadioButtonBar}]
```

Out[ ]=



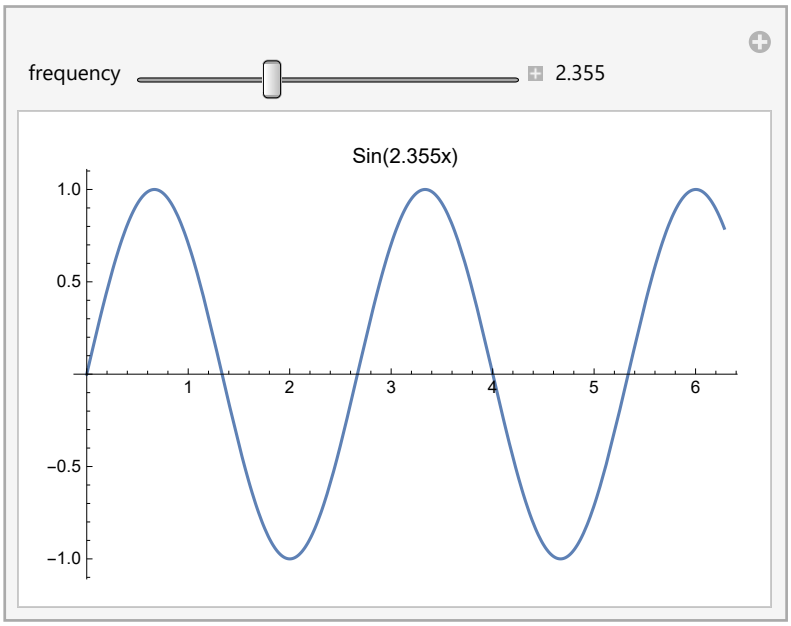
In[ ]:=
```
Manipulate[
 Expand[(a + b)^n], {n, 2, 10, 1}]
```

Out[ ]=



```
Manipulate[
 Plot[Sin[f * x], {x, 0, 2 π}, PlotLabel → "Sin(" <> ToString[f] <> "x)"],
 (* String can be joined togther with the <> operator*)
 {{f, 1, "frequency"}, 1, 5, Appearance → "Labeled"}]
```

Out[ ]=



Chapter8| Sharing Mathematica Notebooks

Chapter 9| Finding Help

In[ ]:=

```
? Plot
```

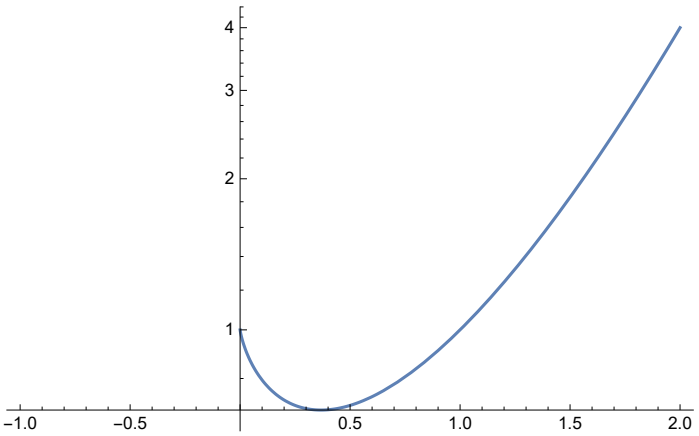
Chapter 10| 2D and 3D Graphics

In[ ]:=

log plot x^x

```
LogPlot[x^x, {x, -1, 2}]
```

Out[ ]=

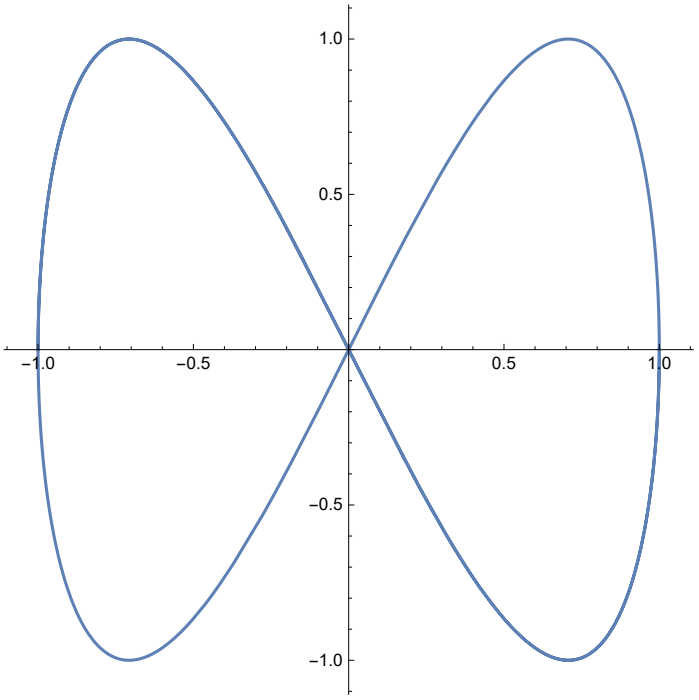


In[ ]:=

parametric plot of sin(u), sin(2u)

```
ParametricPlot[{Sin[u], Sin[2*u]}, {u, (-3*Pi)/2, (3*Pi)/2}]
```

Out[ ]=

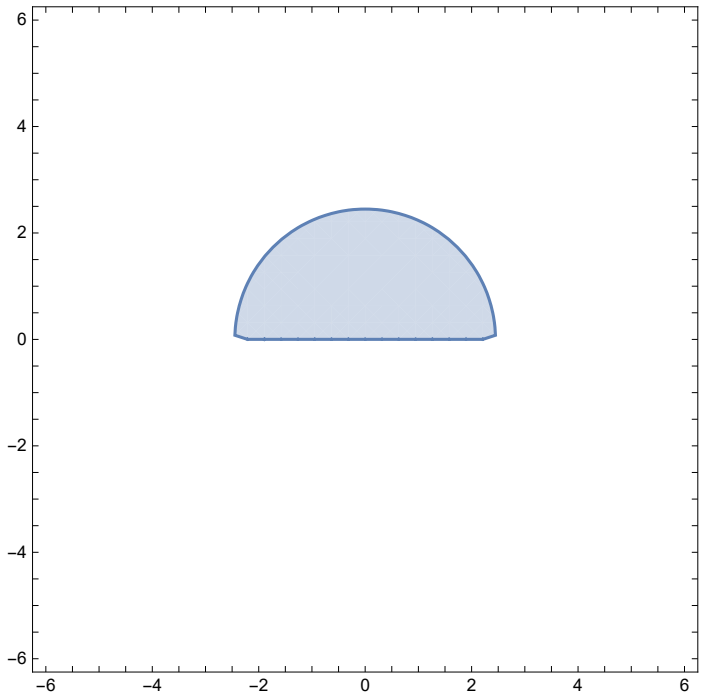


In[ ]:=

plot x^2 + y^2 <= 6 and y>0

```
RegionPlot[Evaluate[x^2 + y^2 <= 6 && y > 0], {x, -6., 6.}, {y, -6., 6.}]
```

Out[ ]=

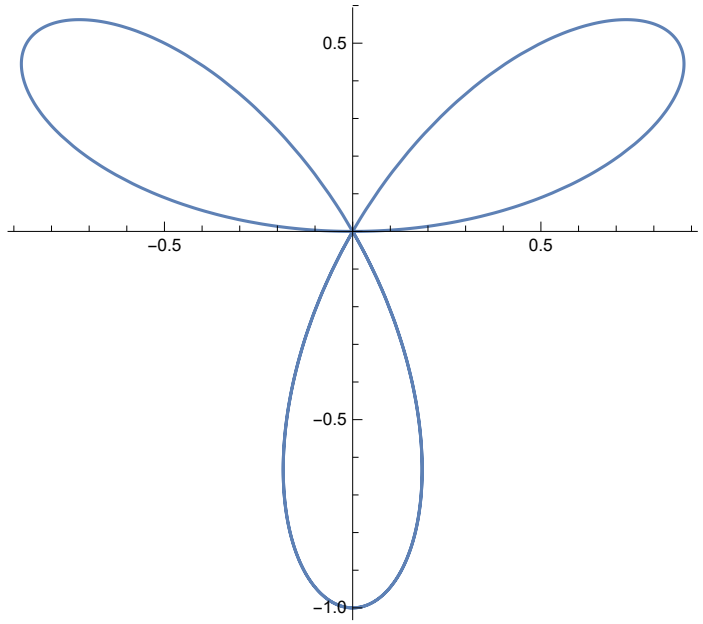


In[ ]:=

polar plot of sin(3t)

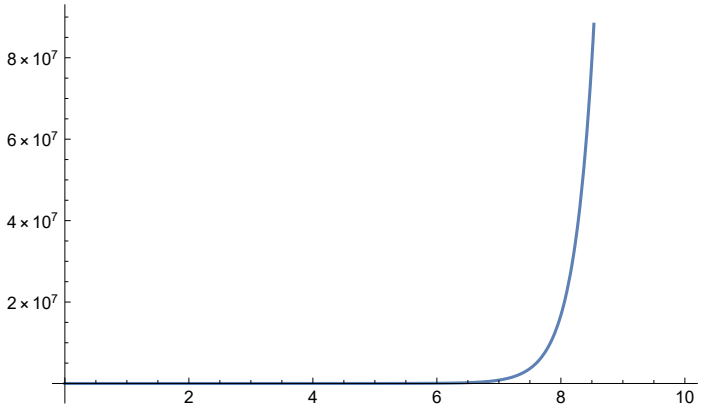
```
PolarPlot[Sin[3*t], {t, (-2*Pi)/3, (2*Pi)/3}]
```

Out[ ]=



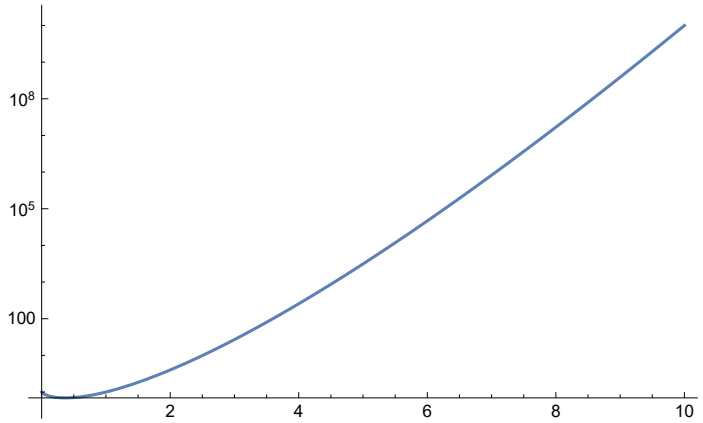
In[ ]:=
```
Plot[x^x, {x, 0, 10}]
```

Out[ ]=



In[ ]:=
```
LogPlot[x^x, {x, 0, 10}]
```

Out[ ]=

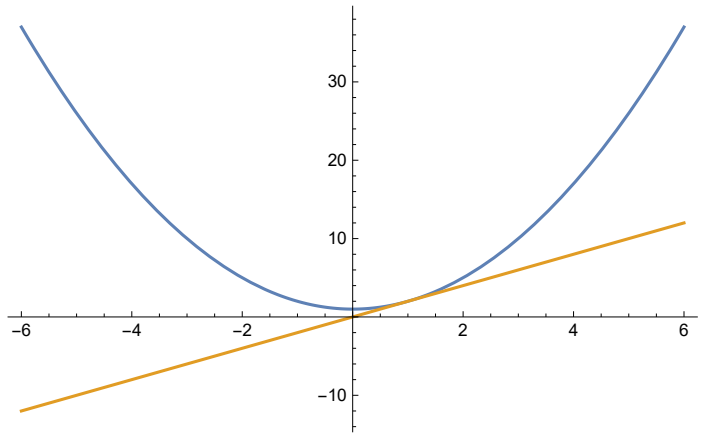


plot x^2 +1 and its derivitive

