

Course

## Practicalities of Using the Wolfram Language

- 1 Starting Out: Elementary Arithmetic
- 2 Introducing Functions
- 3 First Look at Lists
- 4 Displaying Lists
- 5 Operations on Lists
- 6 Making Tables
- 7 Colors and Styles
- 8 Basic Graphics Objects
- 9 Interactive Manipulation
- 10 Images
- 11 Strings and Text
- 12 Sound
- 13 Arrays, or Lists of Lists
- 14 Coordinates and Graphics
- 15 Scope of the Wolfram Language
- 16 Real-World Data
- 17 Units
- 18 Geocomputation

## 1 | Starting Out: Elementary Arithmetic



Scratch Notebook

Work on your code here.

Page 12 / 3

```
In[76]:= ExampleData[{"TestImage", "Mandrill"}]
```



This website uses cookies to optimize your

Book Text	Transcript	Exercises
-----------	------------	-----------

$$m[15] = 1 * 2 * 3 * 4 * 5$$

Out[15]= 120

```

... AbsoluteTime: Warning: the interpretation of the string 04/01/2032 5:56 pm as a date is ambiguous.
... AbsoluteTime: Warning: the interpretation of the string 04/01/2032 8:41 pm as a date is ambiguous.
... AbsoluteTime: Warning: the interpretation of the string 04/01/2032 5:56 pm as a date is ambiguous.
... General: Further output of AbsoluteTime::ambig will be suppressed during this calculation.

```

1.4 Compute 5 squared (i.e. 5x5 or 5 raised to the power 2).

5^2

CHECK MY SOLUTION

**1.5** Compute 3 raised to the fourth power.

 $3^4$ [CHECK MY SOLUTION](#)

**1.6** Compute 10 raised to the power 12 (a trillion).

 $10^{12}$ 

CHECK MY SOLUTION

Stephen Wolfram | An Elementary Introduction to the Wolfram Language

Operations on Lists

Book Text

Transcript

Exercises

Course

Practicalities of Using the Wolfram Language

1 Starting Out: Elementary Arithmetic

2 Introducing Functions

3 First Look at Lists

4 Displaying Lists

5 Operations on Lists

6 Making Tables

7 Colors and Styles

8 Basic Graphics Objects

9 Interactive Manipulation

10 Images

11 Strings and Text

12 Sound

13 Arrays, or Lists of Lists

14 Coordinates and Graphics

15 Scope of the Wolfram Language

16 Real-World Data

17 Units

18 Computation

5 | Operations on Lists

AN ELEMENTARY INTRODUCTION TO THE  
Wolfram Language

05:16

Scratch Notebook

Work on your code here

12 / 3

ExampleData[{"TestImage", "Mandrill"}]

This website uses cookies to enhance your browsing experience, to analyze site usage, and to assist in our marketing efforts. (Click here to learn more)

Exercises for Section 5 | Operations on Lists

5.1 Make a list of the first 10 squares, in reverse order.  
EXPECTED OUTPUT »  
{100, 81, 64, 49, 36, 25, 16, 9, 4, 1}

Reverse[Range[10]^2]  
{100, 81, 64, 49, 36, 25, 16, 9, 4, 1}

5.2 Find the total of the first 10 squares.  
EXPECTED OUTPUT »  
385

Total[Range[10]^2]

5.3 Make a plot of the first 10 squares, starting at 1.  
EXPECTED OUTPUT »  
100

Stephen Wolfram | An Elementary Introduction to the Wolfram Language

Images

3 First Look at Lists

4 Displaying Lists

5 Operations on Lists

6 Making Tables

7 Colors and Styles

8 Basic Graphics Objects

9 Interactive Manipulation

10 Images

11 Strings and Text

12 Sound

13 Arrays, or Lists of Lists

14 Coordinates and Graphics

15 Scope of the Wolfram Language

16 Real-World Data

17 Units

18 Geocomputation

19 Dates and Times

20 Options

21 Graphs and Networks

22 Machine Learning

10 | Images


05:22

Scratch Notebook

Work on your code here

```
in[1]:= 12 / 3
```

```
in[76]:= ExampleData[{"TestImage", "Handrill"}]
```




This website uses cookies to optimize your experience with our

Book Text


Transcript


Exercises

ou{q}=



10.3 Make a table of the results from edge detecting an image with blurring from 1 to 10.

```
in[1]:= Manipulate[EdgeDetect[Blur[], b]], {b, 0, 20}]
```



