octup.us		Numbers	App.lication
		octup.us	**App**
		Namespace	*Whole set of Data & Logic, Config and App.lets. If App.lication is included in octup.us Names, then this Entity belongs to a greater Collection to provide App Functionality in an octup.us way of life.*
	octup.us | Deck: Generat-o-mat	octup.us | App: Compiler | Chapter: Namespace | Deck: Generat-o-mat	Compiler
	octup.us | D&L: Factorial | o	octup.us | Chapter: Examples | D&L: Factorial | o	
	octup.us | D&L: Factorial | Paper Print Layout	octup.us | Chapter: Examples | D&L: Factorial | Paper Print Layout	
	octup.us | D&L: Factorial | Source Code	octup.us | Chapter: Examples | D&L: Factorial | Source Code	
	octup.us | D&L: Factorial | DEBUG	octup.us | Chapter: Examples | D&L: Factorial | DEBUG	
	octup.us | D&L: Factorial with INDEX()	octup.us | Chapter: Examples | D&L: Factorial with INDEX()	
	octup.us | D&L: Factorial with INDEX() | Source Code	octup.us | Chapter: Examples | D&L: Factorial with INDEX() | Source Code	
	octup.us | D&L: Factorial with INDEX() | DEBUG	octup.us | Chapter: Examples | D&L: Factorial with INDEX() | DEBUG	
	octup.us | D&L: Alternative Factorial | o	octup.us | Chapter: Examples | D&L: Alternative Factorial | o	

Tabelle 1

1.	**Die**	**Vier Jahreszeiten**	**von**	**Antonio Vivaldi.**	
2.	Die	Zauberflöte von Wo	lfga	ng Amadeus Mozart.	
3.	Die	Brandenburgischen	Konz	erte von Johann Sebastian	Bach.
4.	Die	9. Sinfonie von Lu	dwig	van Beethoven.	
5.	Eine	kleine Nachtmusik	von	Wolfgang Amadeus Mozart.	
	Ouverture miniature: Allegro giusto, 2/4 Takt, B-Dur				
	Marche: Tempo di marcia viva, 4/4 Takt, G-Dur				
	Danse de la Fée Dragée (Tanz der Zuckerfee): Andante non troppo, 2/4 Takt, e-moll				
	Danse russe Trepak (Russischer Tanz): Tempo di Trepak, molto vivace, 2/4 Takt, G-Dur				
	Danse arabe (Arabischer Tanz): Allegretto, 3/8 Takt, g-moll				
	Danse chinoise (Chinesischer Tanz): Allegro moderato, 4/4 Takt, B-Dur				
	Danse des mirlitons (Tanz der Rohrflöten): Moderato assai, 2/4 Takt, D-Dur				
	Valse des fleurs (Blumenwalzer): Tempo di Valse, 3/4 Takt, D-Dur				

octup.us | Deck: Generat-o-mat

	octup.us		App	App.let	Chapter	Deck	Config	D&L	Gear	Y	X	Print	Code	Run	DEBUG	K	Revision	View
	Numbers		App.lication	App.let	Sheet	Table	Configuration	Data & Logic	Cell	Row	Column	Paper Print Layout	Source Code	o	DEBUG	Commentary	Version	Presenter
			Whole set of Data & Logic, Config and App.lets. If App.lication is included in octup.us Names, then this Entity belongs to a greater Collection to provide App Functionality in an octup.us way of life.	*Small Collection of Functionality consisting of octup.us Sets, which are usually working independently, have some IO, refer to a couple of octup.us D&L Sets and fit in one octup.us Chapter.*	*In Apple Numbers Runtime several Tables can be grouped in a Sheet. octup.us uses #Chapter for naming instead of Sheet to reflect her independence from spreadsheets. A octup.us Set can be addressed from outside a different Chapter by adding a leading Chapter_Name:: to the Set_Name. From there you can further accessing specific octup.us Gears via XY Indices e.g. F4.*	*In Apple Numbers Runtime several Cells make up a Table. octup.us uses #Deck for naming instead of Table to reflect her independence from spreadsheets.*	*Collection of Variables or Constants, that usually Parametrize or adjust the App.lication. #Configuration can be further categorized by #Config: <Label_of_Deck>*	*Set of Data, Constraints, Functions, computation, semantic or Application Logic. #D&L is short for: Data & Logic*	*In Apple Numbers Runtime you enter Values or Functions into Cells. octup.us uses #Gear for naming instead of Cell to reflect her independence from spreadsheets. A Gear can have a Value (Integer, Float, String, Boolean, ...) or a Function(), which returns a Value.*	*In Apple Numbers Runtime a Table can consist of one or more Rows. octup.us uses #Y for naming Rows to reflect her independence from spreadsheets.*	*In Apple Numbers Runtime a Table can consist of one or more Columns. octup.us uses #X for naming Columns to reflect her independence from spreadsheets.*	*If the octup.us Code has to be printed or viewed by Humans, it can be beneficial to include leading ' and trailing ' to preserve Whitespaces or indicating Whitespaces. The octup.us #Print Tag can be used with octup.us Snaps or octup.us Code.*	*If a Gear oder a Set of Gears contain Functions(), than using the #Code Tag will print the Source Code of the Function() instead of computing the Result.* *Copying octup.us Code from Clipboard should correctly be pasted with functionstext as function() into Apple Numbers Runtime.*	*Copy of one or several Gears, Sets or Chapters Runtime Output as a Snapshot into Clipboard as Tab separated Values.*	*Tag, that refers to an octup.us Set, that presents itself optimized for DEBUG Purposes.*	*Useful && Entertaining Metainformation*	*Re Version Numbers, 4 Integer*	*A Deck, that has no Userinput, hence a #View of Data.*
1	**octup.us | Deck: Generat-o-mat**	octup.us | App: Compiler | Chapter: Namespace | Deck: Generat-o-mat	Compiler		Namespace	Generat-o-mat												
2	**octup.us | Deck: ChatGPT**	octup.us | App: Compiler | Chapter: Hintcheat | Deck: ChatGPT	Compiler		Hintcheat	ChatGPT												
3	**octup.us | D&L: Factorial | o**	octup.us | Chapter: Examples | D&L: Factorial | o			Examples			Factorial						X				
4	**octup.us | D&L: Factorial | Paper Print Layout**	octup.us | Chapter: Examples | D&L: Factorial | Paper Print Layout			Examples			Factorial				X						
5	**octup.us | D&L: Factorial | Source Code**	octup.us | Chapter: Examples | D&L: Factorial | Source Code			Examples			Factorial					X					
6	**octup.us | D&L: Factorial | DEBUG**	octup.us | Chapter: Examples | D&L: Factorial | DEBUG			Examples			Factorial							X			
7	**octup.us | D&L: Factorial with INDEX()**	octup.us | Chapter: Examples | D&L: Factorial with INDEX()			Examples			Factorial with INDEX()										
8	**octup.us | D&L: Factorial with INDEX() | Source Code**	octup.us | Chapter: Examples | D&L: Factorial with INDEX() | Source Code			Examples			Factorial with INDEX()					X					
9	**octup.us | D&L: Factorial with INDEX() | DEBUG**	octup.us | Chapter: Examples | D&L: Factorial with INDEX() | DEBUG			Examples			Factorial with INDEX()							X			
10	**octup.us | D&L: Alternative Factorial | o**	octup.us | Chapter: Examples | D&L: Alternative Factorial | o			Examples			Alternative Factorial						X				
11	**octup.us | D&L: Alternative Factorial | Paper Print Layout**	octup.us | Chapter: Examples | D&L: Alternative Factorial | Paper Print Layout			Examples			Alternative Factorial				X						
12	**octup.us | D&L: Alternative Factorial | Source Code**	octup.us | Chapter: Examples | D&L: Alternative Factorial | Source Code			Examples			Alternative Factorial					X					
13	**octup.us | D&L: Alternative Factorial | DEBUG**	octup.us | Chapter: Examples | D&L: Alternative Factorial | DEBUG			Examples			Alternative Factorial							X			
14	**octup.us | D&L: Conditional Statements | o**	octup.us | Chapter: Examples | D&L: Conditional Statements | o			Examples			Conditional Statements						X				
15	**octup.us | D&L: Conditional Statements | Paper Print Layout**	octup.us | Chapter: Examples | D&L: Conditional Statements | Paper Print Layout			Examples			Conditional Statements				X						
16	**octup.us | D&L: Conditional Statements | Source Code**	octup.us | Chapter: Examples | D&L: Conditional Statements | Source Code			Examples			Conditional Statements					X					
17	**octup.us | D&L: Conditional Statements | DEBUG**	octup.us | Chapter: Examples | D&L: Conditional Statements | DEBUG			Examples			Conditional Statements							X			
18	**octup.us | D&L: Loops & Strings | o**	octup.us | Chapter: Examples | D&L: Loops & Strings | o			Examples			Loops & Strings						X				
19	**octup.us | D&L: Loops & Strings | Paper Print Layout**	octup.us | Chapter: Examples | D&L: Loops & Strings | Paper Print Layout			Examples			Loops & Strings				X						
20	**octup.us | D&L: Loops & Strings | Source Code**	octup.us | Chapter: Examples | D&L: Loops & Strings | Source Code			Examples			Loops & Strings					X					
21	**octup.us | D&L: Loops & Strings | DEBUG**	octup.us | Chapter: Examples | D&L: Loops & Strings | DEBUG			Examples			Loops & Strings							X			
22	**octup.us | D&L: Classes | Source Code**	octup.us | App.let | Chapter: Examples | D&L: Classes | Source Code		X	Examples			Classes					X					
23	**octup.us | D&L: Classes | o**	octup.us | App.let | Chapter: Examples | D&L: Classes | o		X	Examples			Classes						X				
24	**octup.us | D&L: Classes | DEBUG**	octup.us | App.let | Chapter: Examples | D&L: Classes | DEBUG		X	Examples			Classes							X			
25	**octup.us | Source Code**	octup.us | App.let: Classes | Chapter: Examples | Source Code		Classes	Examples								X					
26	**octup.us | o**	octup.us | App.let: Classes | Chapter: Examples | o		Classes	Examples									X				
27	**octup.us | DEBUG**	octup.us | App.let: Classes | Chapter: Examples | DEBUG		Classes	Examples										X			
28	**octup.us | Deck: Hello World & Co. | Paper Print Layout | Source Code**	octup.us | Chapter: Examples | Deck: Hello World & Co. | Paper Print Layout | Source Code			Examples	Hello World & Co.							X	X					
29	**octup.us | Deck: Hello World & Co. | Paper Print Layout | o**	octup.us | Chapter: Examples | Deck: Hello World & Co. | Paper Print Layout | o			Examples	Hello World & Co.							X		X				
30	**octup.us | D&L: Fibronaschen | Source Code**	octup.us | Chapter: Examples | D&L: Fibronaschen | Source Code			Examples			Fibronaschen					X					
31	**octup.us | D&L: Fibronaschen | o**	octup.us | Chapter: Examples | D&L: Fibronaschen | o			Examples			Fibronaschen						X				
32																		
33																		
34	**octup.us | Deck: 6 | Config: 7 | D&L: 8 | Gear: 9 | Y: 10 | X: 11 | Print: 12 | Code: 13 | Run: 14 | DEBUG: 15 | K: 16**	octup.us | App: 3 | App.let: 4 | Chapter: 5 | Deck: 6 | Config: 7 | D&L: 8 | Gear: 9 | Y: 10 | X: 11 | Print: 12 |	3	4	5	6	7	8	9	10	11	12	13	14	15	16	0 0 0 1	
35	**octup.us | Table | Configuration | Data & Logic | Cell | Row | Column | Paper Print Layout | Source Code | o | DEBUG**	octup.us | App.lication | App.let | Sheet | Table | Configuration | Data & Logic | Cell | Row | Column | Pape	X	X	X	X	X	X	X	X	X	X	X	X	X		X		
36																		
37	**octup.us | Deck: Deck Evolver**	octup.us | App: Hearthstone | App.let | Chapter: Tutorials | Deck: Deck Evolver	Hearthstone	X	Tutorials	Deck Evolver												
38	**octup.us | Configuration**	octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | Configuration	Hearthstone	Deck Evolver	Tutorials		X											
39	**octup.us | D&L: Decks**	octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Decks	Hearthstone	Deck Evolver	Tutorials			Decks										
40	**octup.us | D&L: New Cards**	octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: New Cards	Hearthstone	Deck Evolver	Tutorials			New Cards										
41	**octup.us | D&L: Card-o-mat**	octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Card-o-mat	Hearthstone	Deck Evolver	Tutorials			Card-o-mat										
41	**octup.us | D&L: Card-o-mat | Print: Input**	octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Card-o-mat | Print: Input	Hearthstone	Deck Evolver	Tutorials			Card-o-mat				Input						
41	**octup.us | D&L: Card-o-mat | Paper Print Layout**	octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Card-o-mat | Paper Print I	Hearthstone	Deck Evolver	Tutorials			Card-o-mat				X						
41	**octup.us | D&L: Card-o-mat | Paper Print Layout | Source Code**	octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Card-o-mat | Paper Print I	Hearthstone	Deck Evolver	Tutorials			Card-o-mat				X	X					
42																		
43	**octup.us | View: Keyboard**	octup.us | App: Star Citizen | App.let: IO | View: Keyboard	Star Citizen	IO														Keyboard
43	**octup.us | Source Code | View: Keyboard**	octup.us | App: Star Citizen | App.let: IO | View: Keyboard | Source Code	Star Citizen	IO									X						Keyboard
43	**octup.us | View: Keys**	octup.us | App: Star Citizen | App.let: IO | View: Keys	Star Citizen	IO														Keys
	octup.us | D&L: Keyroom	octup.us | App: Star Citizen | App.let: IO | D&L: Keyroom	Star Citizen	IO				Keyroom										
43	**octup.us | Config: KeyFilter**	octup.us | App: Star Citizen | App.let: IO | Config: KeyFilter	Star Citizen	IO			KeyFilter											
43	**octup.us | Config: KeySelector**	octup.us | App: Star Citizen | App.let: IO | Config: KeySelector	Star Citizen	IO			KeySelector											
43	**octup.us | View: Keyboard**	octup.us | App: D4 | App.let: IO | View: Keyboard	D4	IO														Keyboard
43	**octup.us | Source Code | View: Keyboard**	octup.us | App: D4 | App.let: IO | View: Keyboard | Source Code	D4	IO									X						Keyboard
43	**octup.us | View: Keys**	octup.us | App: D4 | App.let: IO | View: Keys	D4	IO														Keys
	octup.us | D&L: Keyroom	octup.us | App: D4 | App.let: IO | D&L: Keyroom	D4	IO				Keyroom										
43	**octup.us | Config: KeyFilter**	octup.us | App: D4 | App.let: IO | Config: KeyFilter	D4	IO			KeyFilter											
43	**octup.us | Config: KeySelector**	octup.us | App: D4 | App.let: IO | Config: KeySelector	D4	IO			KeySelector											