```
Plot[Evaluate[{x^2 + 1, D[x^2 + 1, x]}], {x, -6., 6.}]
```

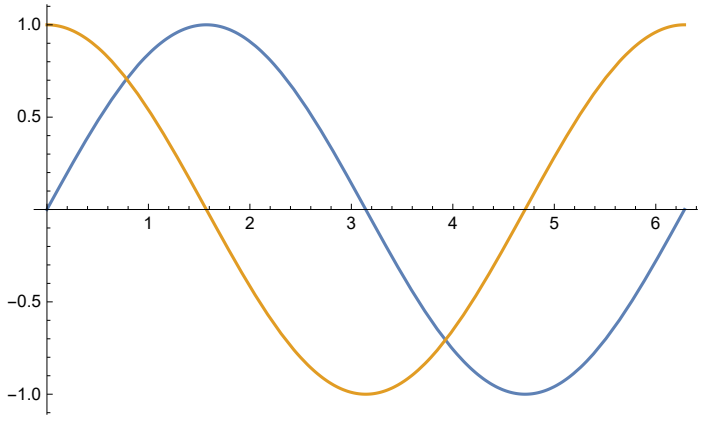
```
(* plotting multiple funs *)
```

Out[ ]=



In[ ]:=
```
Plot[{Sin[x], Cos[x]}, {x, 0, 2 π}]
```

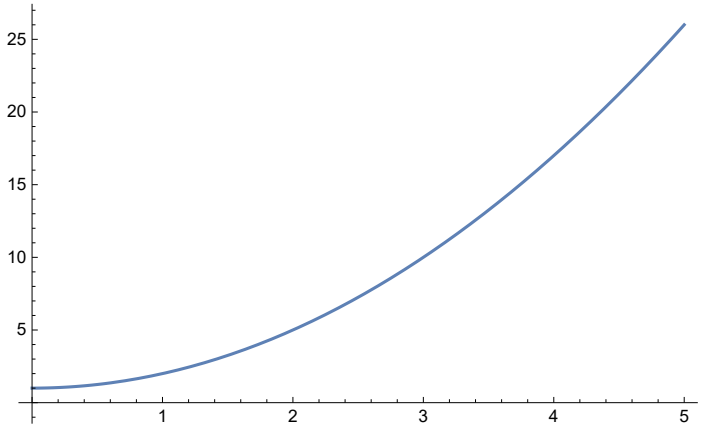
Out[ ]=



In[ ]:=
```
f[x_] := x^2 + 1
```

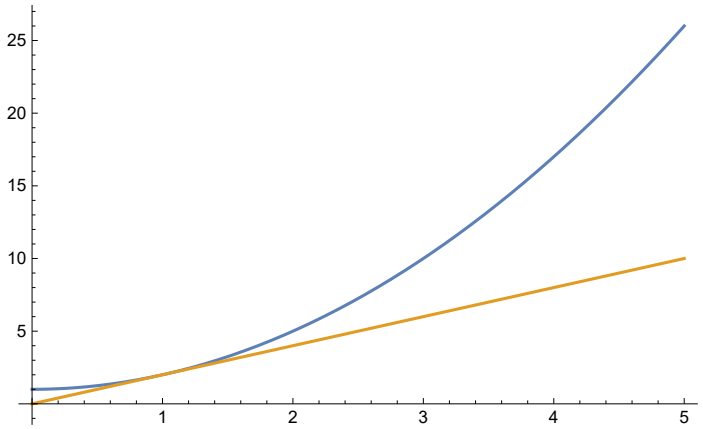
In[ ]:=
```
Plot[f[x], {x, 0, 5}]
```

Out[ ]=



In[ ]:=
```
Plot[{f[x], f'[x]}, {x, 0, 5}]
```

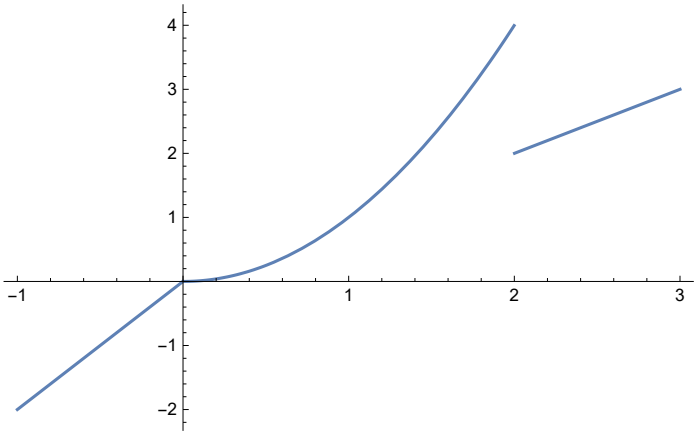
Out[ ]=



In[ ]:=
```
h[x_] := Piecewise[{{2 x, x < 0}, {x^2, 0 ≤ x ≤ 2}, {x, x > 2}}]
```

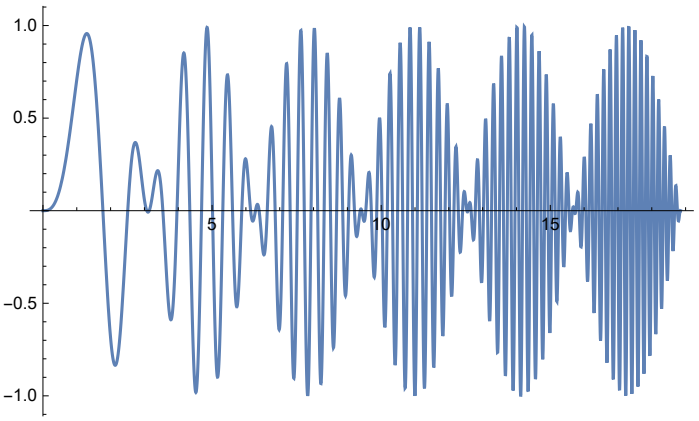
In[ ]:=
```
Plot[h[x], {x, -1, 3}]
```

Out[ ]=



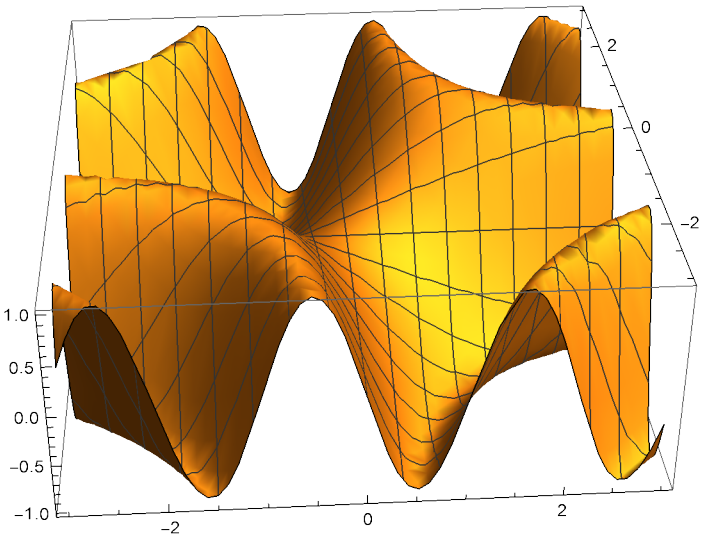
In[ ]:=
```
Plot[Sin[x] × Sin[x^2], {x, 0, 6 π}]
```

Out[ ]=



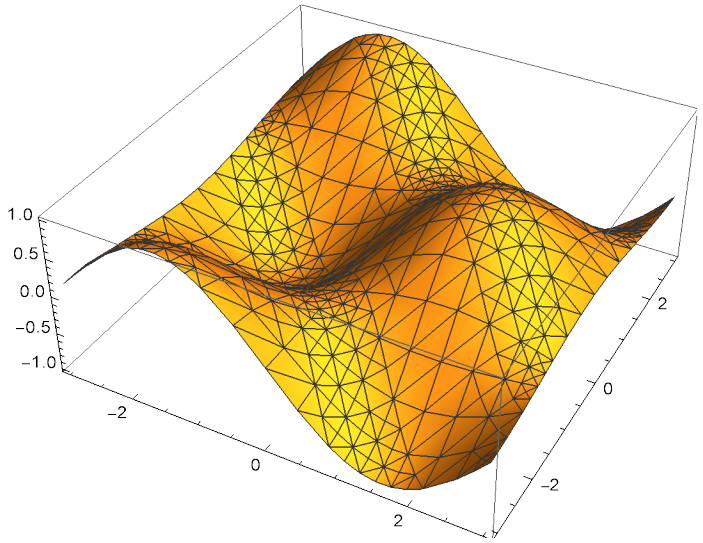
In[ ]:=
```
Plot3D[Sin[x y], {x, -3, 3}, {y, -3, 3}]
```

Out[ ]=



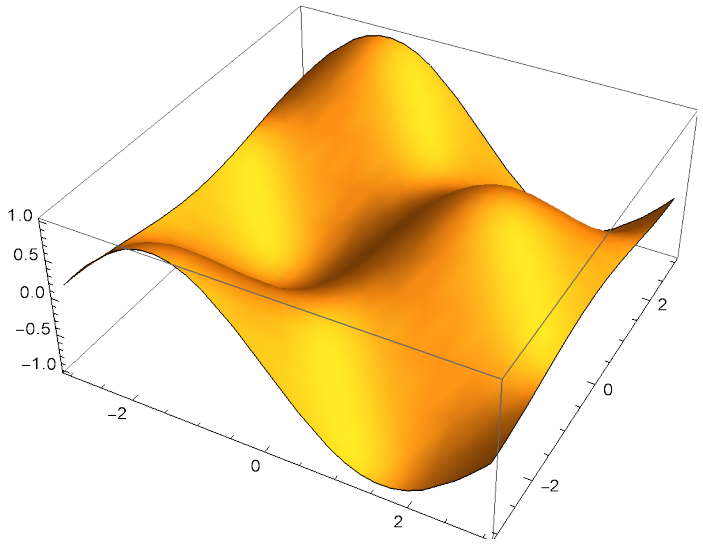
In[ ]:= `Plot3D[Sin[x] × Cos[y], {x, -3, 3}, {y, -3, 3}, Mesh → All]`

Out[ ]=



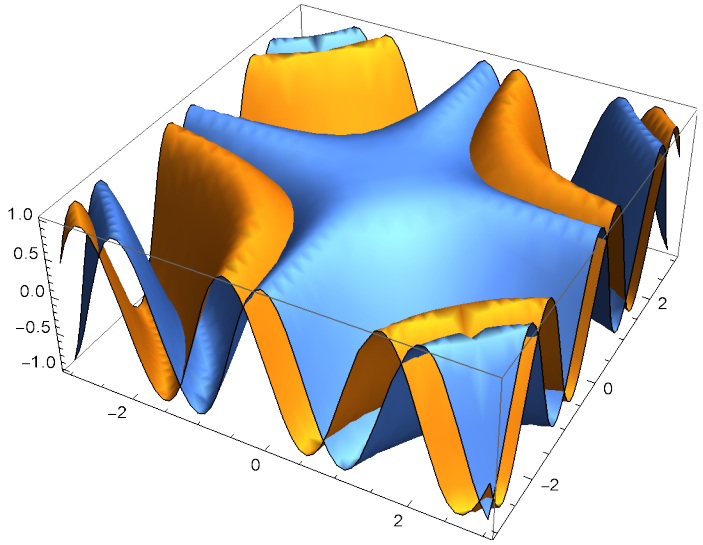
In[ ]:= `Plot3D[Sin[x] × Cos[y], {x, -3, 3}, {y, -3, 3}, Mesh → None]`

Out[ ]=



```
Plot3D[{Sin[x y], Cos[y x]}, {x, -3, 3}, {y, -3, 3}, Mesh → None] (* 3D multiple plot *)
```

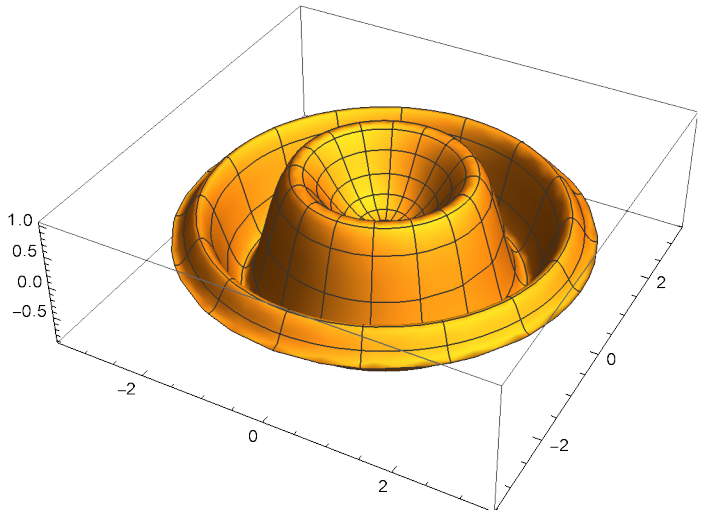
Out[ ]=



In[ ]:=

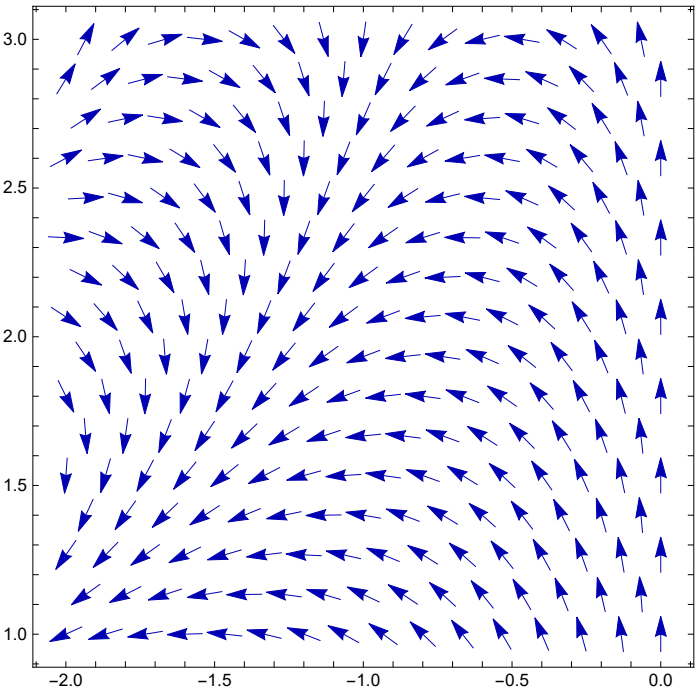
```
RevolutionPlot3D[Sin[t] × Sin[t^2], {t, 0, π}]
```

Out[ ]=



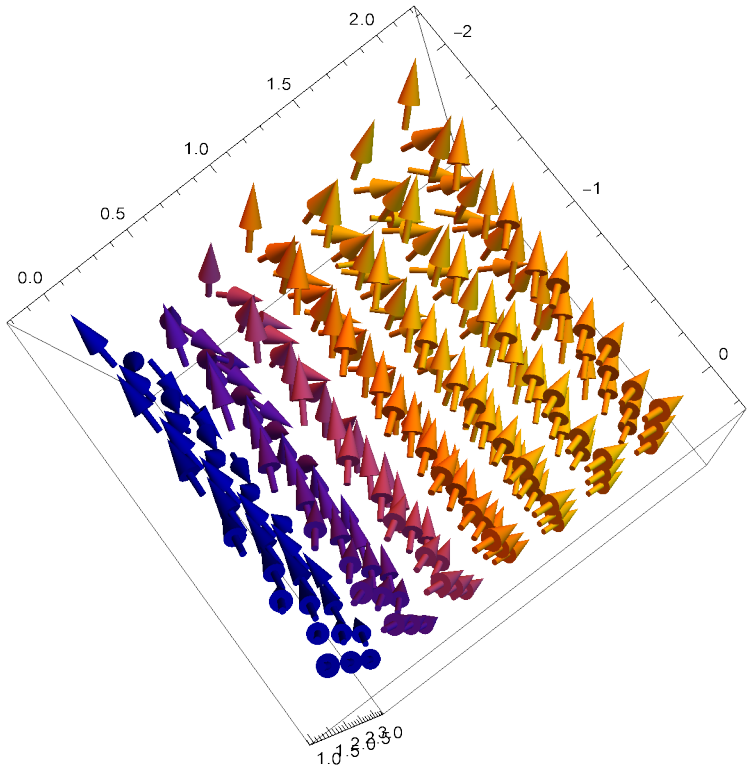
In[ ]:=
```
VectorPlot[{Sin[x y], Cos[x y]}, {x, -2, 0}, {y, 1, 3}]
```

Out[ ]=



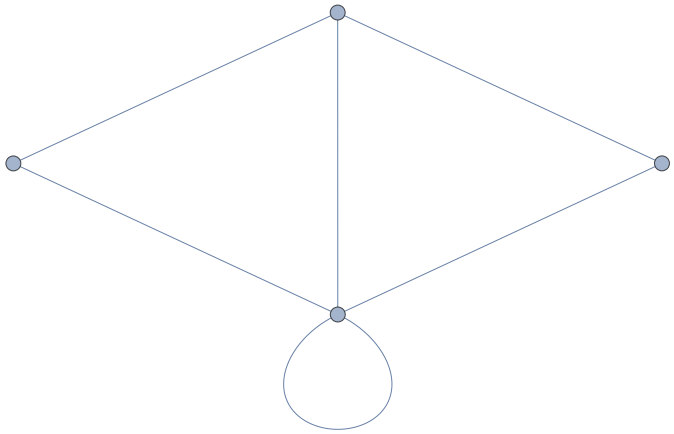
In[ ]:=
```
VectorPlot3D[{Sin[x y], Cos[x y], Sin[z]}, {x, -2, 0}, {y, 1, 3}, {z, 0, 2}]
```

Out[ ]=



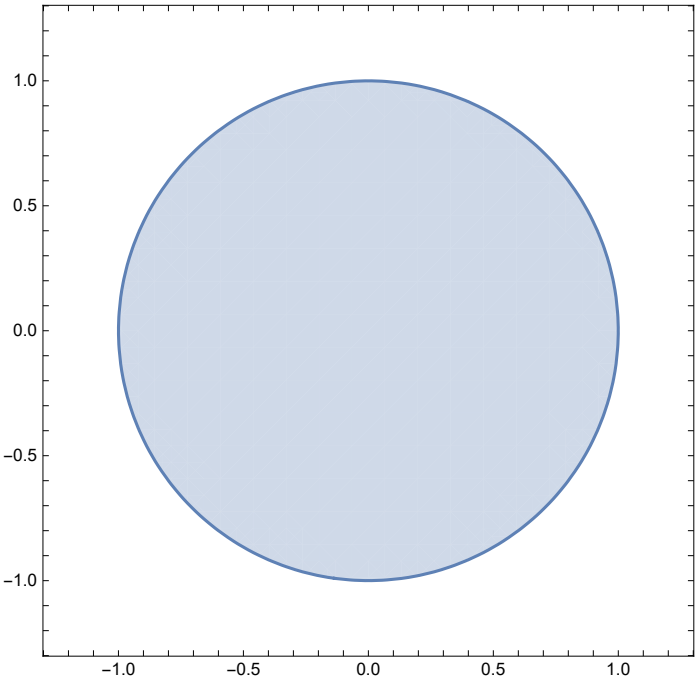
```
Graph[{1 ⟷ 2, 2 ⟷ 3, 3 ⟷ 1, 2 ⟷ 4, 1 ⟷ 4, 2 ⟷ 2}] (* ESC ue → *)
```

Out[ ]=



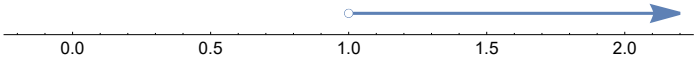
In[ ]:=
```
RegionPlot[x^2 + y^2 ≤ 1, {x, -1.25, 1.25}, {y, -1.25, 1.25}, AxesOrigin → True]
```

Out[ ]=



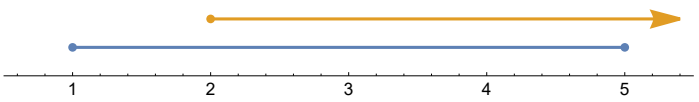
In[ ]:=
```
NumberLinePlot[x > 1, {x, 0, 2}]
```

Out[ ]=



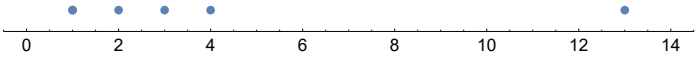
In[ ]:=
```
NumberLinePlot[{Interval[{1, 5}], Interval[{2, ∞}]}]
```

Out[ ]=



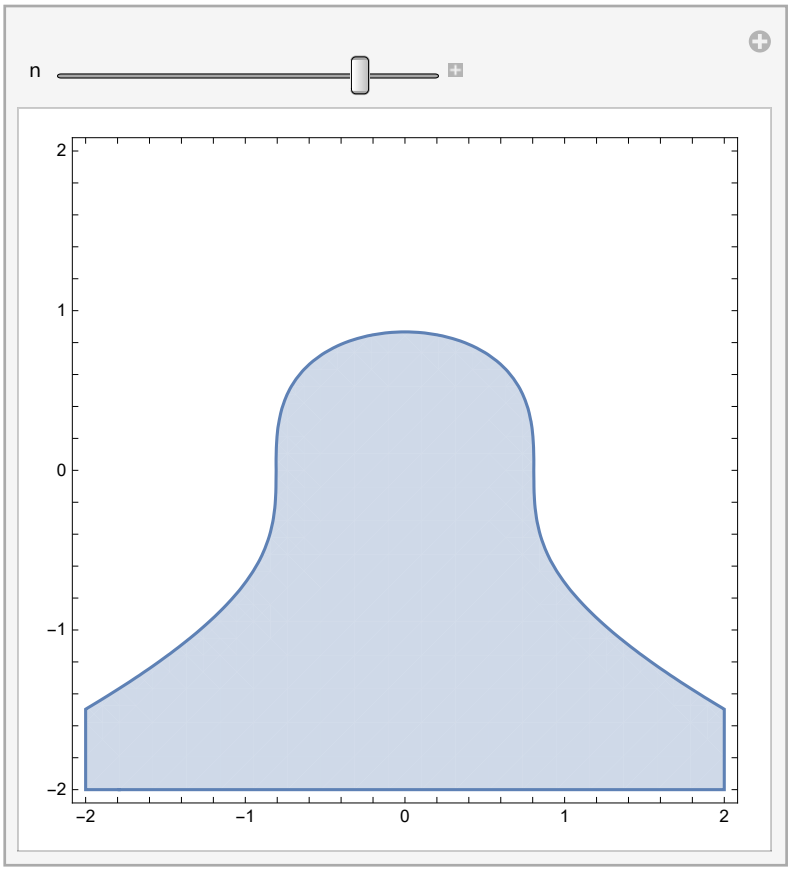
In[ ]:=
```
NumberLinePlot[{1, 1, 2, 3, 4, 13}]
```

Out[ ]=