Stephen Wolfram | An Elementary Introduction to the Wolfram Language

Coordinates and Graphics

7 Colors and Styles

8 Basic Graphics Objects

9 Interactive Manipulation

10 Images

11 Strings and Text

12 Sound

13 Arrays, or Lists of Lists

14 Coordinates and Graphics

15 Scope of the Wolfram Language

16 Real-World Data

17 Units

18 Geocomputation

19 Dates and Times

20 Options

21 Graphs and Networks

22 Machine Learning

23 More about Numbers

24 More Forms of Visualization

25 Ways to Apply Functions

14 | Coordinates and Graphics

AN ELEMENTARY INTRODUCTION TO THE  
Wolfram Language


07:57

Scratch Notebook

Work on your code here

```
#(1) = 12 / 3
```

```
#(76) = ExampleData[{"TestImage", "Handrill"}]
```



Book Text

Transcript

Exercises

Exercises for Section 14 | Coordinates and Graphics

14.1 Make graphics of 5 concentric circles centered at  $\{0, 0\}$   

```
Graphics[Table[Circle[{0, 0}, r], {r, 5}]]
```

14.2 Make 10 concentric circles with random colors.  

```
Graphics[Table[Style[Circle[{0, 0}, r], RandomColor[]], {r, 1, 10}]]
```

14.3 Make graphics of a 10x10 grid of circles with radius 1 cent  

```
Graphics[Table[Circle[{x, y}], {x, 10}, {y, 10}]]
```

14.4 Make a 10x10 grid of points with coordinates at integer po  

```
Graphics[Table[Point[{x, y}], {x, 10}, {y, 10}]]
```

https://www.wolfram.com/wolfram-u/an-elementary-introduction-to-the-wolfram-language/14-coordinates-and-graphics.en.html#tab-3-3

Stephen Wolfram | An Elementary Introduction to the Wolfram Language

Dates and Times

12 Sound

13 Arrays, or Lists of Lists

14 Coordinates and Graphics

15 Scope of the Wolfram Language

16 Real-World Data

17 Units

18 Geocomputation

19 Dates and Times

20 Options

21 Graphs and Networks

22 Machine Learning

23 More about Numbers

24 More Forms of Visualization

25 Ways to Apply Functions

26 Pure Anonymous Functions

27 Applying Functions Repeatedly

28 Tests and Conditionals

29 More about Pure Functions

30 Rearranging Lists

19 | Dates and Times

AN ELEMENTARY INTRODUCTION TO THE  
Wolfram Language


06:52

Scratch Notebook

Work on your code here

In[1] = 12 / 3

In[76] = ExampleData[{"TestImage", "Handrill"}]



Book Text

Transcript

Exercises

Exercises for Section 19 | Dates and Times

19.1 Compute how many days have elapsed since January 1, 1900.

Now = DateObject[{1900, 1, 1}]

19.2 Compute what day of the week January 1, 2000 was.

DayName[DateObject[{2000, 1, 1}]]

19.3 Find the date a hundred thousand days ago.

Today - Quantity[100000, "Days"]

19.4 Find the local time in Delhi.

LocalTime[Entity["City", {"Delhi", "Delhi", ...}]]

19.5 Find the length of daylight today by subtracting today's sunrise from today's sunset.

https://www.wolfram.com/wolfram-u/an-elementary-introduction-to-the-wolfram-language/19-dates-and-times.en.html#tab-3-3

as described in our

Stephen Wolfram | An Elementary Introduction to the Wolfram Language

Ways to Apply Functions

18 Geocomputation

19 Dates and Times

20 Options

21 Graphs and Networks

22 Machine Learning

23 More about Numbers

24 More Forms of Visualization

25 Ways to Apply Functions

26 Pure Anonymous Functions

27 Applying Functions Repeatedly

28 Tests and Conditionals

29 More about Pure Functions

30 Rearranging Lists

31 Parts of Lists

32 Patterns

33 Expressions and Their Structure

34 Associations

35 Natural Language Understanding

36 Creating Websites and Apps

25 | Ways to Apply Functions

AN ELEMENTARY INTRODUCTION TO THE  
Wolfram Language


06:24

Scratch Notebook

Work on your code here

```
in[1] := 12 / 3
```

```
in[2] := ExampleData[{"TestImage", "Handrill"}]
```



Book Text

Transcript

Exercises

Exercises for Section 25 | Ways to Apply Functions

25.1 Use /@ and Range to reproduce the result of Table[f[n], {n, 10}].

```
f /@ Range[5]
```

25.2 Use /@ twice to generate Table[f[g[n]], {n, 10}].

```
f /@ g /@ Range[10]
```

25.3 Use // to create a[b[c[d[x]]]].

```
x // d // c // b // a
```

25.4 Make a list of letters of the alphabet, with a frame around

```
Framed /@ Alphabet[]
```

Ожидание ответа от wolfram-u.wolframcloud.com... website uses cookies to optimize your experience with our service on the site, as described in our