octup.us | Chapter: Examples | Set: Hello World & Co. | Paper Print Layout | Source Code

octup.us | Chap	'B'	'C'	'D'	'E'	'F'	'G'	'H'	'I'	="EOL &ZE
=ZEILE()	' '	'Hello World'	''	''	''	''	''	''	="EOL &ZE
=ZEILE()	' '	''	''	''	''	''	''	''	="EOL &ZE
=ZEILE()	' '	''	''	''	''	''	''	''	="EOL &ZE
=ZEILE()	' '	'1. Declaring and assigning variables:'	''	''	''	''	''	''	="EOL &ZE
=ZEILE()	' '	' '	'X'	'10'	''	''	''	''	="EOL &ZE
=ZEILE()	' '	' '	'Y'	'20'	''	''	''	''	="EOL &ZE
=ZEILE()	' '	''	''	''	''	''	''	''	="EOL &ZE
=ZEILE()	' '	'2. Basic arithmetic operations:'	''	''	''	''	''	''	="EOL &ZE
=ZEILE()	' '	' '	'+'	=E6+E7	''	''	''	''	="EOL &ZE
=ZEILE()	' '	' '	'*'	=E6×E7	''	''	''	''	="EOL &ZE
=ZEILE()	' '	''	''	''	''	''	''	''	="EOL &ZE
=ZEILE()	' '	'4. Looping with a for loop 5 Times:'	''	''	''	''	''	''	="EOL &ZE
=ZEILE()	' '	' '	'1'	=WENN(D14=1;1;WENN(D14=2;2;WENN(D14=3;3;WENN(D14=4;4;WENN(D14=5;5;""))))	''	''	''	''	="EOL &ZE
=ZEILE()	' '	' '	'2'	=WENN(D15=1;1;WENN(D15=2;2;WENN(D15=3;3;WENN(D15=4;4;WENN(D15=5;5;""))))	''	''	''	''	="EOL &ZE
=ZEILE()	' '	' '	'3'	=WENN(D16=1;1;WENN(D16=2;2;WENN(D16=3;3;WENN(D16=4;4;WENN(D16=5;5;""))))	''	''	''	''	="EOL &ZE
=ZEILE()	' '	' '	'4'	=WENN(D17=1;1;WENN(D17=2;2;WENN(D17=3;3;WENN(D17=4;4;WENN(D17=5;5;""))))	''	''	''	''	="EOL &ZE
=ZEILE()	' '	' '	'5'	=WENN(D18=1;1;WENN(D18=2;2;WENN(D18=3;3;WENN(D18=4;4;WENN(D18=5;5;""))))	''	''	''	''	="EOL &ZE
=ZEILE()	' '	' '	'6'	=WENN(D19=1;1;WENN(D19=2;2;WENN(D19=3;3;WENN(D19=4;4;WENN(D19=5;5;""))))	''	''	''	''	="EOL &ZE
=ZEILE()	' '	' '	'7'	=WENN(D20=1;1;WENN(D20=2;2;WENN(D20=3;3;WENN(D20=4;4;WENN(D20=5;5;""))))	''	''	''	''	="EOL &ZE
=ZEILE()	' '	''	'8'	=WENN(D21=1;1;WENN(D21=2;2;WENN(D21=3;3;WENN(D21=4;4;WENN(D21=5;5;""))))	''	''	''	''	="EOL &ZE
=ZEILE()	' '	' '	'9'	=WENN(D22=1;1;WENN(D22=2;2;WENN(D22=3;3;WENN(D22=4;4;WENN(D22=5;5;""))))	''	''	''	''	="EOL &ZE
=ZEILE()	' '	' '	'10'	=WENN(D23=1;1;WENN(D23=2;2;WENN(D23=3;3;WENN(D23=4;4;WENN(D23=5;5;""))))	''	''	''	''	="EOL &ZE
=ZEILE()	' '	'(There are no native Loops in octup.us Code, so nested IF	''	''	''	''	''	''	="EOL &ZE
=ZEILE()	' '	'(Advantage beside guaranteed Runtime and	''	''	''	''	''	''	="EOL &ZE
=ZEILE()	''	''	''	''	''	''	''	''	="EOF