```
Manipulate[
 RegionPlot[x^2 + y^3 < n , {x, -2, 2}, {y, -2, 2}], {n, -1, 1}]
```

Out[ ]=



In[ ]:=
```
Clear[x, f, h]
```

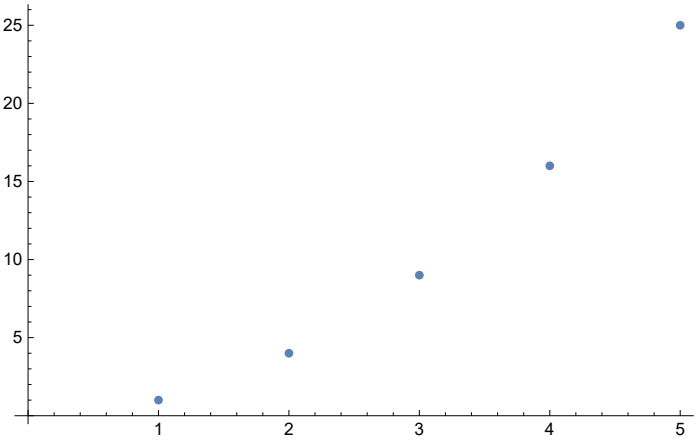
Chapter 11| Visualizing Data

In[ ]:=
```
data = {1, 4, 9, 16, 25}
```

Out[ ]=
```
{1, 4, 9, 16, 25}
```

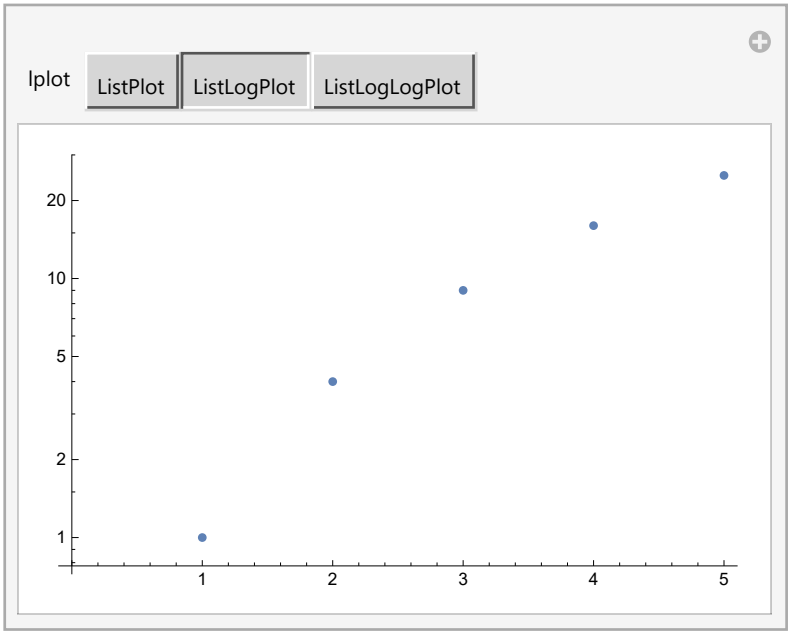
In[ ]:=
```
ListPlot[data]
```

Out[ ]=



In[ ]:=
```
Manipulate[
 lplot[data],
 {lplot, {ListPlot, ListLogPlot, ListLogLogPlot}},
 Initialization :> (data = {1, 4, 9, 16, 25})
]
```

Out[ ]=



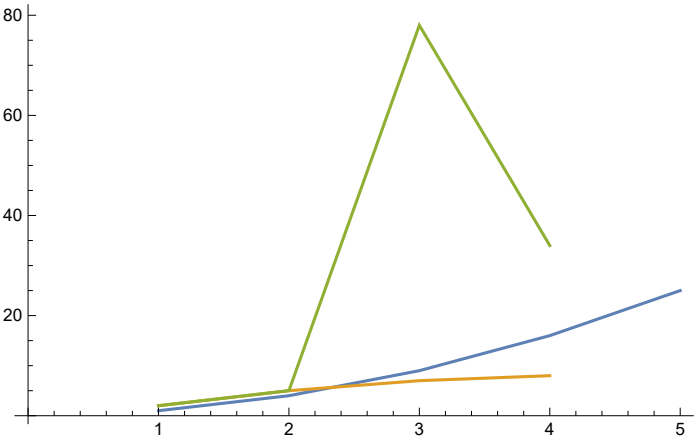
In[ ]:=
```
ds1 = {1, 4, 9, 16, 25};
```

In[ ]:=
```
ds2 = {2, 5, 7, 8};
```

In[ ]:=
```
ds3 = {2, 5, 78, 34};
```

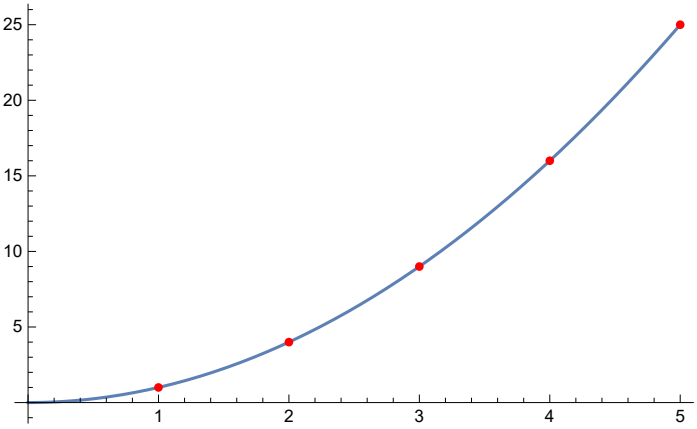
In[•]:=
```
ListLinePlot[{ds1, ds2, ds3}]
```

Out[•]=



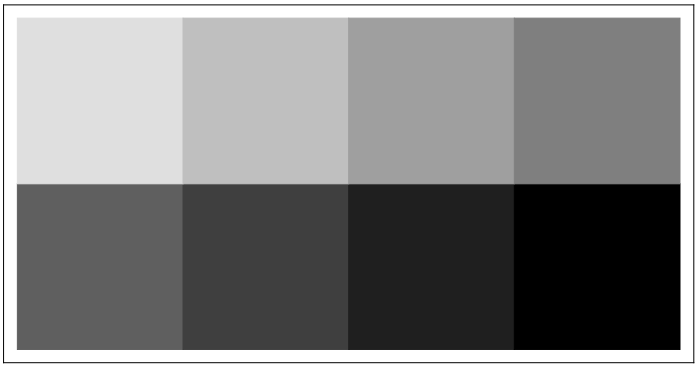
In[•]:=
```
Show[ Plot[x^2, {x, 0, 5}],
 ListPlot[{1, 4, 9, 16, 25}, PlotStyle → {Red, PointSize[Medium]}]]
```

Out[•]=



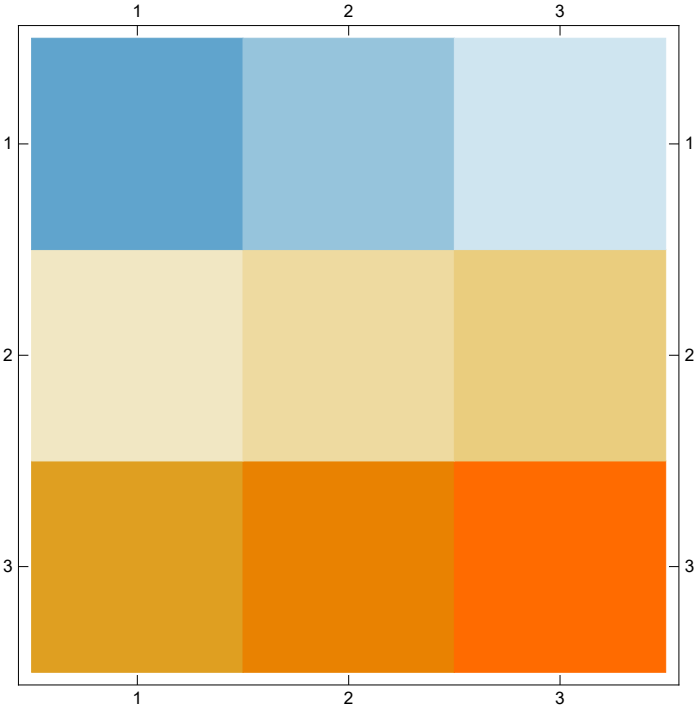
In[•]:=
```
ArrayPlot[{{1, 2, 3, 4}, {5, 6, 7, 8}}]
```

Out[•]=



In[ ]:=
```
MatrixPlot[{{-10, -5, -1}, {2, 4, 6}, {20, 30, 40}}]
```

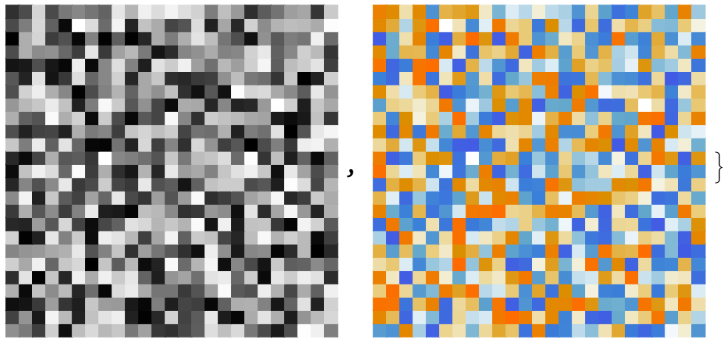
Out[ ]=



In[ ]:=
```
d = RandomInteger[{-100, 100}, {25, 25}];
```

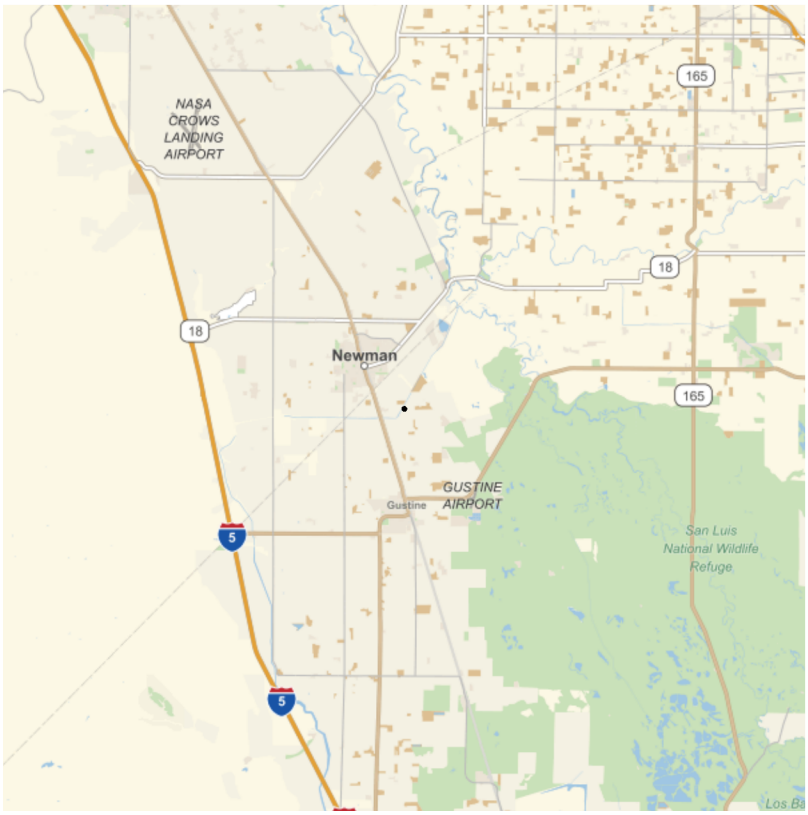
In[ ]:=
```
{ArrayPlot[d, Frame → False], MatrixPlot[d, Frame → False]}
```

Out[ ]=



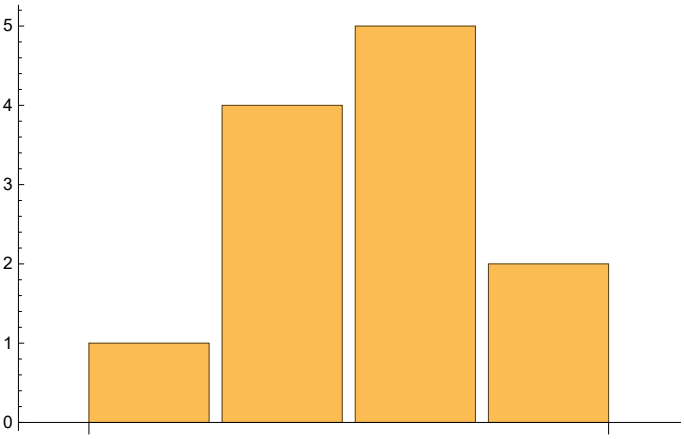
In[ ]:=
```
GeoGraphics[Point[GeoPosition[{37.2969, -121, 819}]]]
```

Out[ ]=



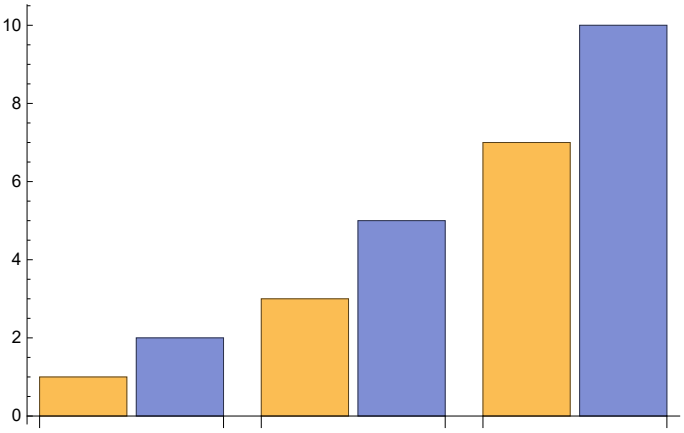
In[ ]:=
```
BarChart[{1, 4, 5, 2}]
```

Out[ ]=



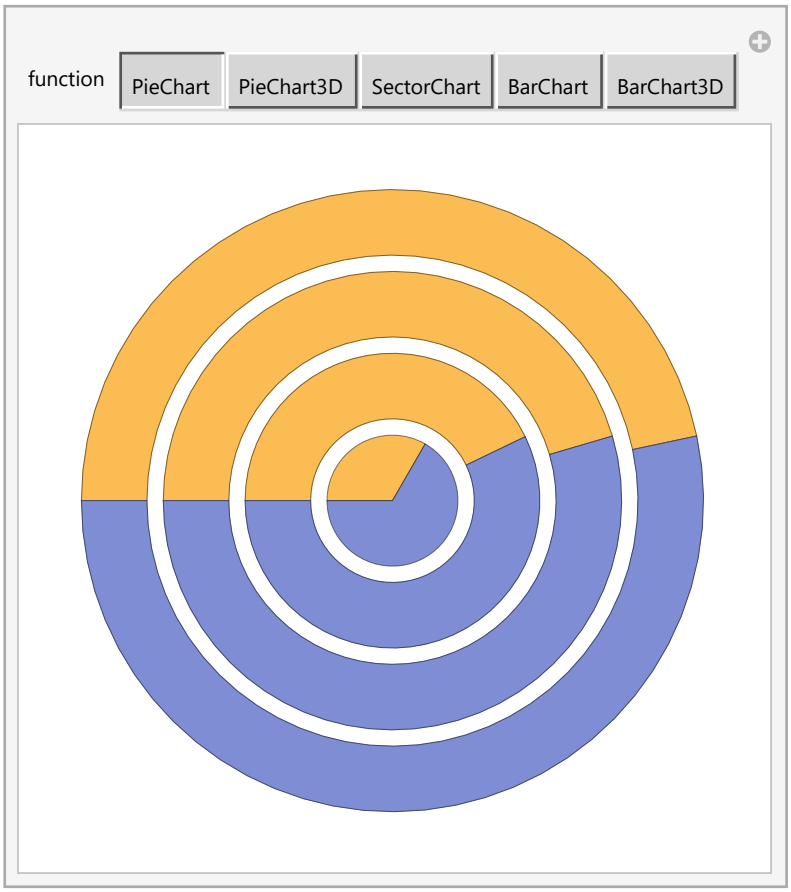
In[ ]:=
```
BarChart[{{1, 2}, {3, 5}, {7, 10}}]
```

Out[ ]=