## 30 | Rearranging Lists

# Wolfram Language



08:15

Scratch Notebook

Work on your code here

 $\text{Ans}(11) = 12 / 3$ 

```
in[76]= ExampleData[{"TestImage", "Mandrill"}]
```



This website uses cookies to optimize your experience with our service on the site, as described in our

## Exercises for Section 30 | Rearranging Lists

**30.1** Use Thread to make a list of rules with each letter of the al

Thread[Alphabet[]] → Range[Length[Alphabet[]]

**30.2** Make a 4x6 grid of the first 24 letters of the alphabet.

```
Grid[Partition[Alphabet[], 6]]
```

**30.3** Make a grid of the digits in  $2^{1000}$ , with 50 digits per row,

```
Grid[Partition[IntegerDigits[2^1000], 50],
```

**30.4** Make a grid of the first 400 characters in the Wikipedia arti

```
Grid[Partition[Characters[StringTake[Wikipedia,
  Frame → All]]
```

## 35 | Natural Language Understanding

Wolfram  
Language

05:48

Scratch Notebook

Work on your code here

10/11- 12/3

```
In[78]:= ExampleData[{"TestImage", "Mandrill"}]
```

### Exercises for Section 35 | Natural Language Understanding

**35.1** Use Interpreter to find the location of the Eiffel Tower.

```
Interpreter["Location"]["eiffel tower"]
```

**35.2** Use Interpreter to find a university referred to as “U of T”.

```
Interpreter["University"]["U of T"]
```

**35.3** Use Interpreter to find the chemicals referred to as C<sub>2</sub>H<sub>4</sub>, C<sub>2</sub>H<sub>6</sub>, and C<sub>2</sub>H<sub>2</sub>.

```
Interpreter["Chemical"][{ "C2H4", "C2H6", "C
```

**35.4** Use Interpreter to interpret the date "20140108".



Stephen Wolfram | An Elementary Introduction to the Wolfram Language

Defining Your Own Functions

32 Patterns

33 Expressions and Their Structure

34 Associations

35 Natural Language Understanding

36 Creating Websites and Apps

37 Layout and Display

38 Assigning Names to Things

39 Immediate and Delayed Values

40 Defining Your Own Functions

41 More about Patterns

42 String Patterns and Templates

43 Storing Things

44 Importing and Exporting

45 Datasets

46 Writing Good Code

47 Debugging Your Code

Certification

Track My Progress

40 | Defining Your Own Functions

AN ELEMENTARY INTRODUCTION TO THE  
Wolfram Language


08:33

Scratch Notebook

Work on your code here

```
in[1] = 12 / 3
```

```
in[2] = ExampleData[{"TestImage", "Handrill"}]
```



This website uses cookies to optimize your experience with our service on the site, as described in our

Book Text

Transcript

Exercises

Exercises for Section 40 | Defining Your Own Functions

40.1 Define a function `f` that computes the square of its argument

```
f[x_] := x^2
```

40.2 Define a function `poly` that takes an integer, and makes a plot of a polynomial

```
poly[n_Integer] := Graphics[Style[RegularPolygon[n], Red]]
```

40.3 Define a function `f` that takes a list of two elements and produces a list of two elements

```
f[{a_, b_}] := {b, a}
```

40.4 Create a function `f` that takes two arguments and gives the sum of their squares

Stephen Wolfram | An Elementary Introduction to the Wolfram Language

Datasets

32 Patterns

33 Expressions and Their Structure

34 Associations

35 Natural Language Understanding

36 Creating Websites and Apps

37 Layout and Display

38 Assigning Names to Things

39 Immediate and Delayed Values

40 Defining Your Own Functions

41 More about Patterns

42 String Patterns and Templates

43 Storing Things

44 Importing and Exporting

45 Datasets

46 Writing Good Code

47 Debugging Your Code

Certification

Track My Progress

45 | Datasets

AN ELEMENTARY INTRODUCTION TO THE  
Wolfram Language


09:22

Scratch Notebook

Work on your code here:

12 / 3

```
in[re]- ExampleData[{"TestImage", "Mandrill"}]
```



Book Text

Transcript

Exercises

Exercises for Section 45 | Datasets

Make a word cloud of the planets, with weights determined by their number of moons.

```
WordCloud[Normal[planets[All, "Moons", Length]]]
```

CHECK MY SOLUTION

45.2 Make a bar chart of the number of moons for each planet.

```
BarChart[planets[All, "Moons", Length], ChartLabels -> Automatic]
```

CHECK MY SOLUTION

45.3 Make a dataset of the masses of the planets, sorted by their number of moons.

```
planets[SortBy[Length[#Moons] &], "Mass"]
```

CHECK MY SOLUTION

This website uses cookies to optimize your experience with our service on the site, as described in our