octup.us | Chapter: Examples | Set: Hello World & Co. | Paper Print Layout | Runtime Snapshot

octup	B	C	D	E	F	G	H	I	EOL_1
2		Hello World							EOL_2
3									EOL_3
4									EOL_4
5		1. Declaring and assigning variables:							EOL_5
6			X	10					EOL_6
7			Y	20					EOL_7
8									EOL_8
9		2. Basic arithmetic operations:							EOL_9
10			+	30					EOL_10
11			*	200					EOL_11
12									EOL_12
13		4. Looping with a for loop 5 Times:							EOL_13
14			1	1					EOL_14
15			2	2					EOL_15
16			3	3					EOL_16
17			4	4					EOL_17
18			5	5					EOL_18
19			6						EOL_19
20			7						EOL_20
21			8						EOL_21
22			9						EOL_22
23			10						EOL_23
24		(There are no native Loops in octup.us Code, so nested IF Statements are used to mimic the functionality.							EOL_24
25		(Advantage beside guaranteed Runtime and chances of dealing with the Halting Problem is, that you can							EOL_25
26									EOF

octup.us | Chapter: Examples | D&L: Factorial | Source Code

=ZEILE(A1)	=WENN(ZEILE(B1)=1;1;#REF!×A1)	="The Faculty of "&A1&" as written in "&A1&"! is "& B1
=ZEILE(A2)	=WENN(ZEILE(B2)=1;1;B1×A2)	="The Faculty of "&A2&" as written in "&A2&"! is "& B2
=ZEILE(A3)	=WENN(ZEILE(B3)=1;1;B2×A3)	="The Faculty of "&A3&" as written in "&A3&"! is "& B3
=ZEILE(A4)	=WENN(ZEILE(B4)=1;1;B3×A4)	="The Faculty of "&A4&" as written in "&A4&"! is "& B4
=ZEILE(A5)	=WENN(ZEILE(B5)=1;1;B4×A5)	="The Faculty of "&A5&" as written in "&A5&"! is "& B5

octup.us | Chapter: Examples | D&L: Factorial | Runtime Snapshot

1	1	The Faculty of 1 as written in 1! is 1
2	2	The Faculty of 2 as written in 2! is 2
3	6	The Faculty of 3 as written in 3! is 6
4	24	The Faculty of 4 as written in 4! is 24
5	120	The Faculty of 5 as written in 5! is 120

octup.us | Chapter: Examples | D&L: Factorial | DEBUG

1	1	The Faculty of 1 as written in 1! is 1
2	2	The Faculty of 2 as written in 2! is 2
3	6	The Faculty of 3 as written in 3! is 6
4	24	The Faculty of 4 as written in 4! is 24
5	120	The Faculty of 5 as written in 5! is 120

octup.us | Chapter: Examples | D&L: Factorial with INDEX() | Source Code

=ZEILE()	=WENN(A1=1;1;INDEX(B;A1−1)×A1)	="The Faculty of "&A1&" as written in "&A1&"! is "& B1
=ZEILE()	=WENN(A2=1;1;INDEX(B;A2−1)×A2)	="The Faculty of "&A2&" as written in "&A2&"! is "& B2
=ZEILE()	=WENN(A3=1;1;INDEX(B;A3−1)×A3)	="The Faculty of "&A3&" as written in "&A3&"! is "& B3
=ZEILE()	=WENN(A4=1;1;INDEX(B;A4−1)×A4)	="The Faculty of "&A4&" as written in "&A4&"! is "& B4
=ZEILE()	=WENN(A5=1;1;INDEX(B;A5−1)×A5)	="The Faculty of "&A5&" as written in "&A5&"! is "& B5

octup.us | Chapter: Examples | D&L: Factorial with INDEX() | Runtime Snapshot

1	1	The Faculty of 1 as written in 1! is 1
2	2	The Faculty of 2 as written in 2! is 2
3	6	The Faculty of 3 as written in 3! is 6
4	24	The Faculty of 4 as written in 4! is 24
5	120	The Faculty of 5 as written in 5! is 120

octup.us | Chapter: Examples | D&L: Factorial with INDEX() | DEBUG

1	1	The Faculty of 1 as written in 1! is 1
2	2	The Faculty of 2 as written in 2! is 2
3	6	The Faculty of 3 as written in 3! is 6
4	24	The Faculty of 4 as written in 4! is 24
5	120	The Faculty of 5 as written in 5! is 120

octup.us | Chapter: Examples | D&L: Alternative Factorial | Source Code

=ZEILE(A1)	=A1×C1×D1×E1×F1×	=WENN(A1>1;A1−1;1	=WENN(C1>1;C1−1;1	=WENN(D1>1;D1−1;1	=WENN(E1>1;E1−1;1	=WENN(F1>1;F1−1;1	=WENN(G1>1;G1−1;1	="The Faculty of "&A1
=ZEILE(A2)	=A2×C2×D2×E2×F2×	=WENN(A2>1;A2−1;1	=WENN(C2>1;C2−1;1	=WENN(D2>1;D2−1;1	=WENN(E2>1;E2−1;1	=WENN(F2>1;F2−1;1	=WENN(G2>1;G2−1;1	="The Faculty of "&A2
=ZEILE(A3)	=A3×C3×D3×E3×F3×	=WENN(A3>1;A3−1;1	=WENN(C3>1;C3−1;1	=WENN(D3>1;D3−1;1	=WENN(E3>1;E3−1;1	=WENN(F3>1;F3−1;1	=WENN(G3>1;G3−1;1	="The Faculty of "&A3
=ZEILE(A4)	=A4×C4×D4×E4×F4×	=WENN(A4>1;A4−1;1	=WENN(C4>1;C4−1;1	=WENN(D4>1;D4−1;1	=WENN(E4>1;E4−1;1	=WENN(F4>1;F4−1;1	=WENN(G4>1;G4−1;1	="The Faculty of "&A4
=ZEILE(A5)	=A5×C5×D5×E5×F5×	=WENN(A5>1;A5−1;1	=WENN(C5>1;C5−1;1	=WENN(D5>1;D5−1;1	=WENN(E5>1;E5−1;1	=WENN(F5>1;F5−1;1	=WENN(G5>1;G5−1;1	="The Faculty of "&A5

octup.us | Chapter: Examples | D&L: Alternative Factorial | Runtime Snapshot

1	1	1	1	1	1	1	The Faculty of 1 as written in 1! is 1
2	2	1	1	1	1	1	The Faculty of 2 as written in 2! is 2
3	6	2	1	1	1	1	The Faculty of 3 as written in 6! is 6
4	24	3	2	1	1	1	The Faculty of 4 as written in 24! is 24
5	120	4	3	2	1	1	The Faculty of 5 as written in 120! is 120

octup.us | Chapter: Examples | D&L: Alternative Factorial | DEBUG

1	1	The Faculty of 1 as written in 1! is 1
2	1	The Faculty of 2 as written in 2! is 2
3	1	The Faculty of 3 as written in 6! is 6
4	2	The Faculty of 4 as written in 24! is 24
5	3	The Faculty of 5 as written in 120! is 120

octup.us | Chapter: Examples | D&L: Conditional Statements | Source Code

1	Monday
2	Tuesday
3	Wednesday
4	Thursday
5	Friday
6	Saturday
7	Sunday
8	Invalid day
day	
Consule	=XVERWEIS(B9;A;B;B8)

octup.us | Chapter: Examples | D&L: Conditional Statements | Runtime Snapshot

1	Monday
2	Tuesday
3	Wednesday
4	Thursday
5	Friday
6	Saturday
7	Sunday
8	Invalid day
day	
Consule	**Invalid day**

octup.us | Chapter: Examples | D&L: Conditional Statements | DEBUG

1	Monday
2	Tuesday
3	Wednesday
4	Thursday
5	Friday
6	Saturday
7	Sunday
8	Invalid day
day	
Consule	Invalid day

octup.us | Chapter: Examples | D&L: Loops & Strings | Source Code

1	Apple
2	Banana
3	Cherry
4	Orange
5	Ice
6	
Start	2
End	3
Newline	=" "
	=WENN(UND(B$8≤A1;B$9≥A1);WENN(LÄNGE(B11)>0;B11;"")&B1&B$10;WENN(LÄNGE(B11)>0;B11;""))
	=WENN(UND(B$8≤A2;B$9≥A2);WENN(LÄNGE(B12)>0;B12;"")&B2&B$10;WENN(LÄNGE(B12)>0;B12;""))
	=WENN(UND(B$8≤A3;B$9≥A3);WENN(LÄNGE(B13)>0;B13;"")&B3&B$10;WENN(LÄNGE(B13)>0;B13;""))
	=WENN(UND(B$8≤A4;B$9≥A4);WENN(LÄNGE(B14)>0;B14;"")&B4&B$10;WENN(LÄNGE(B14)>0;B14;""))
	=WENN(UND(B$8≤A5;B$9≥A5);WENN(LÄNGE(B15)>0;B15;"")&B5&B$10;WENN(LÄNGE(B15)>0;B15;""))
	=WENN(UND(B$8≤A6;B$9≥A6);WENN(LÄNGE(B16)>0;B16;"")&B6&B$10;WENN(LÄNGE(B16)>0;B16;""))
	=WENN(UND(B$8≤A7;B$9≥A7);WENN(LÄNGE(B17)>0;B17;"")&B7&B$10;WENN(LÄNGE(B17)>0;B17;""))
Consule	=B17