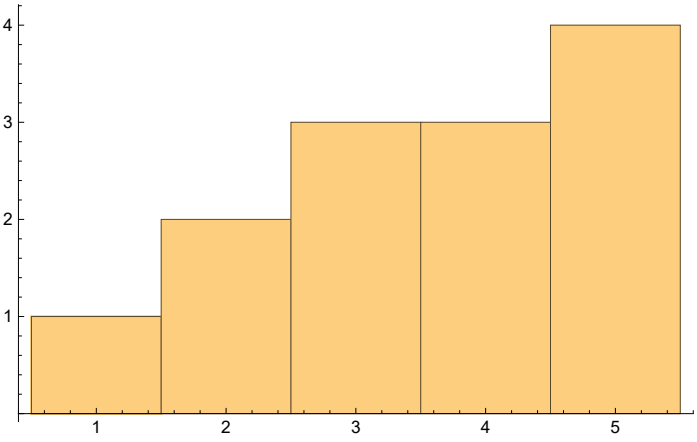
In[ ]:=
```
Manipulate[
 function[{{1, 2}, {3, 4}, {5, 6}, {7, 8}}],
 {function, {PieChart, PieChart3D, SectorChart, BarChart, BarChart3D}}]
```

Out[ ]=



In[ ]:=
```
Histogram[{1, 2, 2, 3, 3, 3, 4, 4, 4, 5, 5, 5, 5}]
```

Out[ ]=

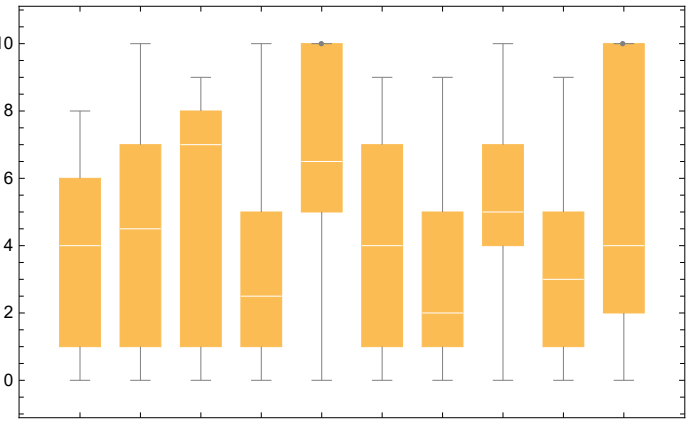


In[ ]:=
```
WordCloud[{J, A, N, A, R, S}]
```

Out[ ]=

In[ ]:=
```
BoxWhiskerChart[RandomInteger[{0, 10}, {10, 10}]]
```

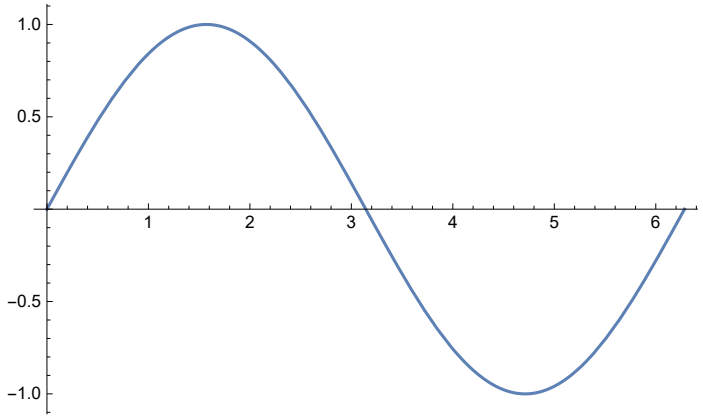
Out[ ]=



Chapter 12| Styling and Customizing Graphs

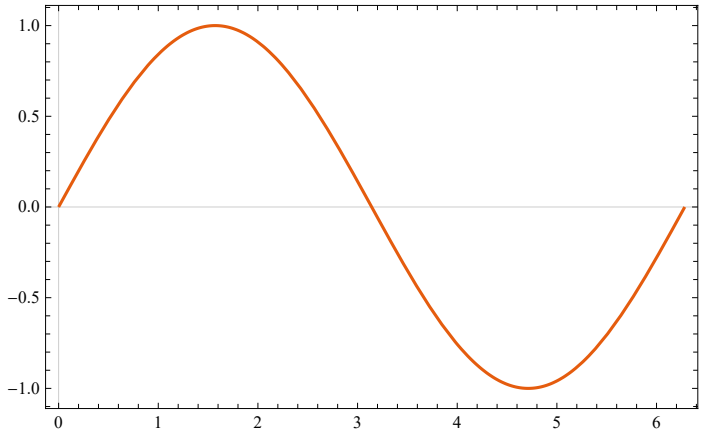
In[•]:=
```
Plot[Sin[x], {x, 0, 2 π}]
```

Out[•]=



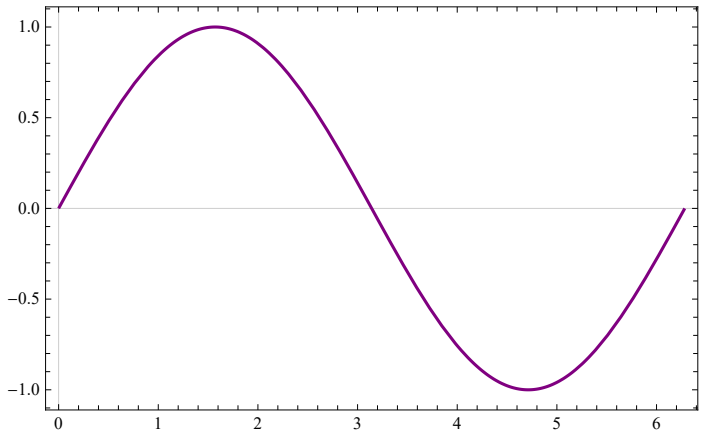
In[•]:=
```
Plot[Sin[x], {x, 0, 2 π}, PlotTheme → "Scientific"]
```

Out[•]=



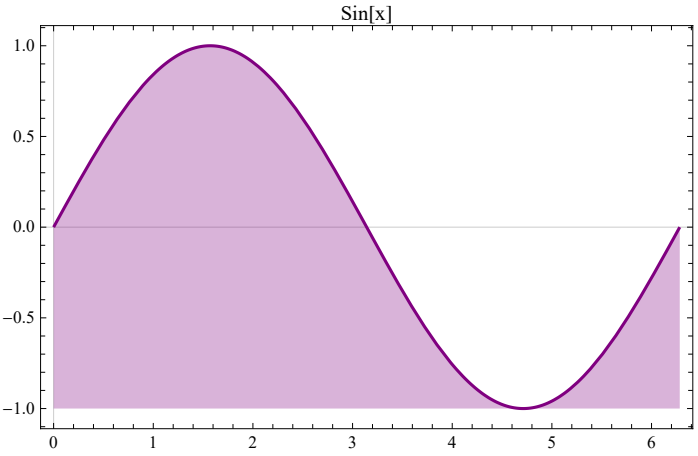
In[•]:=
```
Plot[Sin[x], {x, 0, 2 π}, PlotTheme → "Scientific", PlotStyle → Purple]
```

Out[•]=



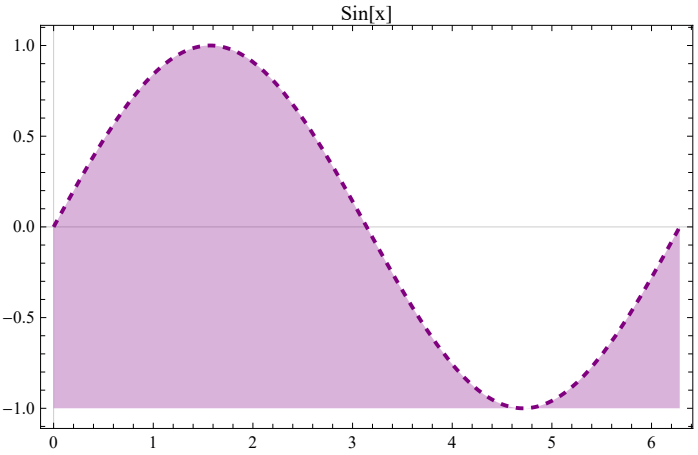
In[•]:=
```
Plot[Sin[x], {x, 0, 2 π}, PlotTheme → "Scientific",
 PlotStyle → Purple, Filling → Bottom, PlotLabel → "Sin[x]"]
```

Out[•]=



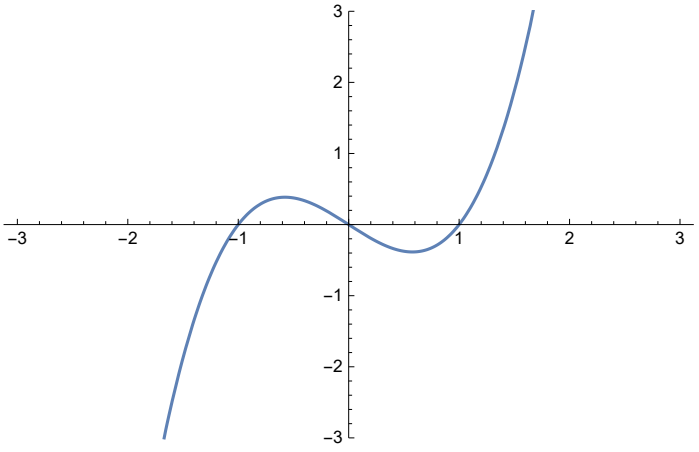
In[•]:=
```
Plot[Sin[x], {x, 0, 2 π}, PlotTheme → "Scientific",
 PlotStyle → {Purple, Thick, Dashed}, Filling → Bottom, PlotLabel → "Sin[x]"]
```

Out[•]=



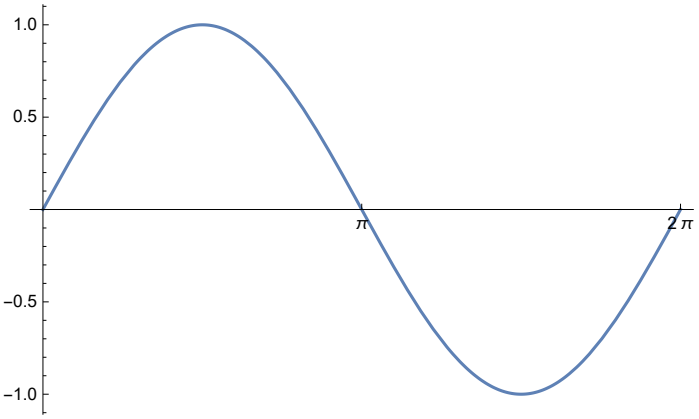
In[•]:=
```
Plot[x^3 - x, {x, -3, 3}, AxesOrigin → {0, 0}, PlotRange → {-3, 3}]
```

Out[•]=



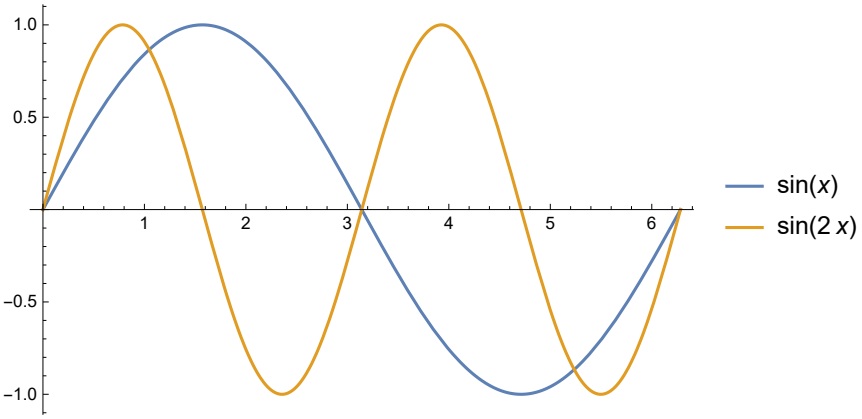
In[•]:=
```
Plot[Sin[x], {x, 0, 2 π}, Ticks → {{π, 2 π}, Automatic}]
```

Out[•]=



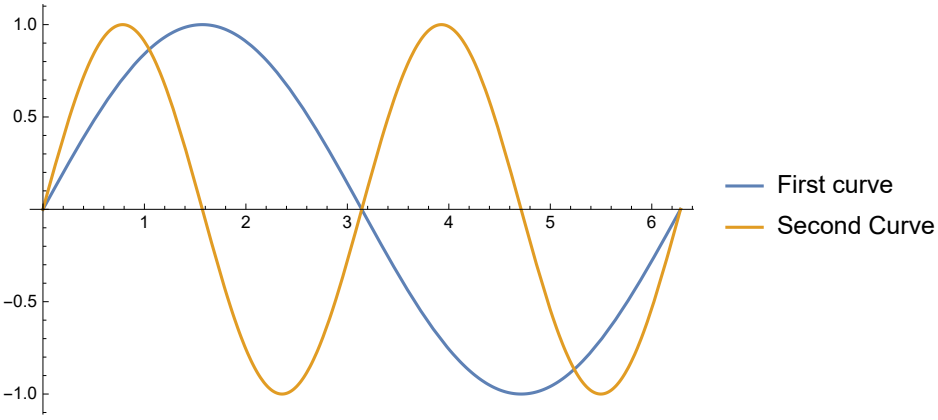
In[•]:=
```
Plot[{Sin[x], Sin[2 x]}, {x, 0, 2 π}, PlotLegends → "Expressions"]
```

Out[•]=



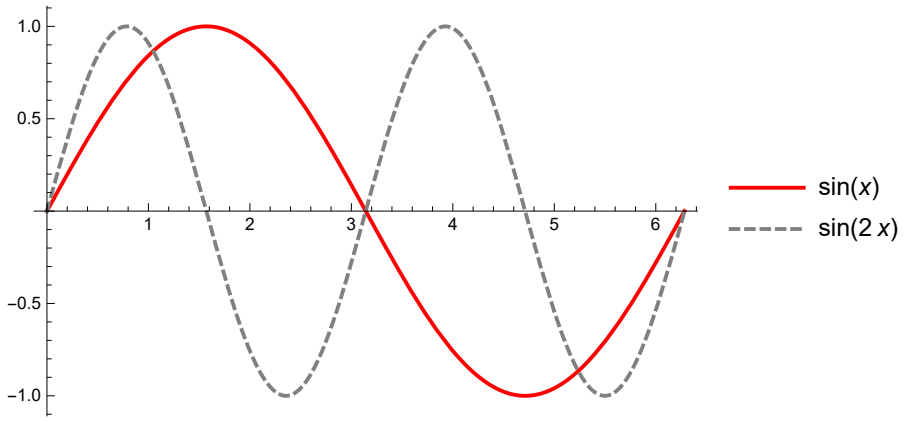
In[•]:=
```
Plot[{Sin[x], Sin[2 x]}, {x, 0, 2 π}, PlotLegends → {"First curve", "Second Curve"}]
```

Out[•]=



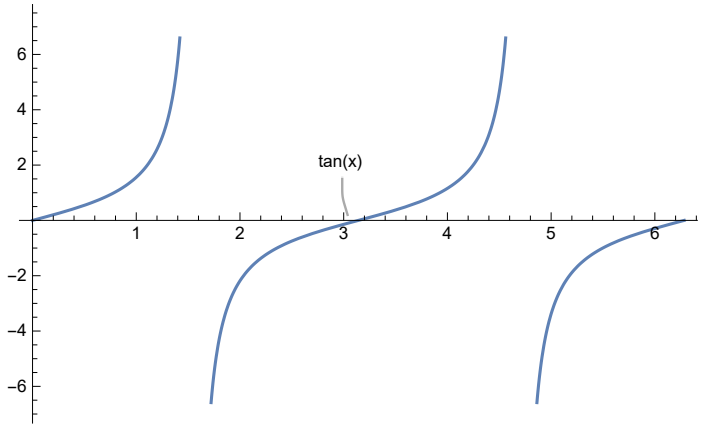
In[ ]:=
```
Plot[{Sin[x], Sin[2 x]}, {x, 0, 2 π},
 PlotStyle → {Directive[Red, Thick], Directive[Gray, Thick, Dashed]},
 PlotLegends → "Expressions"]
```

Out[ ]=



In[ ]:=
```
Plot[Callout[Tan[x], "tan(x)", π], {x, 0, 2 π}]
```

Out[ ]=



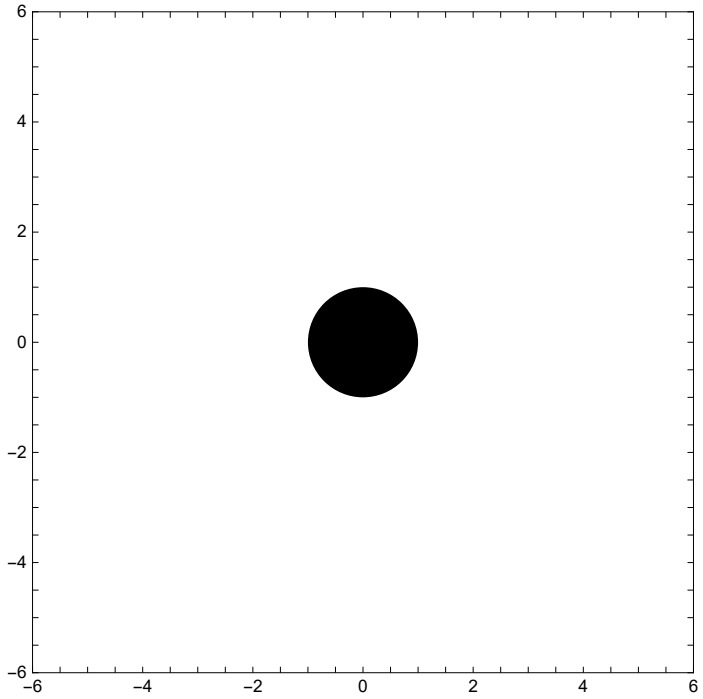
Chapter 13| Creating Figures and Diagrams with Graphics Primitives

In[ ]:=
```
Disk[]
```

Out[ ]=
```
Disk[{0, 0}]
```

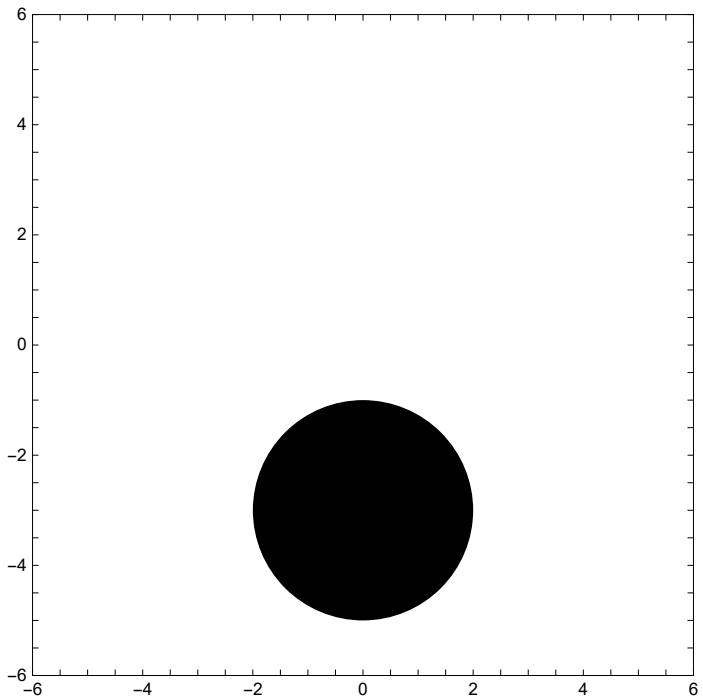
In[ ]:=
```
Graphics[Disk[], PlotRange → 6, Frame → True]
```

Out[ ]=



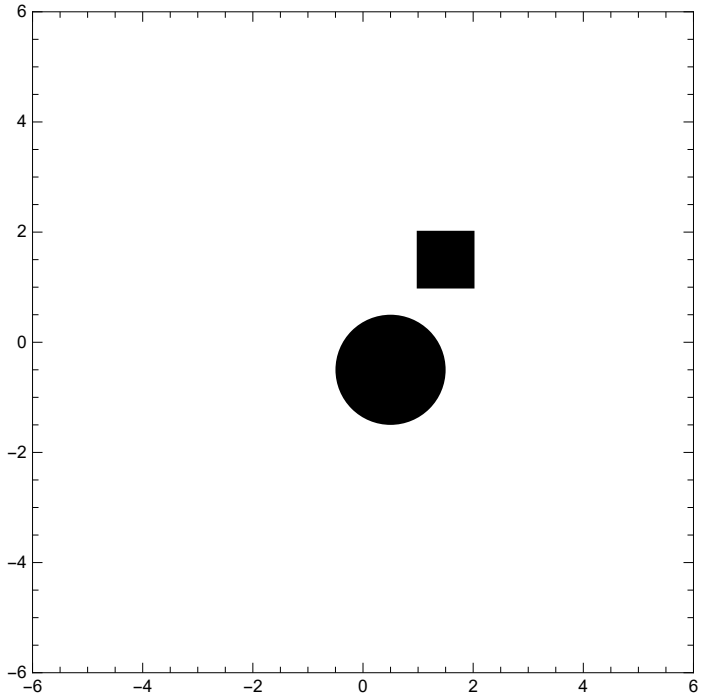
In[ ]:=
```
Graphics[Disk[{0, -3}, 2], PlotRange → 6, Frame → True]
```

Out[ ]=



In[ ]:=
```
Graphics[{Disk[{0.5, -0.5}, 1], Rectangle[{1, 1}, {2, 2}]}, PlotRange → 6, Frame → True]
```

Out[ ]=

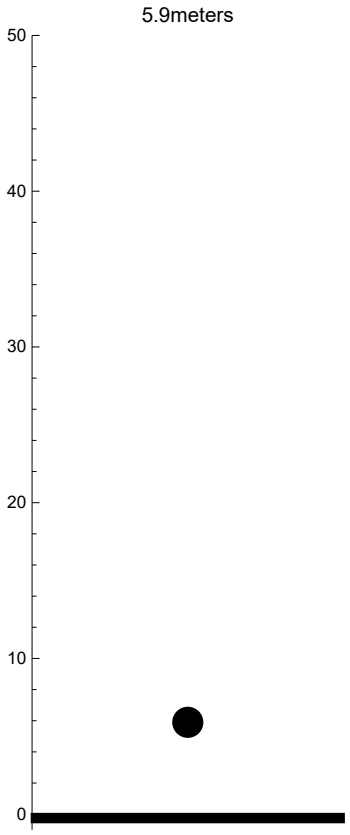


In[ ]:=
```
t = 2;
```

In[ ]:=
```
d = 1/2 (-9.8) t^2 + 50;
```

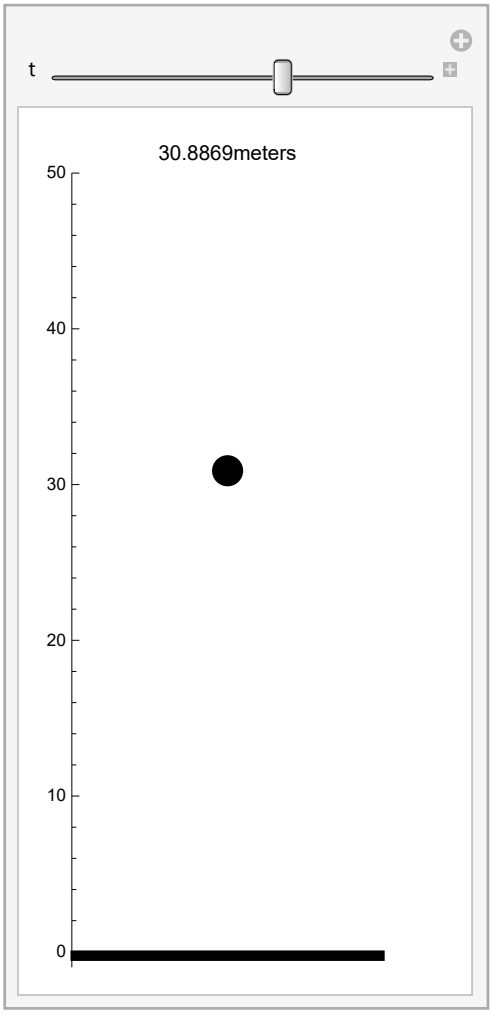
In[ ]:=
```
Graphics[{Disk[{10, d}], Rectangle[{0, -0.5}, {20, 0}]}, PlotRange → {{0, 20}, {-1, 50}},
 Axes → {False, True},
 PlotLabel → ToString[d] <> "meters"]
```

Out[ ]=



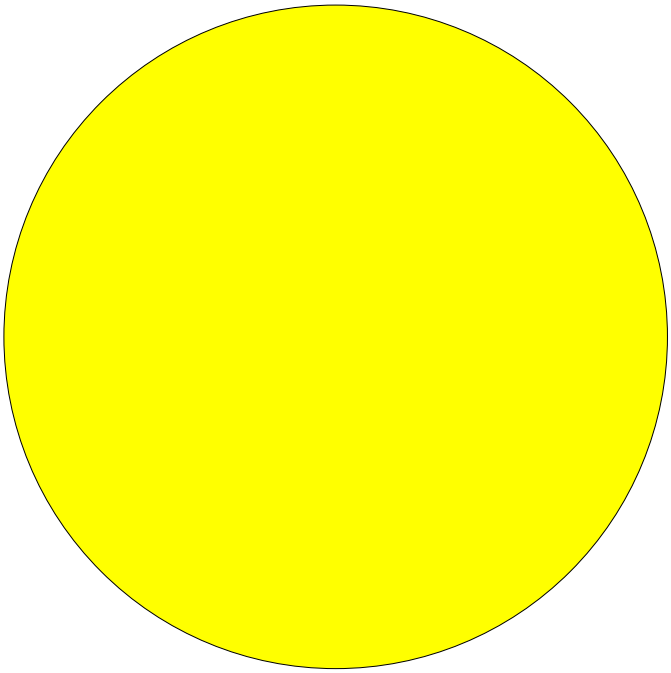
In[ ]:=
```
DynamicModule[{d, t},
 Manipulate[Graphics[
   {Disk[{10, d[t]}], Rectangle[{0, -0.5}, {20, 0}]}, PlotRange → {{0, 20}, {-1, 50}},
   Axes → {False, True},
   PlotLabel → ToString[d[t]] <> "meters"],
  {t, 0, 3.2}, Initialization :→ (d[t_] := 1/2 (-9.8) t^2 + 50)]]
```

Out[ ]=



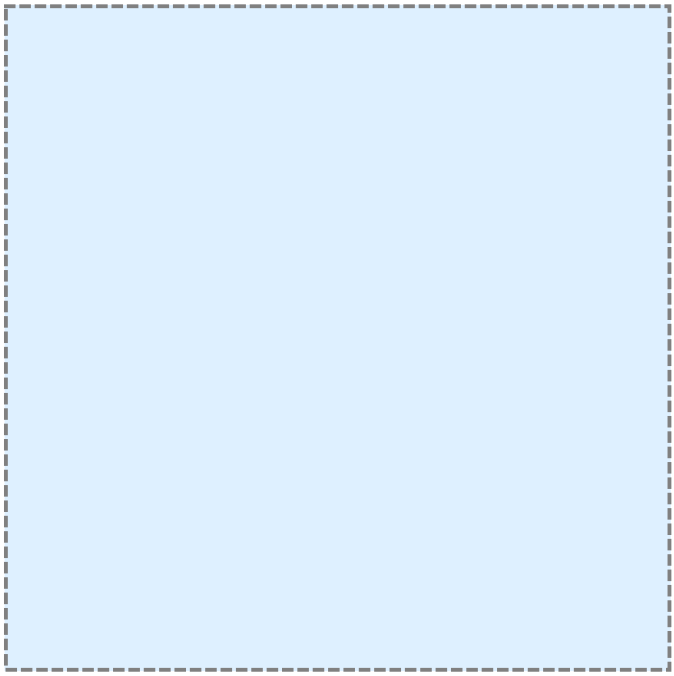
In[ ]:=
```
Graphics[{Yellow, EdgeForm[{Black}], Disk[]}]