octup.us | Chapter: Examples | D&L: Loops & Strings | Runtime Snapshot

1	Apple
2	Banana
3	Cherry
4	Orange
5	Ice
6	
Start	2
End	3
Newline	
	Banana
	Banana Cherry
	Banana Cherry
	Banana Cherry
	Banana Cherry
	Banana Cherry
Consule	**Banana Cherry**

octup.us | Chapter: Examples | D&L: Loops & Strings | DEBUG

1	Apple
2	Banana
3	Cherry
4	Orange
5	Ice
6	
Start	2
End	3
Newline	
	Banana
	Banana Cherry
	Banana Cherry
	Banana Cherry
	Banana Cherry
Consule	Banana Cherry

octup.us | App.let | Chapter: Examples | D&L: Classes | Source Code

octup.us | App.let | Cl	Class	Variable or Function L	Object	Object	Object	Object
			1	2	3	4
=ZEILE()	Car	Doors	4	2	4	
=ZEILE()		Brand	Toyota	Volkswagen	Audi	
=ZEILE()		Wheels	4	4	4	
=ZEILE()		Engine	Diesel	Electric	Hybrid Diesel	
=ZEILE()		__n	=WENN(WENNFEHLE	=WENN(WENNFEHLE	=WENN(WENNFEHLER(FINDEN(TEIL(F4;1;1	
=ZEILE()		present()	="I have a"&D7&" "&E	="I have a"&E7&" "&E	="I have a"&F7&" "&F4	
=ZEILE()	Model	Model	Corolla	ID	A3	
=ZEILE()		__n	=WENN(WENNFEHLE	=WENN(WENNFEHLE	=WENN(WENNFEHLER(FINDEN(TEIL(F9;1;1	
=ZEILE()		show()	=D8&", it is a"&D10&"	=E8&", it is a"&E10&"	=F8&", it is a"&F10&" "&F9	
=ZEILE()						

octup.us | App.let | Chapter: Examples | D&L: Classes | Runtime Snapshot

octup	**Class**	**Variable or Function Label**	**Object**	**Object**	**Object**	**Object**
			1	**2**	**3**	**4**
3	Car	Doors	4	2	4	
4		Brand	Toyota	Volkswagen	Audi	
5		Wheels	4	4	4	
6		Engine	Diesel	Electric	Hybrid Diesel	
7		*__n*			n	
8		present()	I have a Toyota	I have a Volkswagen	I have an Audi	
9	Model	Model	Corolla	ID	A3	
10		*__n*		n	n	
11		show()	I have a Toyota, it is a Corolla	I have a Volkswagen, it is an ID	I have an Audi, it is an A3	
12						

octup.us | App.let | Chapter: Examples | D&L: Classes | DEBUG

octup.us	Class	Variable or Function Label	Object	Object	Object	Object
			1	2	3	4
3	Car	Doors	4	2	4	
4		Brand	Toyota	Volkswagen	Audi	
5		Wheels	4	4	4	
6		Engine	Diesel	Electric	Hybrid Diesel	
7		__n			n	
8		present()	I have a Toyota	I have a Volkswagen	I have an Audi	
9	Model	Model	Corolla	ID	A3	
10		__n		n	n	
11		show()	I have a Toyota, it is a Corolla	I have a Volkswagen, it is an ID	I have an Audi, it is an A3	
12						

```
class Model extends Car {
  constructor(brand, mod) {
    super(brand);
    this.model = mod;
  }
  show() {
    return this.present() + ', it is a ' + this.model;
  }
}
let myCar = new Model("Toyota", "Corolla");
console.log(myCar.show()); // Outputs: "I have a Toyota, it is a Corolla"

class Car {
  constructor(brand) {
    this.carname = brand;
  }
  present() {
    return 'I have a ' + this.carname;
  }
}
```

octup.us | App.let: Classes | Chapter: Examples | Source Code

octup.us | App.let:	User Interface Label	IO
=ZEILE()	Console out Object Selector	3
=ZEILE()		
=ZEILE()		
=ZEILE()		
Consule	Output	=XVERWEIS(C3;'octup.us | App.let | Chapter: Examples | D&L: Classes | Runtime Snapshot'::$2:$2;'octup.us | App.let

octup.us | App.let: Classes | Chapter: Examples | Runtime Snapshot

octup.us | App.let: Classes | C	User Interface Label	IO
3	Console out Object Selector	3
4		
5		
6		
Consule	**Output**	**I have an Audi, it is an A3**

octup.us | Chapter: Examples | D&L: Fibronaschen | Source Code

octup.us | Chapter:	Examples | D&L: Fibronaschen	fibonacci
0	1	0, 1
=B2+A2	=A3+B2	=C2&", "&A3&", "&B3
=B3+A3	=A4+B3	=C3&", "&A4&", "&B4
=B4+A4	=A5+B4	=C4&", "&A5&", "&B5
=B5+A5	=A6+B5	=C5&", "&A6&", "&B6
=B6+A6	=A7+B6	=C6&", "&A7&", "&B7
=B7+A7	=A8+B7	=C7&", "&A8&", "&B8
=B8+A8	=A9+B8	=C8&", "&A9&", "&B9
=B9+A9	=A10+B9	=C9&", "&A10&", "&B10

octup.us | Chapter: Examples | D&L: Fibronaschen | Runtime Snapshot

octup.us | Chapter: Examples	D&L: Fibronaschen | Runtim	fibonacci
0	1	0, 1
1	2	0, 1, 1, 2
3	5	0, 1, 1, 2, 3, 5
8	13	0, 1, 1, 2, 3, 5, 8, 13
21	34	0, 1, 1, 2, 3, 5, 8, 13, 21, 34
55	89	0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89
144	233	0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233
377	610	0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610
987	1597	0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597

The fibonacci function uses an iterative approach to generate Fibonacci numbers up to a given n value. Here's a brief description of the algorithm:

If n is less than or equal to 0, the function returns an empty array ([]), as there are no Fibonacci numbers to generate.
If n is equal to 1, the function returns an array containing only the first Fibonacci number, which is 0.
If n is equal to 2, the function returns an array containing the first two Fibonacci numbers, [0, 1].
For n greater than 2, the function initializes an array fibArray with the first two Fibonacci numbers, [0, 1].
It then enters a loop starting from i = 2 and continues until i reaches n. In each iteration, it calculates the next Fibonacci number by adding the previous two Fibonacci numbers: fibArray[i - 1] + fibArray[i - 2].
The calculated Fibonacci number is then appended to the fibArray using the push method.
After the loop ends, the function returns the fibArray containing the generated Fibonacci numbers up to n.
By following this algorithm, the fibonacci function efficiently generates the Fibonacci sequence iteratively for the given input n.

Input: fibonacci(15)
Output: [0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377]

Explanation: The input is 15, so the function generates the first 15 Fibonacci numbers and returns them as an array: [0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377].

```
<!DOCTYPE html>
<html>
<head>
    <style>
        body {
            font-family: Arial, sans-serif;
        }
        h1 {
            color: #333;
        }
    </style>
</head>
<body>
    <h1>Fibonacci Series</h1>
    <p id="fibSeries"></p>

    <script>
        var fibArray = [0, 1];
        for (var i = 2; i < 10; i++) {
            fibArray.push(fibArray[i - 1] + fibArray[i - 2]);
        }
        document.getElementById("fibSeries").innerText = fibArray.join(", ");
    </script>
</body>
</html>
```

octup.us | App: Hearthstone | App.let | Chapter: Tutorials | Set: Deck Evolver

			0	5,5	1	2	3	4	5	6	7	8	9	10
2		**Biology Project**		**1,5**	2	1								
2		**Druid of the Reef**		**2,0**	2	2								
2		**Jade Idol**		**1,0**	1	1								
2		**Lost in the Park**		**1,0**	1	1								
1		**Naturalize**		**2,0**	2									
1		**Peaceful Piper**		**2,0**	2									
1		**Raven Idol**		**1,0**	1									
2		**Untapped Potential**		**1,0**	1	1								
1		**Worthy Expedition**		**1,0**	1									
2		**Astalor Bloodsworn**		**1,0**	1	1								
2		**Capture Coldtooth Mine**		**2,0**	2	2								
1		**Druid of the Saber**		**2,0**	2									
2		**Grizzled Wizard**		**1,0**	1	1								
1		**Jerry Rig Carpenter**		**1,0**	1									
2		**Wrath**		**2,0**	2	2								
2		**Bad Luck Albatross**		**1,0**	1	1								
1		**Deathlord**		**1,0**	1									
2		**Bad Luck Albatross**		**1,0**	1	1								
1		**Deathlord**		**1,0**	1									
2		**Feral Rage**		**1,5**	1	2								
1		**Archmage Vargoth**		**1,0**	1									
2		**Rhythm and Roots**		**1,0**	1	1								
2		**Nourish**		**1,0**	1	1								
1		**Spreading Plague**		**1,0**	1									
2		**Malfurion the Pestilent**		**1,0**	1	1								
1		**Wildheart Guff**		**1,0**		1								
1		**Secure the Deck**		**2,0**		2								
		Sir Finley Mrrgglton												
1		**Kobold Stickyfinger**		**1,0**		1								
1		**Photographer Fizzle**		**1,0**		1								
		Savage Roar												
1		**Poison Seeds**		**2,0**		2								
		Living Roots												
1		**Widowbloom Seedsman**		**2,0**		2								
1		**Wardruid Loti**		**1,0**		1								
1		**Flobbidinous Floop**		**1,0**		1								
1		**Tyrantus**		**1,0**		1								