```

Out[ ]=



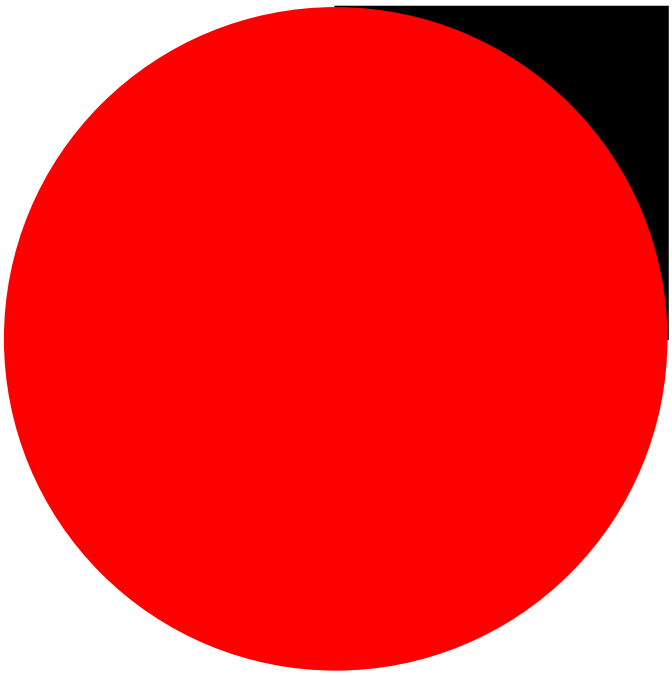
In[ ]:=
```
Graphics[{LightBlue, EdgeForm[{Gray, Thick, Dashed}], Rectangle[]}]
```

Out[ ]=



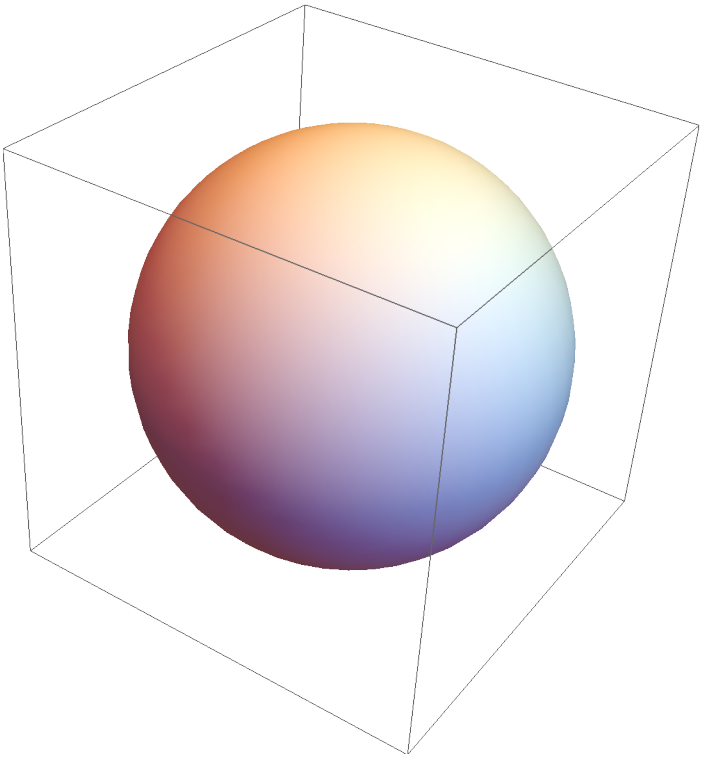
In[ ]:=
```
Graphics[{Black, Rectangle[{0, 0}, {1, 1}], Red, Disk[{0, 0}, 1]}]
```

Out[ ]=



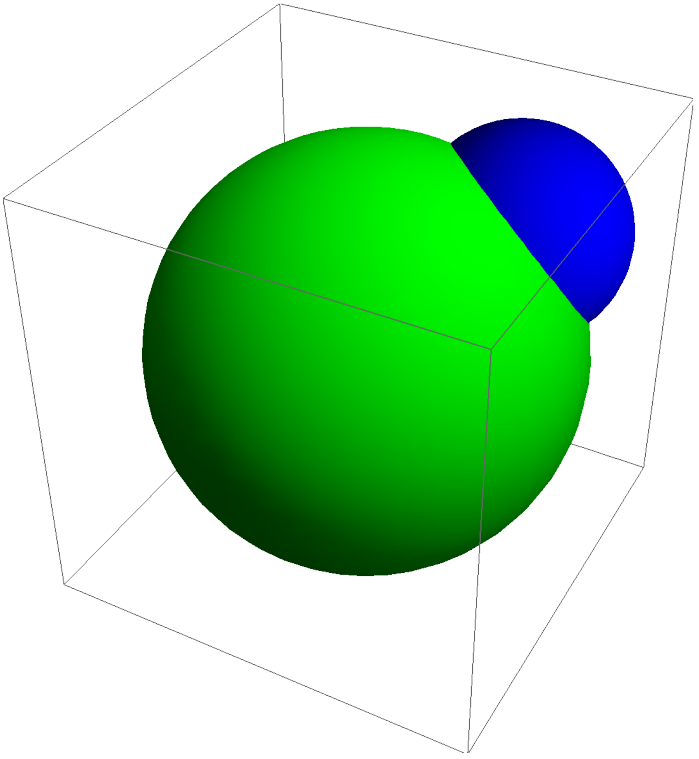
In[ ]:=
```
Graphics3D[Sphere[{0, 0, 0}, 0.5]]
```

Out[ ]=



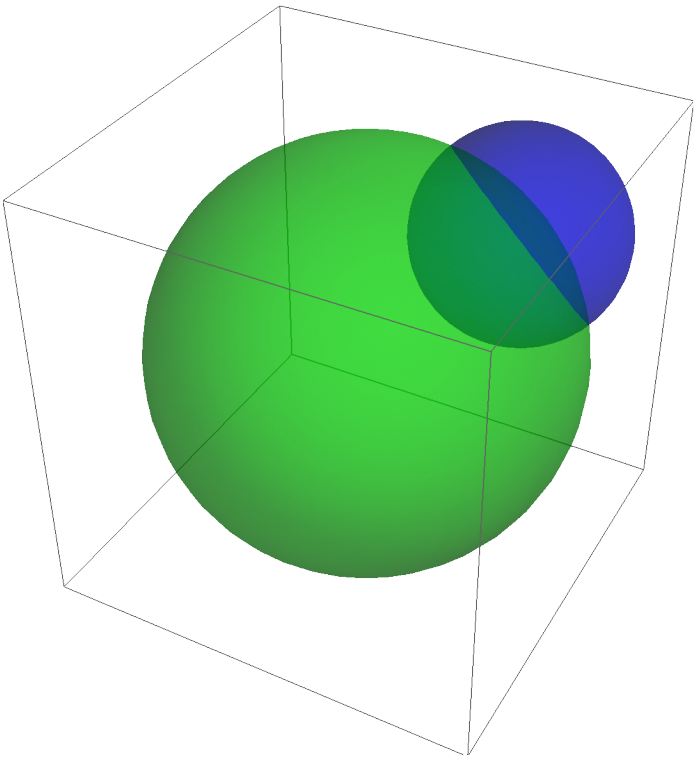
In[ ]:=
```
Graphics3D[{Blue, Sphere[{0, 0, 0}, 0.5], Green, Sphere[{-0.5, -0.5, -0.5}, 1]}]
```

Out[ ]=



In[ ]:=
```
Graphics3D[
 {Blue, Opacity[0.5], Sphere[{0, 0, 0}, 0.5], Green, Sphere[{-0.5, -0.5, -0.5}, 1]}]
```

Out[ ]=



Chapter 14| Algebraic Manipulation and Equation

In[•]:=
$$\frac{2\,a\,b}{b\,c}$$

Out[•]=
$$\frac{2\,a}{c}$$

In[•]:= `Expand[(a + b) (a + c) (b + c)]`

Out[•]= $a^2\,b + a\,b^2 + a^2\,c + 2\,a\,b\,c + b^2\,c + a\,c^2 + b\,c^2$

In[•]:= `Factor[a^2 b + a b^2 + a^2 c + 2 a b c + b^2 c + a c^2 + b c^2]`

Out[•]= $(a + b)\,(a + c)\,(b + c)$

In[•]:= `Togther[1/(x + 1) + 1/(x - 1)]`

Out[•]= `Togther[1/(-1 + x) + 1/(1 + x)]`

In[•]:= `Apart[2 x/((-1 + x) (1 + x))]`

Out[•]=
$$\frac{1}{-1 + x} + \frac{1}{1 + x}$$

In[•]:= `Collect[a x^2 + b x^2 y + c y x, y]`

Out[•]= $a\,x^2 + (b\,x^2 + c\,x)\,y$

In[•]:= `Simplify[Sin[x]^2 + Cos[x]^2]`

Out[•]= 1

In[•]:= `Simplify[Sqrt[x^2], x > 0]`

Out[•]= x

In[•]:= `Solve[x^2 + 2 x - 1 == 0, x]`

Out[•]= $\{\{x \to -1 - \sqrt{2}\}, \{x \to -1 + \sqrt{2}\}\}$

In[•]:= `ReplaceAll[{x, x + 1, x + 2}, x → 2]`

Out[•]= {2, 3, 4}

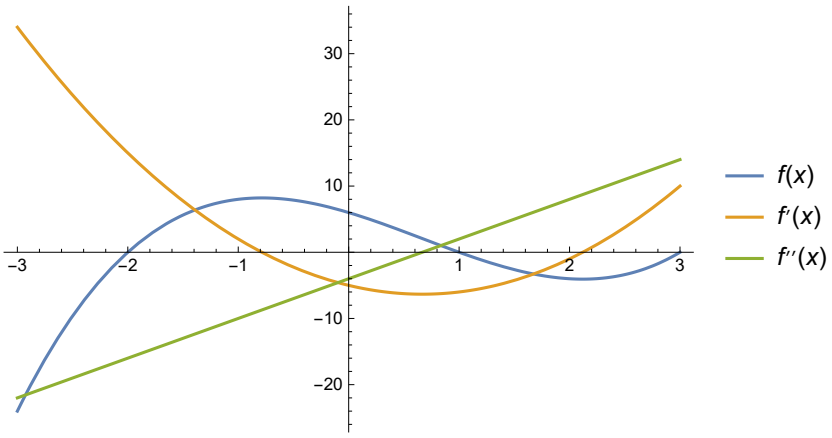
In[•]:= `FindRoot[Sin[x^2] - Cos[x], {x, π}]`

Out[•]= {x → 3.29304}

Chapter 15| Calculus

```
In[•]:= D[x^2 Sin[x], x] (* Diffrentation *)
Out[•]= x^2 Cos[x] + 2 x Sin[x]

In[•]:= D[x^2 Sin[x], {x, 3}]
Out[•]= 6 Cos[x] - x^2 Cos[x] - 6 x Sin[x]

In[•]:= Sin'[x]
Out[•]= Cos[x]

In[•]:= f[x_] := x^3 - 2 x^2 - 5 x + 6

In[•]:= f'[x]
Out[•]= -5 - 4 x + 3 x^2

In[•]:= f''[x]

In[•]:= -4 + 6 x
Out[•]= -4 + 6 x

In[•]:= Plot[{f[x], f'[x], f''[x]}, {x, -3, 3}, PlotLegends → "Expressions"]
```

Out[•]=




```
In[•]:= Limit[1/x, x → 1]
Out[•]= 1

In[•]:= Limit[1/x, x → Infinity]
Out[•]= 0
```

In[]:= `Integrate[x^2 + 2 x + 1, x]`

Out[]= $x + x^2 + \frac{x^3}{3}$

In[]:= $\int (x^2 + 2x + 1)\, dx$

Out[]= $x + x^2 + \frac{x^3}{3}$

In[]:=
```
Manipulate[
 Integrate[x^2 e^x, {x, 0, a}],
 {a, 0, 8}]
```

Out[]=

In[]:= `Clear[x, f]`

Chapter 16| Differential Equations

In[]:= `DSolve[y'[x] == x^2 Sin[x], y[x], x]`

Out[]= $\{\{y[x] \to c_1 - (-2 + x^2)\,\text{Cos}[x] + 2x\,\text{Sin}[x]\}\}$

In[]:= `DSolve[{y'[x] == x^2 Sin[x], y[1] == 1}, y[x], x]`

Out[]= $\{\{y[x] \to 1 - \text{Cos}[1] + 2\,\text{Cos}[x] - x^2\,\text{Cos}[x] - 2\,\text{Sin}[1] + 2x\,\text{Sin}[x]\}\}$

In[]:= `NDSolve[{y'[x] == x^2 Sin[x], y[1] == 1}, y, {x, 0, 10}]`

Out[]= {{y → InterpolatingFunction[Domain: {{0., 10.}} Output: scalar]}}

In[]:= `nSlon = NDSolveValue[{y'[x] == x^2 Sin[x], y[1] == 1}, y, {x, 0, 10}]`

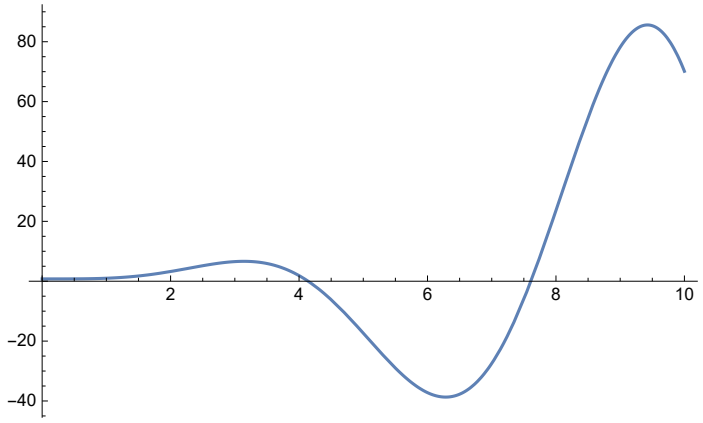
Out[]= InterpolatingFunction[Domain: {{0., 10.}} Output: scalar]

In[]:= `nSlon[5]`

Out[]= -17.3367

In[•]:= **Plot[nSlon[X], {X, 0, 10}]**

Out[•]=



In[•]:= **Clear[nSlon]**

Chapter 17 | Linear Algebra

In[•]:= **vec1 = {1, 2, 3, 4, 57, 9, 34, 8, 9, 0} (* vectors*)**

{1, 2, 3, 4, 57, 9, 34, 8, 9, 0} mjnn

In[•]:= **vec2 = Table[i², {i, 1, 10}]**

Out[•]= {1, 4, 9, 16, 25, 36, 49, 64, 81, 100}

In[•]:= **myFunction[x_] := x Sin[x]**

In[•]:= **vec3 = Array[myFunction, 5]**

Out[•]= {Sin[1], 2 Sin[2], 3 Sin[3], 4 Sin[4], 5 Sin[5]}

In[•]:= **vec4 = {a, c, π, e, 1, 2, 3, 4.5}**

Out[•]= {a, c, π, e, 1, 2, 3, 4.5}

In[•]:= **{VectorQ[vec1], VectorQ[vec2], VectorQ[vec3], VectorQ[vec4]}**

Out[•]= {True, True, True, True}

In[•]:= **vec1 + vec2**

Out[•]= {2, 6, 12, 20, 82, 45, 83, 72, 90, 100}

In[•]:= **Cross[{1, 3, 5}, {π, e, 0}]**

Out[•]= {−5 e, 5 π, e − 3 π}

In[•]:= **Norm[vec3]**

Out[•]= $\sqrt{\text{Sin}[1]^2 + 4\,\text{Sin}[2]^2 + 9\,\text{Sin}[3]^2 + 16\,\text{Sin}[4]^2 + 25\,\text{Sin}[5]^2}$

In[•]:= **mat1 = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};**

```
In[•]:= MatrixForm[mat1]
```

Out[•]//MatrixForm=

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$

```
In[•]:= mat2 = Table[i * j, {i, 1, 5}, {j, 1, 5}];
```

```
In[•]:= MatrixForm[mat2]
```

Out[•]//MatrixForm=

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 4 & 6 & 8 & 10 \\ 3 & 6 & 9 & 12 & 15 \\ 4 & 8 & 12 & 16 & 20 \\ 5 & 10 & 15 & 20 & 25 \end{pmatrix}$$

```
In[•]:= myList = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
```

```
In[•]:= Part[myList, 5]
```

```
Out[•]= 5
```

```
In[•]:= myList[[5]]
```

```
Out[•]= 5
```

```
In[•]:= Part[myList, 1 ;; 5]
```

```
Out[•]= {1, 2, 3, 4, 5}
```

```
In[•]:= Take[myList, {3, 6}]
```

```
Out[•]= {3, 4, 5, 6}
```

```
In[•]:= Take[( 1  2
               4  5 ), 2]
```

```
Out[•]= {{1, 2}, {4, 5}}
```

```
In[•]:= Diagonal[( 1  2
                   4  6 )] // MatrixForm
```

Out[•]//MatrixForm=

$$\begin{pmatrix} 1 \\ 6 \end{pmatrix}$$

```
In[•]:= Transpose[( 1  2
                    4  6 )] // MatrixForm
```

Out[•]//MatrixForm=

$$\begin{pmatrix} 1 & 4 \\ 2 & 6 \end{pmatrix}$$

```
In[•]:= Inverse[( 1  2
                  4  6 )]
```

```
Out[•]= {{-3, 1}, {2, -1/2}}
```

In[ ]:= `Eigenvalues[{{1, 2}, {4, 6}}]`

Out[ ]= $\left\{\frac{1}{2}\left(7+\sqrt{57}\right), \frac{1}{2}\left(7-\sqrt{57}\right)\right\}$

Chapter 18| Probabiity and Statics

In[ ]:=

pooker full house

↳ Properties

Out[ ]=

	number of possible hands	approximate probability	approximate chance
5-card hand	3744	0.001441	≈ 1 in 694
7-card hand	3 473 184	0.02596	≈ 1 in 39

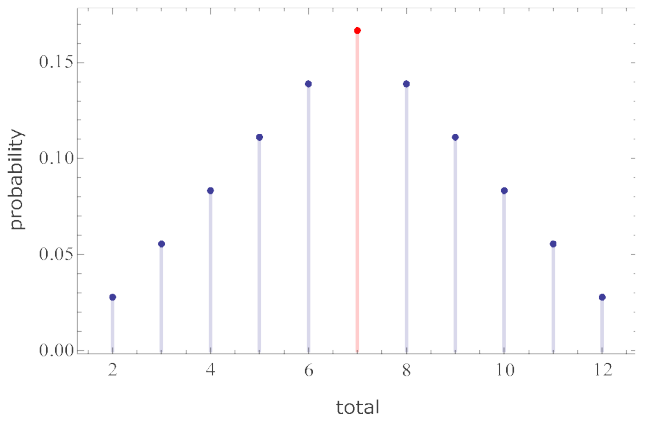
(assuming random selection from a standard 52-card deck)
(the value of a 7-card hand is determined by its best 5-card subset)

In[ ]:=

rolling a 7 on two 6 – sided dice

↳ Distribution of total

Out[ ]=



(assuming fair 6-sided dice)

In[ ]:= `Probability[x == 3, x ≈ DiscreteUniformDistribution[{1, 6}]]`

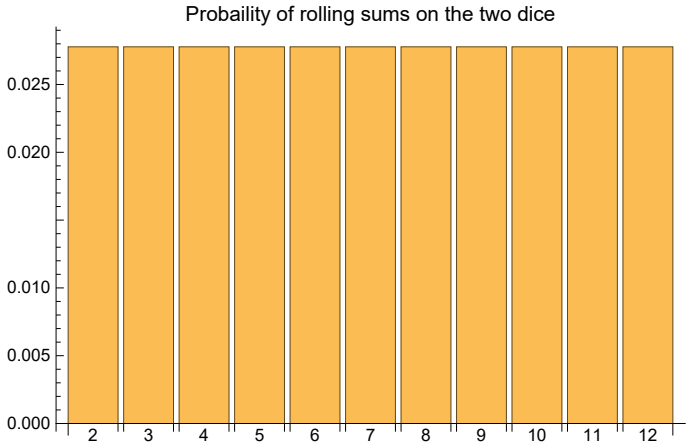
Out[ ]= $\frac{1}{6}$

In[•]:=
```
probs = Table[
  Probability[x + y == 12, x ≃ DiscreteUniformDistribution[{1, 6}] &&
    y ≃ DiscreteUniformDistribution[{1, 6}]],
  {result, 2, 12, 1}]
```

Out[•]= $\left\{\frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}, \frac{1}{36}\right\}$

In[•]:=
```
BarChart[probs, ChartLabels → Table[i, {i, 2, 12, 1}],
 PlotLabel → "Probaility of rolling sums on the two dice"]
```

Out[•]=

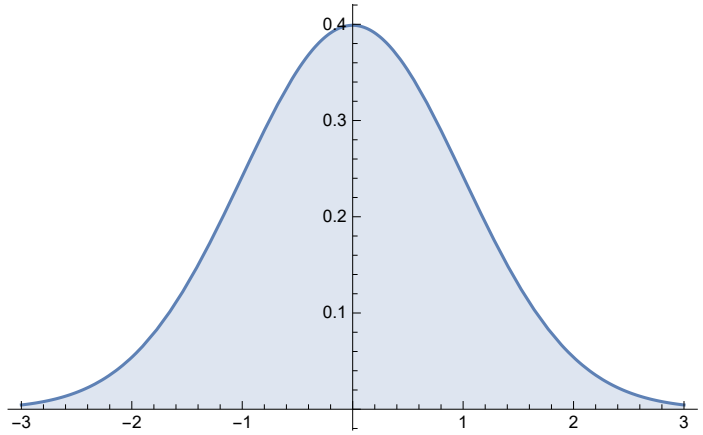


In[•]:=
```
PDF[DiscreteUniformDistribution[{1, 6}], x]
```

Out[•]= $\begin{cases} \frac{1}{6} & 1 \le x \le 6 \\ 0 & \text{True} \end{cases}$

In[•]:=
```
Plot[PDF[NormalDistribution[0, 1], x], {x, -3, 3}, Filling → Axis]
```

Out[•]=



In[•]:=
```
myFun[x_] := PDF[NormalDistribution[0, 1], x]
```

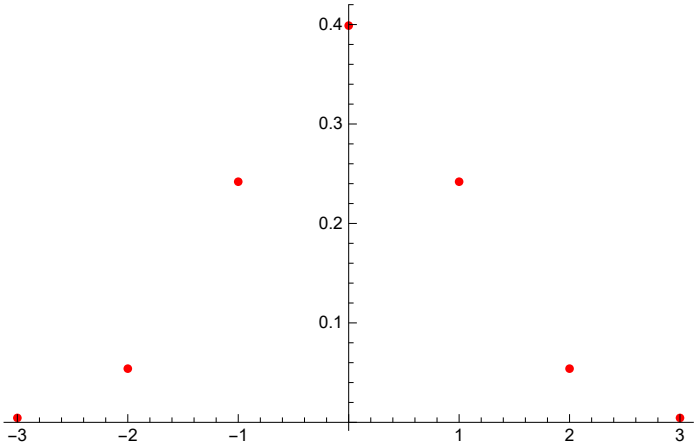
In[•]:=
```
myFun[0]
```

Out[•]= $\frac{1}{\sqrt{2\,\pi}}$

In[•]:=
```
points = Table[{x, myFun[x]}, {x, -3, 3, 1}];
```

In[ ]:=
```
ListPlot[points, PlotStyle → {Red, PointSize[Medium]}]
```

Out[ ]=



In[ ]:=
```
data = {1, 2, 2, 3, 4, 5, 6, 6, 7, 8, 8, 8, 7}
Mean[data]
```

Out[ ]= {1, 2, 2, 3, 4, 5, 6, 6, 7, 8, 8, 8, 7}

Out[ ]= $\frac{67}{13}$

In[ ]:=
```
Median[data]
```

Out[ ]= 6

In[ ]:=
```
Commonest[data]
```

Out[ ]= {8}

In[ ]:=
```
Mean[{p1, p2, p3}]
```

Out[ ]= $\frac{1}{3}\left(p1 + p2 + p3\right)$

In[ ]:=
```
Variance[data]
```

Out[ ]= $\frac{82}{13}$

In[ ]:=
```
StandardDeviation[data]
```

Out[ ]= $\sqrt{\frac{82}{13}}$

In[ ]:=
```
InterquartileRange[data]
```

Out[ ]= $\frac{9}{2}$

In[ ]:=
```
list1 = {1, 2, 2, 3, 3, 4, 5, 5, 5, 7};
list2 = {1, 3, 4, 5, 6, 7, 8, 9, 0, 1};
```

```
In[ ]:= Covariance[list1, list2]
```

$$\text{Out[ ]=}\ \frac{26}{45}$$

```
In[ ]:= Correlation[list1, list2]
```

$$\text{Out[ ]=}\ 2\sqrt{\frac{13}{5117}}$$

```
In[ ]:= myData = CountryData["Iceland", {{"GDP"}, {1970, 2010}}]
```

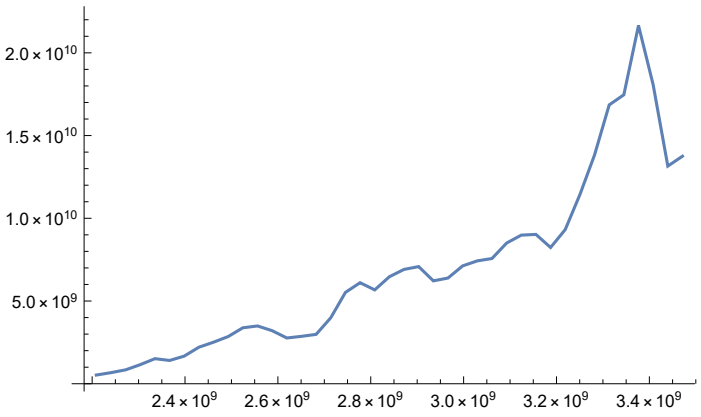
Out[ ]= TimeSeries[Time: 01 Jan 1970 to 01 Jan 2010 Data points: 41]

```
In[ ]:= myData["Values"]
```

Out[ ]= QuantityArray[Dimensions: {41} Unit: USDollars / Years]

```
In[ ]:= ListLinePlot[myData]
```

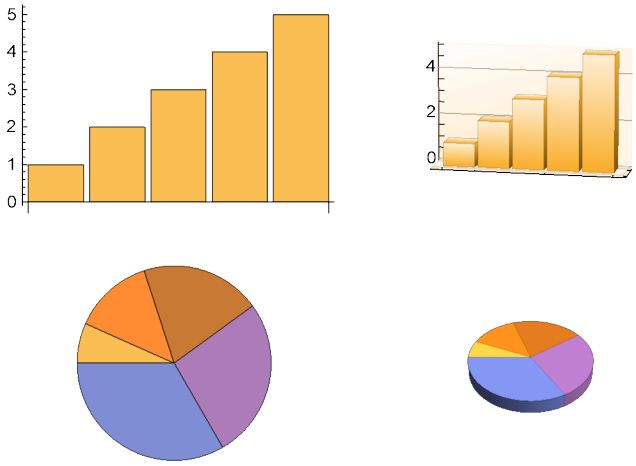
Out[ ]=




```
In[ ]:= data = {1, 2, 3, 4, 5};
```

In[ ]:=
```
GraphicsGrid[{
  {BarChart[data], BarChart3D[data]},
  {PieChart[data], PieChart3D[data]}
 }]
```

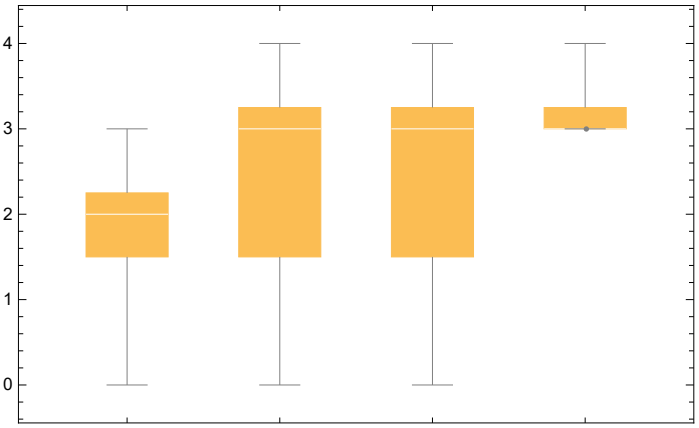
Out[ ]=



In[ ]:=
```
boxData = RandomInteger[{0, 4}, {4, 5}];
```

In[ ]:=
```
BoxWhiskerChart[boxData]
```

Out[ ]=

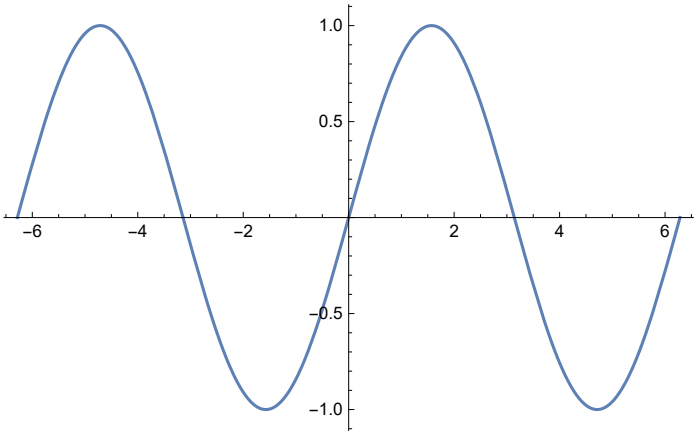


Chapter 19| Importing and Exporting Data

```
(* Import[] *)
```

In[2]:=
```
Plot[Sin[x], {x, -2 π, 2 π}]
```

Out[2]=



In[3]:=
```
Export["data.png", Plot[Sin[x], {x, -2 π, 2 π}]]
```

Out[3]=
```
data.png
```

Chapter 20| Filtering and Manipulation

In[4]:=
```
Part[{1, 2, 3, 45, 67, 8}, 4]
```

Out[4]=
```
45
```

In[5]:=
```
{1, 3, 56, 78, 90, 88}[[3]]
```

Out[5]=
```
56
```

In[6]:=
```
listOfScores = Import["http://www.handsonstart.com/ExampleDataScores.txt", "Data"]
```

Out[6]=
```
{{Joe, Smith, 94}, {Jane, Smith, 85},
 {Bob, Example, 82}, {Bill, Student, 83}, {Michelle, Abacus, 98}}
```

In[7]:=
```
listOfScores[[1]]
```

Out[7]=
```
{Joe, Smith, 94}
```

In[8]:=
```
listOfScores[[1, 3]]
```

Out[8]=
```
94
```

In[9]:=
```
TableForm[listOfScores]
```

Out[9]//TableForm=

Joe	Smith	94
Jane	Smith	85
Bob	Example	82
Bill	Student	83
Michelle	Abacus	98

In[10]:=
```
listOfScores[[3, All]]
```

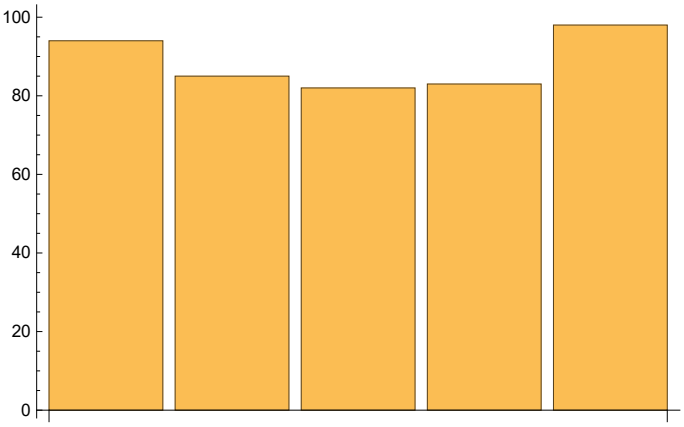
Out[10]=
```
{Bob, Example, 82}
```

In[11]:=
```
listOfScores[[All, 3]]
```

Out[11]=
```
{94, 85, 82, 83, 98}
```

In[12]:=
```
BarChart[listOfScores[[All, 3]]]
```

Out[12]=



newScore =

In[13]:=
```
{listOfScores, {"Jana", "Anas", 95}}
```

Out[13]=
```
{{{Joe, Smith, 94}, {Jane, Smith, 85}, {Bob, Example, 82},
  {Bill, Student, 83}, {Michelle, Abacus, 98}}, {Jana, Anas, 95}}
```

Chapter 21| Working with Curated Data

In[14]:=

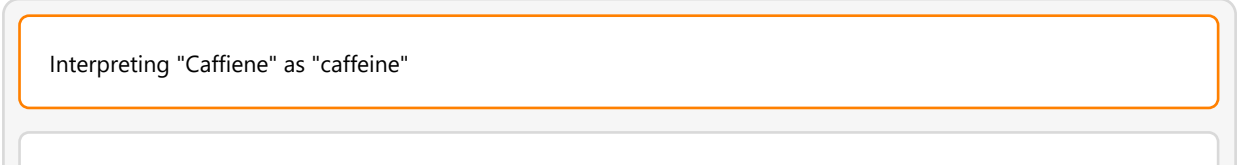
Out[14]=
```
38 019 178 people
```

In[15]:=
```
CountryData["Canada", "Airports"]
```

Out[15]=
```
1467
```

Chapter 22| Using Wolfram Alpha Data in Mathematica

In[16]:=
```
WolframAlpha["Caffiene"]
```



Assuming "Caffiene" is a chemical compound | Use as a word or a movie instead

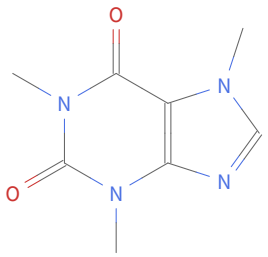
Input interpretation:

caffeine

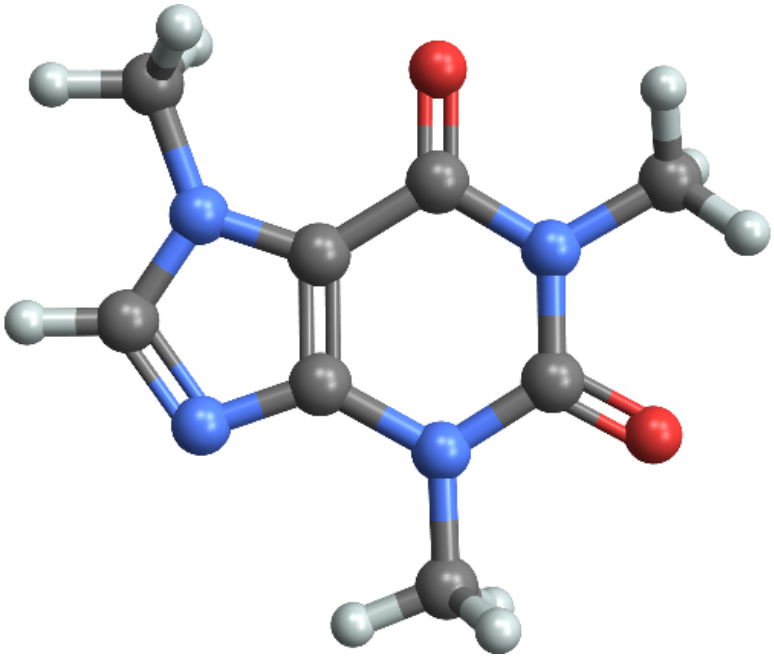
Chemical names and formulas: More

formula	$C_8H_{10}N_4O_2$
name	caffeine
IUPAC name	1,3,7−trimethylpurine−2,6−dione

Structure diagram: Skeletal structure | Show bond information | Show graph properties | Step−by−step



3D structure: Show bonds only | Show space filling model



Basic properties: Step−by−step

molar mass	194.19 g/mol
phase	solid (at STP)
melting point	235.3 °C
mass density	1.23 g/cm^3

+ Units

Out[16]=

Hydrophobicity and permeability properties:

experimental LogP hydrophobicity	−0.5
predicted LogP hydrophobicity	−0.23
experimental LogS	−0.97
predicted LogS	−1.25
experimental Caco-2 permeability	−4.41

Basic drug properties: More

approval status	approved | small molecule
drug categories	anorexigenic agent | central nervous system stimulant | phosphodiesterase inhibitor
dosage forms	oral: capsule | oral: elixir | oral: liquid | oral: pill | oral: solution | oral: solution / drops | rectal: suppository | oral: suspension | oral: syrup | oral: tablet | oral: tablet, extended release

Solid properties (at STP):

mass density	1.23 g/cm^3
vapor pressure	150 mmHg (at 80 °C)
refractive index	1.494

Thermodynamic properties: More

specific heat of fusion	0.11 kJ/g

(at STP)

+ Units

Chemical identifiers: More

CAS registry number	58−08−2

Beilstein number	17705
PubChem compound ID	2519
PubChem substance ID	148854
SMILES identifier	CN1C=NC2=C1C(=O)N(C(=O)N2C)C

NFPA label: Table

Safety properties: More

autoignition point	600 °C

Toxicity properties: More

lethal dosage	192 mg/kg (oral dose for rats)

+ Units

WolframAlpha

Chapter 23| Statistical Functionality for Data Analysis

Chapter 24| Creating Programs

In[17]:=
```
Do[Print[x], {x, 1, 5}]
```

```
1
2
3
4
5
```

In[18]:=
```
For [i = 1, i ≤ 5, i++, Print[i]]
```

```
1
2
3
4
5
```

In[19]:=
```
If[π > 3, "My test succeeded", " My test failed!"]
```

Out[19]=
```
My test succeeded
```

In[20]:=
```
Which[π > 5, "Greater than 5", π > 3, " Greater than 3", π > 1, " Greater than 1"]
```

Out[20]=
```
 Greater than 3
```

Chapter 25| Creating Parallel and GPU Programs