octup.us | Chapter: Examples | Set: Hello World & Co. | Paper Print Layout | Source Code

octup.us | Chap	'B'	'C'	'D'	'E'	'F'	'G'	'H'	'I'	="EOL _"&ZE
=ZEILE()	' '	'Hello World'	''	''	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	''	''	''	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	''	''	''	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	'1. Declaring and assigning variables:'	''	''	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	' '	'X'	'10'	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	' '	'Y'	'20'	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	''	''	''	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	'2. Basic arithmetic operations:'	''	''	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	' '	'+'	=E6+E7	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	' '	'*'	=E6×E7	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	''	''	''	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	'4. Looping with a for loop 5 Times:'	''	''	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	' '	'1'	=WENN(D14=1;1;WENN(D14=2;2;WENN(D14=3;3;WENN(D14=4;4;WENN(D14=5;5;"")))))	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	' '	'2'	=WENN(D15=1;1;WENN(D15=2;2;WENN(D15=3;3;WENN(D15=4;4;WENN(D15=5;5;"")))))	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	' '	'3'	=WENN(D16=1;1;WENN(D16=2;2;WENN(D16=3;3;WENN(D16=4;4;WENN(D16=5;5;"")))))	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	' '	'4'	=WENN(D17=1;1;WENN(D17=2;2;WENN(D17=3;3;WENN(D17=4;4;WENN(D17=5;5;"")))))	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	' '	'5'	=WENN(D18=1;1;WENN(D18=2;2;WENN(D18=3;3;WENN(D18=4;4;WENN(D18=5;5;"")))))	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	' '	'6'	=WENN(D19=1;1;WENN(D19=2;2;WENN(D19=3;3;WENN(D19=4;4;WENN(D19=5;5;"")))))	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	' '	'7'	=WENN(D20=1;1;WENN(D20=2;2;WENN(D20=3;3;WENN(D20=4;4;WENN(D20=5;5;"")))))	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	' '	'8'	=WENN(D21=1;1;WENN(D21=2;2;WENN(D21=3;3;WENN(D21=4;4;WENN(D21=5;5;"")))))	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	' '	'9'	=WENN(D22=1;1;WENN(D22=2;2;WENN(D22=3;3;WENN(D22=4;4;WENN(D22=5;5;"")))))	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	' '	'10'	=WENN(D23=1;1;WENN(D23=2;2;WENN(D23=3;3;WENN(D23=4;4;WENN(D23=5;5;"")))))	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	'(There are no native Loops in octup.us Code, so nested IF	''	''	''	''	''	''	="EOL _"&ZE
=ZEILE()	' '	'(Advantage beside guaranteed Runtime and	''	''	''	''	''	''	="EOL _"&ZE
=ZEILE()	''	''	''	''	''	''	''	''	="EOF

octup.us | Chapter: Examples | Set: Hello World & Co. | Paper Print Layout | Runtime Snapshot

octup	B	C	D	E	F	G	H	I	EOL_1
2		Hello World							EOL_2
3									EOL_3
4									EOL_4
5		1. Declaring and assigning variables:							EOL_5
6			X	10					EOL_6
7			Y	20					EOL_7
8									EOL_8
9		2. Basic arithmetic operations:							EOL_9
10			+	30					EOL_10
11			*	200					EOL_11
12									EOL_12
13		4. Looping with a for loop 5 Times:							EOL_13
14			1	1					EOL_14
15			2	2					EOL_15
16			3	3					EOL_16
17			4	4					EOL_17
18			5	5					EOL_18
19			6						EOL_19
20			7						EOL_20
21			8						EOL_21
22			9						EOL_22
23			10						EOL_23
24		(There are no native Loops in octup.us Code, so nested IF Statements are used to mimic the functionality.							EOL_24
25		(Advantage beside guaranteed Runtime and chances of dealing with the Halting Problem is, that you can a							EOL_25
26									EOF

octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Decks

0	1	2	3	4	5	6	7	8	9
			0,7						
04.05.23 14:30	05.05.23 12:08	04.05.23 15:27	04.05.23 16:01	4.5.2023 17:06:49	04.05.23 18:13	05.05.23 12:08	04.06.23 19:57	04.06.23 19:57	04.06.23 19:57
### Octerion # Class: Druid # Format: Wild # # 2x (1) Biology Project # 2x (1) Druid of the Reef # 1x (1) Jade Idol # 1x (1) Lost in the Park # 2x (1) Naturalize # 2x (1) Peaceful Piper # 1x (1) Raven Idol # 1x (1) Untapped Potential # 1x (1) Worthy Expedition # 1x (2) Astalor Bloodsworn # 2x (2) Capture Coldtooth Mine # 2x (2) Druid of the Saber # 1x (2) Grizzled Wizard # 1x (2) Jerry Rig Carpenter # 2x (2) Wrath # 1x (3) Bad Luck Albatross # 1x (3) Deathlord # 1x (3) Feral Rage # 1x (4) Archmage Vargoth # 1x (4) Rhythm and Roots # 1x (5) Nourish # 1x (6) Spreading Plague # 1x (7) Malfurion the Pestilent # AAEBAZarBBD+DZdotLsCoM0CmdMC1pkD+KED3KID/bADp7UDo/YDsIAE2p8EuKAE4qQFq+AFB+kB3xWP9gKsgASwpQSA1AT83wUAAA== # # To use this deck, copy it to your clipboard and create a new deck in Hearthstone	### Octerion # Class: Druid # Format: Wild # # 1x (1) Biology Project # 2x (1) Druid of the Reef # 1x (1) Jade Idol # 1x (1) Lost in the Park # 2x (1) Secure the Deck # 1x (1) Untapped Potential # 1x (2) Astalor Bloodsworn # 2x (2) Capture Coldtooth Mine # 1x (2) Grizzled Wizard # 2x (2) Wrath # 1x (3) Bad Luck Albatross # 2x (3) Feral Rage # 1x (3) Photographer Fizzle # 1x (3) Wardruid Loti # 1x (4) Flobbidinous Floop # 2x (4) Poison Seeds # 1x (4) Rhythm and Roots # 2x (4) Widowbloom Seedsman # 1x (5) Kobold Stickyfinger # 1x (5) Nourish # 1x (5) Wildheart Guff # 1x (7) Malfurion the Pestilent # 1x (10) Tyrantus # AAEBAZarBBC0uwLixwKu0gKZ0wKP9gL1/AL4oQP9sAOIsQOntQOj9gOJiwTanwTipAWs0QWr4AUHig75rQOsgAS4oASwpQSA1ATB3wQAAA== # # To use this deck, copy it to your clipboard and create a new deck in Hearthstone								# 1x (0) Jo
23	23	0	0	0	0	0	0	0	1

octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | Configuration

REGEX	Cards	(?<=# \dx \(\d|\))[a-zA-Z|7 :',.-]+			
REGEX	Quantity of a card	(?=\dx \(\d+\)	)\d		

octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Decks-1

04.05.23 15:27	04.05.23 16:01	4.5.2023 17:06:49	04.05.23 18:13	05.05.23 12:08
### Octerion # Class: Druid # Format: Wild # # 2x (1) Biology Project # 2x (1) Druid of the Reef # 1x (1) Jade Idol # 1x (1) Lost in the Park # 1x (1) Naturalize # 1x (1) Raven Idol # 2x (1) Secure the Deck # 1x (1) Sir Finley Mrrgglton # 1x (1) Untapped Potential # 1x (1) Worthy Expedition # 1x (2) Astalor Bloodsworn # 2x (2) Capture Coldtooth Mine # 1x (2) Druid of the Saber # 1x (2) Grizzled Wizard # 1x (2) Jerry Rig Carpenter # 2x (2) Wrath # 1x (3) Bad Luck Albatross # 1x (3) Deathlord # 2x (3) Feral Rage # 1x (4) Archmage Vargoth # 1x (4) Rhythm and Roots # 1x (5) Kobold Stickyfinger # 1x (5) Wildheart Guff # 1x (7) Malfurion the Pestilent # AAEBAZarBBLpAf4N3xWEF5dotLsCmdMC1pkD+KED3KID/bADiLEDp7UDo/YDsIAEiYsE4qQFq+AFBo/2AvmtA6yABLigBLClBIDUBAAA # # To use this deck, copy it to your clipboard and create a new deck in Hearthstone	### Octerion # Class: Druid # Format: Wild # # 2x (1) Biology Project # 2x (1) Druid of the Reef # 1x (1) Jade Idol # 1x (1) Lost in the Park # 2x (1) Secure the Deck # 1x (1) Sir Finley Mrrgglton # 1x (1) Untapped Potential # 1x (2) Astalor Bloodsworn # 2x (2) Capture Coldtooth Mine # 1x (2) Druid of the Saber # 1x (2) Grizzled Wizard # 1x (2) Jerry Rig Carpenter # 2x (2) Wrath # 1x (3) Bad Luck Albatross # 1x (3) Deathlord # 2x (3) Feral Rage # 1x (3) Photographer Fizzle # 1x (3) Savage Roar # 2x (4) Poison Seeds # 1x (4) Rhythm and Roots # 1x (5) Kobold Stickyfinger # 1x (5) Wildheart Guff # 1x (7) Malfurion the Pestilent # AAEBAZarBBDmBf4N3xWEF7S7ApnTAvihA/2wA4ixA6e1A6P2A7CABImLBOKkBazRBavgBQeKDo/2AvmtA6yABLigBLClBIDUBAAA # # To use this deck, copy it to your clipboard and create a new deck in Hearthstone	### Octerion # Class: Druid # Format: Wild # # 2x (1) Biology Project # 2x (1) Druid of the Reef # 1x (1) Jade Idol # 2x (1) Living Roots # 1x (1) Lost in the Park # 2x (1) Secure the Deck # 1x (1) Sir Finley Mrrgglton # 1x (1) Untapped Potential # 1x (2) Astalor Bloodsworn # 2x (2) Capture Coldtooth Mine # 1x (2) Druid of the Saber # 1x (2) Grizzled Wizard # 2x (2) Wrath # 1x (3) Bad Luck Albatross # 1x (3) Deathlord # 2x (3) Feral Rage # 1x (3) Photographer Fizzle # 2x (4) Poison Seeds # 1x (4) Rhythm and Roots # 1x (5) Kobold Stickyfinger # 1x (5) Wildheart Guff # 1x (7) Malfurion the Pestilent # AAEBAZarBA7+Dd8VhBe0uwKZ0wL4oQP9sAOIsQOntQOj9gOJiwTipAWs0QWr4AUIig6P9gL5rQOsgAS4oASwpQSA1ASB1AQAAA== # # To use this deck, copy it to your clipboard and create a new deck in Hearthstone	### Octerion # Class: Druid # Format: Wild # # 2x (1) Biology Project # 2x (1) Druid of the Reef # 1x (1) Jade Idol # 1x (1) Lost in the Park # 2x (1) Secure the Deck # 1x (1) Sir Finley Mrrgglton # 1x (1) Untapped Potential # 1x (2) Astalor Bloodsworn # 2x (2) Capture Coldtooth Mine # 1x (2) Druid of the Saber # 2x (2) Grizzled Wizard # 2x (2) Wrath # 1x (3) Bad Luck Albatross # 2x (3) Feral Rage # 1x (3) Photographer Fizzle # 2x (4) Poison Seeds # 1x (4) Rhythm and Roots # 1x (4) Widowbloom Seedsman # 1x (5) Kobold Stickyfinger # 1x (5) Nourish # 1x (5) Wildheart Guff # 1x (7) Malfurion the Pestilent # AAEBAZarBA7fFYQXtLsCmdMC+KED/bADiLEDo/YDiYsE2p8Ewd8E4qQFrNEFq+AFCIoOj/YC+a0Dp7UDrIAEuKAEsKUEgNQEAAA= # # To use this deck, copy it to your clipboard and create a new deck in Hearthstone	### Octerion # Class: Druid # Format: Wild # # 1x (1) Biology Project # 2x (1) Druid of the Reef # 1x (1) Jade Idol # 1x (1) Lost in the Park # 2x (1) Secure the Deck # 1x (1) Untapped Potential # 1x (2) Astalor Bloodsworn # 2x (2) Capture Coldtooth Mine # 1x (2) Grizzled Wizard # 2x (2) Wrath # 1x (3) Bad Luck Albatross # 2x (3) Feral Rage # 1x (3) Photographer Fizzle # 1x (3) Wardruid Loti # 1x (4) Flobbidinous Floop # 2x (4) Poison Seeds # 1x (4) Rhythm and Roots # 2x (4) Widowbloom Seedsman # 1x (5) Kobold Stickyfinger # 1x (5) Nourish # 1x (5) Wildheart Guff # 1x (7) Malfurion the Pestilent # 1x (10) Tyrantus # AAEBAZarBBC0uwLixwKu0gKZ0wKP9gL1/AL4oQP9sAOIsQOntQOj9gOJiwTanwTipAWs0QWr4AUHig75rQOsgAS4oASwpQSA1ATB3wQAAA== # # To use this deck, copy it to your clipboard and create a new deck in Hearthstone

octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Card-o-mat

1			1		Biology Project		Biology Project	
2			2		Druid of the Reef		Druid of the Reef	
3			3		Jade Idol		Jade Idol	
4			4		Lost in the Park		Lost in the Park	
5			5		Naturalize		Naturalize	
6			6		Peaceful Piper		Peaceful Piper	
7			7		Raven Idol		Raven Idol	
8			8		Untapped Potential		Untapped Potential	
9			9		Worthy Expedition		Worthy Expedition	
10			10		Astalor Bloodsworn		Astalor Bloodsworn	
11			11		Capture Coldtooth Mine		Capture Coldtooth Mine	
12			12		Druid of the Saber		Druid of the Saber	
13			13		Grizzled Wizard		Grizzled Wizard	
14			14		Jerry Rig Carpenter		Jerry Rig Carpenter	
15			15		Wrath		Wrath	
16			16		Bad Luck Albatross		Bad Luck Albatross	
17			17		Deathlord		Deathlord	
18			18		Feral Rage		Feral Rage	
19			19		Archmage Vargoth		Archmage Vargoth	
20			20		Rhythm and Roots		Rhythm and Roots	
21			21		Nourish		Nourish	
22			22		Spreading Plague		Spreading Plague	
23			23		Malfurion the Pestilent		Malfurion the Pestilent	
24	23	1			Biology Project		Secure the Deck	
25					Druid of the Reef		Photographer Fizzle	
26					Jade Idol		Wardruid Loti	
27					Lost in the Park		Flobbidinous Floop	
28			24		Secure the Deck		Poison Seeds	
29					Untapped Potential		Widowbloom Seedsman	
30					Astalor Bloodsworn		Kobold Stickyfinger	
31					Capture Coldtooth Mine		Wildheart Guff	
32					Grizzled Wizard		Tyrantus	
33					Wrath			
34					Bad Luck Albatross			
35					Feral Rage			
36			25		Photographer Fizzle			
37			26		Wardruid Loti			
38			27		Flobbidinous Floop			
39			28		Poison Seeds			
40					Rhythm and Roots		Jo	1
41			29		Widowbloom Seedsman			
42			30		Kobold Stickyfinger			
43					Nourish			
44			31		Wildheart Guff			
45					Malfurion the Pestilent			
46			32		Tyrantus			
47	23	2	33	1				
48		3	34	2				
49		4	35	3				
50		5	36	4				
51		6	37	5				
52		7	38	6				
53		8	39	7				
54		9	40		Jo			
55	1	10	41	8				
56			42	9				
57			43	10				
58			44	11				
59			45	12				
60			46	13				
61			47	14				
62			48	15				
63			49	16				
64			50	17				
65			51	18				
66			52	19				
67			53	20				
68			54	21				
69			55	22				
70			56	23				
71			57	24				
72			58	25				
73			59	26				
74			60	27				
75			61	28				
76			62	29				
77			63	30				
78			64	31				
79			65	32				
80			66	33				
81			67	34				
82			68	35				
83			69	36				
84			70	37				
85			71	38				
86			72	39				
87			73	40				
88			74	41				
89			75	42				
90			76	43				
91			77	44				
92			78	45				
93			79	46				
94			80	47				
95			81	48				
96			82	49				
97			83	50				
98			84	51				
99			85	52				
100			86	53				
101			87	54				
102			88	55				
103			89	56				
104			90	57				
105			91	58				
106			92	59				
107			93	60				
108			94	61				
109			95	62				
110			96	63				
111			97	64				
112			98	65				
113			99	66				
114			100	67				
115			101	68				
116			102	69				
117			103	70				
118			104	71				
119			105	72				
120			106	73				
121			107	74				
122			108	75				
123			109	76				
124			110	77				
125			111	78				
126			112	79				
127			113	80				
128			114	81				
129			115	82				
130			116	83				
131			117	84				
132			118	85				
133			119	86				
134			120	87				
135			121	88				
136			122	89				
137			123	90				
138			124	91				
139			125	92				
140			126	93				
141			127	94				
142			128	95				
143			129	96				
144			130	97				
145			131	98				
146			132	99				
147			133	100				
148			134	101				
149			135	102				
150			136	103				
151			137	104				
152			138	105				
153			139	106				
154			140	107				
155			141	108				
156			142	109				
157			143	110				
158			144	111				
159			145	112				
160			146	113				
161			147	114				
162			148	115				
163			149	116				
164			150	117				
165			151	118				
166			152	119				
167			153	120				
168			154	121				
169			155	122				
170			156	123				
171			157	124				
172			158	125				
173			159	126				
174			160	127				
175			161	128				
176			162	129				
177			163	130				
178			164	131				
179			165	132				
180			166	133				
181			167	134				
182			168	135				
183			169	136				
184			170	137				
185			171	138				
186			172	139				
187			173	140				
188			174	141				
189			175	142				
190			176	143				
191			177	144				
192			178	145				
193			179	146				
194			180	147				
195			181	148				
196			182	149				
197			183	150				
198			184	151				
199			185	152				
200			186	153				
201			187	154				
202			188	155				
203			189	156				
204			190	157				
205			191	158				
206			192	159				
207			193	160				
208			194	161				
209			195	162				
210			196	163				
211			197	164				
212			198	165				
213			199	166				
214			200	167				
215			201	168				
216			202	169				
217			203	170				
218			204	171				
219			205	172				
220			206	173				
221			207	174				
222			208	175				
223			209	176				
224			210	177				
225			211	178				
226			212	179				
227			213	180				
228			214	181				
229			215	182				
230			216	183				
231			217	184				
232			218	185				
233			219	186				
234			220	187				
235			221	188				
236			222	189				
237			223	190				
238			224	191				
239			225	192				
240			226	193				
241			227	194				
242			228	195				
243			229	196				
244			230	197				
245			231	198				
246			232	199				
247			233	200				
248			234	201				
249			235	202				
250			236	203				
251			237	204				
252			238	205				
253			239	206				
254			240	207				
255			241	208				
256			242	209				
257			243	210				
258			244	211				
259			245	212				
260			246	213				
261			247	214				
262			248	215				
263			249	216				
264			250	217				
265			251	218				
266			252	219				
267			253	220				
268			254	221				
269			255	222				
270			256	223				
271			257	224				
272			258	225				
273			259	226				
274			260	227				
275			261	228				
276			262	229				
277			263	230				
278			264	231				
279			265	232				
280			266	233				
281			267	234				
282			268	235				
283			269	236				
284			270	237				
285			271	238				
286			272	239				
287			273	240				
288			274	241				
289			275	242				
290			276	243				
291			277	244				
292			278	245				
293			279	246				
294			280	247				
295			281	248				
296			282	249				
297			283	250				
298			284	251				
299			285	252				
300			286	253				

octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: New Cards

Lost in the Park
Naturalize
Grizzled Wizard
Jerry Rig Carpenter
Untapped Potential
Astalor Bloodsworn
Capture Coldtooth Mine
Druid of the Saber
Wrath
jo
-
-
-
-
-
-
-
-
-
-

octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | Configuration

REGEX	Cards	(?<=# \dx \(\d\))[a-zA-Z|7 :',.-]+			
REGEX	Quantity of a card	(?=\dx \(\d+\))	)\d		

octup.us | App: Hearthstone | App.let | Chapter:

Lost in the Park
Naturalize
Peaceful Piper
Raven Idol
Untapped Potential
Worthy Expedition
Astalor Bloodsworn
Capture Coldtooth Mine
Druid of the Saber
Grizzled Wizard
Jerry Rig Carpenter
Wrath

octup.us | App: Hearthstone | App.let | Chapter: Tutorials | Set: Deck Evolver

		octup.us | App: Hearthstone | App.let | Chapter: Tuto	5,5	1	2	3	4	5	6	7	8	9	10
7		**Biology Project**	**1,9**	2	2	2	2	2	2	1			
7		**Druid of the Reef**	**2,0**	2	2	2	2	2	2	2			
6		**Jade Idol**	**1,0**		1	1	1	1	1	1			
4		**Bad Luck Albatross**	**1,0**				1	1	1	1			
2		**Deathlord**	**1,0**				1	1					
4		**Bad Luck Albatross**	**1,0**				1	1	1	1			
2		**Deathlord**	**1,0**				1	1					
4		**Feral Rage**	**2,0**				2	2	2	2			
		Archmage Vargoth											
4		**Rhythm and Roots**	**1,0**				1	1	1	1			
2		**Nourish**	**1,0**						1	1			
1		**Spreading Plague**	**1,0**	1									
6		**Malfurion the Pestilent**	**1,0**	1	1		1	1	1	1			
4		**Wildheart Guff**	**1,0**				1	1	1	1			
5		**Secure the Deck**	**2,0**			2	2	2	2	2			
3		**Sir Finley Mrrgglton**	**1,0**				1	1	1				
4		**Kobold Stickyfinger**	**1,0**				1	1	1	1			
4		**Photographer Fizzle**	**1,0**				1	1	1	1			
1		**Savage Roar**	**1,0**				1						
4		**Poison Seeds**	**2,0**				2	2	2	2			
1		**Living Roots**	**2,0**					2					
2		**Widowbloom Seedsman**	**1,5**						1	2			
1		**Wardruid Loti**	**1,0**							1			
1		**Flobbidinous Floop**	**1,0**							1			
1		**Tyrantus**	**1,0**							1			

octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Decks

0	1	2	3	4	5	6	7	8	9
			0,7						
04.05.23 14:30	04.05.23 14:31	04.05.23 15:27	04.05.23 16:01	4.5.2023 17:06:49	04.05.23 18:13	05.05.23 12:08	=jetzt()	=jetzt()	=jetzt()
### Octerion # Class: Druid # Format: Wild # # 2x (1) Biology Project # 2x (1) Druid of the Reef # 1x (5) … # 1x (6) Spreading Plague # 1x (7) Malfurion the Pestilent # AAEBAZarBBD+DZdotLsCoM0CmdMC1pkD+KED3KID/bADp7UDo/YDslAE2p8EuKAE4qQFq+AFB+kB3xWP9gKsgASwpQSA1AT83wUAAA== # # To use this deck, copy it to your clipboard and create a new deck in Hearthstone	### Octerion # Class: Druid # Format: Wild # # 2x (1) Biology Project # 2x (1) Druid of the Reef # 1x (1) Jade Idol # 1x (1) … # 1x (7) Malfurion the Pestilent # AAEBAZarBBL+Dd8Vl2i0uwKgzQKZ0wLWmQP4oQPcogP9sAOntQOj9gOwgASJiwTanwS4oATipAWr4AUG6QGP9gKsgASwpQSA1AT83wUAAA== # # To use this deck, copy it to your clipboard and create a new deck in Hearthstone	### Octerion # Class: Druid # Format: Wild # # 2x (1) Biology Project # 2x (1) Druid of the Reef # 1x (1) Jade Idol # 1x (1) Lost in the Park # 1x (1) Naturalize # 1x (1) … # 2x (1) Secure the Deck # 1x (2) Grizzled Wizard # 1x (2) Jerry Rig Carpenter # AAEBAZarBBLpAf4N3xWEF5dotLsCmdMC1pkD+KED3KID/bADiLEDp7UDo/YDslAEiYsE4qQFq+AFBo/2AvmtA6yABLigBLCIBIDUBAAA # # To use this deck, copy it to your clipboard and create a new deck in Hearthstone	### Octerion # Class: Druid # Format: Wild # # 2x (1) Biology Project # 2x (1) Druid of the Reef # 1x (1) Jade Idol # 1x (1) Lost in the Park # 2x (1) Secure the Deck # 1x (1) Sir Finley Mrrgglton # 1x (1) Untapped Potential # 1x (2) Astalor Bloodsworn # 2x (2) Capture Coldtooth Mine # 1x (2) Druid of the Saber # 1x (2) Grizzled Wizard # 1x (2) Jerry Rig Carpenter # 2x (2) Wrath # 1x (3) Bad Luck Albatross # 1x (3) Deathlord # 2x (3) Feral Rage # 1x (3) Photographer Fizzle # 1x (3) Savage Roar # 2x (4) Poison Seeds # 1x (4) Rhythm and Roots # 1x (5) Kobold Stickyfinger # 1x (5) Wildheart Guff # 1x (7) Malfurion the Pestilent # AAEBAZarBBDmBf4N3xWEF7S7ApnTAvihA/2wA4ixA6e1A6P2A7CABImLBOKkBazRBavgBQeKDo/2AvmtA6yABLigBLCIBIDUBAAA # # To use this deck, copy it to your clipboard and create a new deck in Hearthstone	### Octerion # Class: Druid # Format: Wild # # 2x (1) Biology Project # 2x (1) Druid of the Reef # 1x (1) Jade Idol # 2x (1) Living Roots # 1x (1) Lost in the Park # 2x (1) Secure the Deck # 1x (1) Sir Finley Mrrgglton # 1x (1) Untapped Potential # 1x (2) Astalor Bloodsworn # 2x (2) Capture Coldtooth Mine # 1x (2) Druid of the Saber # 1x (2) Grizzled Wizard # 2x (2) Wrath # 1x (3) Bad Luck Albatross # 1x (3) Deathlord # 2x (3) Feral Rage # 1x (3) Photographer Fizzle # 2x (4) Poison Seeds # 1x (4) Rhythm and Roots # 1x (5) Kobold Stickyfinger # 1x (5) Wildheart Guff # 1x (7) Malfurion the Pestilent # AAEBAZarBA7+Dd8VhBe0uwKZ0wL4oQP9sAOIsQOntQOj9gOJiwTipAWs0QWr4AUlig6P9gL5rQOsgAS4oAswpQSA1ASB1AQAAA== # # To use this deck, copy it to your clipboard and create a new deck in Hearthstone	### Octerion # Class: Druid # Format: Wild # # 2x (1) Biology Project # 2x (1) Druid of the Reef # 1x (1) Jade Idol # 1x (1) Lost in the Park # 2x (1) Secure the Deck # 1x (1) Sir Finley Mrrgglton # 1x (2) Astalor Bloodsworn # 2x (2) Capture Coldtooth Mine # 2x (2) Druid of the Saber # 2x (2) Grizzled Wizard # 2x (2) Wrath # 1x (3) Bad Luck Albatross # 2x (3) Feral Rage # 1x (3) Photographer Fizzle # 2x (4) Poison Seeds # 1x (4) Rhythm and Roots # 1x (4) Widowbloom Seedsman # 1x (5) Kobold Stickyfinger # 1x (5) Nourish # 1x (5) Wildheart Guff # 1x (7) Malfurion the Pestilent # AAEBAZarBA7fFYQXtLsCmdMC+KED/bADiLEDo/YDiYsE2p8Ewd8E4qQFrNEFq+AFCloOj/YC+a0Dp7UDrlAEuKAEsKUEgNQEAAA= # # To use this deck, copy it to your clipboard and create a new deck in Hearthstone	### Octerion # Class: Druid # Format: Wild # # 1x (1) Biology Project # 2x (1) Druid of the Reef # 1x (1) Jade Idol # 1x (1) Lost in the Park # 2x (1) Secure the Deck # 1x (1) Untapped Potential # 1x (2) Astalor Bloodsworn # 2x (2) Capture Coldtooth Mine # 1x (2) Grizzled Wizard # 2x (2) Wrath # 1x (3) Bad Luck Albatross # 2x (3) Feral Rage # 1x (3) Photographer Fizzle # 1x (3) Wardruid Loti # 1x (4) Flobbidinous Floop # 2x (4) Poison Seeds # 1x (4) Rhythm and Roots # 2x (4) Widowbloom Seedsman # 1x (5) Kobold Stickyfinger # 1x (5) Nourish # 1x (5) Wildheart Guff # 1x (7) Malfurion the Pestilent # 1x (10) Tyrantus # AAEBAZarBBC0uwLixwKu0gKZ0wKP9gL1/AL4oQP9sAOIsQOntQOj9gOJiwTanwTipAWs0QWr4AUHig75rQOsgAS4oASwpQSA1ATB3wQAAA== # # To use this deck, copy it to your clipboard and create a new deck in Hearthstone	# 2x (1) jo		
4	4	8	23	22	22	23	1	0	0

octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Card-o-mat

1			1	Biology Project	Biology Project	
2			2	Druid of the Reef	Druid of the Reef	
3			3	Spreading Plague	Spreading Plague	
4			4	Malfurion the Pestilent	Malfurion the Pestilent	
5	4	1		Biology Project	Jade Idol	
6				Druid of the Reef	Lost in the Park	1
7			5	Jade Idol	Naturalize	2
8				Malfurion the Pestilent	Secure the Deck	
9	4	2		Biology Project	Grizzled Wizard	3
10				Druid of the Reef	Jerry Rig Carpenter	4
11				Jade Idol	Sir Finley Mrrgglton	
12			6	Lost in the Park	Untapped Potential	5
13			7	Naturalize	Astalor Bloodsworn	6
14			8	Secure the Deck	Capture Coldtooth Mine	7
15			9	Grizzled Wizard	Druid of the Saber	8
16			10	Jerry Rig Carpenter	Wrath	9
17	8	3		Biology Project	Bad Luck Albatross	
18				Druid of the Reef	Deathlord	
19				Jade Idol	Feral Rage	
20				Lost in the Park	Photographer Fizzle	
21				Secure the Deck	Savage Roar	
22			11	Sir Finley Mrrgglton	Poison Seeds	
23			12	Untapped Potential	Rhythm and Roots	
24			13	Astalor Bloodsworn	Kobold Stickyfinger	
25			14	Capture Coldtooth Mine	Wildheart Guff	
26			15	Druid of the Saber	Living Roots	
27				Grizzled Wizard	Widowbloom Seedsman	
28				Jerry Rig Carpenter	Nourish	
29			16	Wrath	Wardruid Loti	
30			17	Bad Luck Albatross	Flobbidinous Floop	
31			18	Deathlord	Tyrantus	
32			19	Feral Rage	jo	10
33			20	Photographer Fizzle		
34			21	Savage Roar		
35			22	Poison Seeds		
36			23	Rhythm and Roots		
37			24	Kobold Stickyfinger		
38			25	Wildheart Guff		
39				Malfurion the Pestilent		
40	23	4		Biology Project		
41				Druid of the Reef		
42				Jade Idol		
43			26	Living Roots		
44				Lost in the Park		
45				Secure the Deck		
46				Sir Finley Mrrgglton		
47				Untapped Potential		
48				Astalor Bloodsworn		
49				Capture Coldtooth Mine		
50				Druid of the Saber		
51				Grizzled Wizard		
52				Wrath		
53				Bad Luck Albatross		
54				Deathlord		
55				Feral Rage		
56				Photographer Fizzle		
57				Poison Seeds		
58				Rhythm and Roots		
59				Kobold Stickyfinger		
60				Wildheart Guff		
61				Malfurion the Pestilent		
62	22	5		Biology Project		
63				Druid of the Reef		
64				Jade Idol		
65				Lost in the Park		
66				Secure the Deck		
67				Sir Finley Mrrgglton		
68				Untapped Potential		
69				Astalor Bloodsworn		
70				Capture Coldtooth Mine		
71				Druid of the Saber		
72				Grizzled Wizard		
73				Wrath		
74				Bad Luck Albatross		
75				Feral Rage		
76				Photographer Fizzle		
77				Poison Seeds		
78				Rhythm and Roots		
79			27	Widowbloom Seedsman		
80				Kobold Stickyfinger		
81			28	Nourish		
82				Wildheart Guff		
83				Malfurion the Pestilent		
84	22	6		Biology Project		
85				Druid of the Reef		
86				Jade Idol		
87				Lost in the Park		
88				Secure the Deck		
89				Untapped Potential		
90				Astalor Bloodsworn		
91				Capture Coldtooth Mine		
92				Grizzled Wizard		
93				Wrath		
94				Bad Luck Albatross		
95				Feral Rage		
96				Photographer Fizzle		
97			29	Wardruid Loti		
98			30	Flobbidinous Floop		
99				Poison Seeds		
100				Rhythm and Roots		
101				Widowbloom Seedsman		
102				Kobold Stickyfinger		
103				Nourish		
104				Wildheart Guff		
105				Malfurion the Pestilent		
106			31	Tyrantus		
107	23	7	32	jo		
108	1	8	33			
109		9	34			
110		10	35			
111			36			
112			37			
113			38			
114			39			
115			40			
116			41			
117			42			
118			43			
119			44			
120			45			
121			46			
122			47			
123			48			
124			49			
125			50			
126			51			
127			52			
128			53			
129			54			
130			55			
131			56			
132			57			
133			58			
134			59			
135			60			
136			61			
137			62			
138			63			
139			64			
140			65			
141			66			
142			67			
143			68			
144			69			
145			70			
146			71			
147			72			
148			73			
149			74			
150			75			
151			76			
152			77			
153			78			
154			79			
155			80			
156			81			
157			82			
158			83			
159			84			
160			85			
161			86			
162			87			
163			88			
164			89			
165			90			
166			91			
167			92			
168			93			
169			94			
170			95			
171			96			
172			97			
173			98			
174			99			
175			100			
176			101			
177			102			
178			103			
179			104			
180			105			
181			106			
182			107			
183			108			
184			109			
185			110			
186			111			
187			112			
188			113			
189			114			
190			115			
191			116			
192			117			
193			118			
194			119			
195			120			
196			121			
197			122			
198			123			
199			124			
200			125			
201			126			
202			127			
203			128			
204			129			
205			130			
206			131			
207			132			
208			133			
209			134			
210			135			
211			136			
212			137			
213			138			
214			139			
215			140			
216			141			
217			142			
218			143			
219			144			
220			145			
221			146			
222			147			
223			148			
224			149			
225			150			
226			151			
227			152			
228			153			
229			154			
230			155			
231			156			
232			157			
233			158			
234			159			
235			160			
236			161			
237			162			
238			163			
239			164			
240			165			
241			166			
242			167			
243			168			
244			169			
245			170			
246			171			
247			172			
248			173			
249			174			
250			175			
251			176			
252			177			
253			178			
254			179			
255			180			
256			181			
257			182			
258			183			
259			184			
260			185			
261			186			
262			187			
263			188			
264			189			
265			190			
266			191			
267			192			
268			193			
269			194			
270			195			
271			196			
272			197			
273			198			
274			199			
275			200			
276			201			
277			202			
278			203			
279			204			
280			205			
281			206			
282			207			
283			208			
284			209			
285			210			
286			211			
287			212			
288			213			
289			214			
290			215			
291			216			
292			217			
293			218			
294			219			
295			220			
296			221			
297			222			
298			223			
299			224			
300			225			

octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Card-o-mat | Paper Print Layout | Source Code

octup	X:2	X:3	X:4	X:5	X:6 Y:1
'0'	''	''	''	''	Y:2
'1'	=WENN(A1=BEREICH.VERSCHIEBEN('octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Decks'::$A$5;0;MAX(C$1:C1))+SUMME(B$1:B1);BEREICH.VERSCHIEBEN('octup.us \| App: Hearthstone \| App.let: Deck Evolver \| Chapter: Tutorials \| D&L: Decks'::$A$5;0;MAX(C$1:C1));0)	=WENN(BEREICH.VERSCHIEBEN('octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Decks'::$A$5;0;MAX(C$1:C1))+SUMME(B$1:B1)<A2;MAX(C$1:C1)+1;0)	=WENNFEHLER(WENN(VERGLEICH(E2;E$1:E1;0)>0;;MAX(D$1:D1)+1);MAX(D$1:D1)+1)	=REGEX.EXTRAHIEREN(BEREICH.VERSCHIEBEN('octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Decks'::$A$4;0;MAX(C$1:C2));'octup.us \| App: Hearthstone \| App.let: Deck Evolver \| Chapter: Tutorials \| Configuration'::$C$2;A2−SUMME(B$1:B2))	Y:3
'2'	=WENN(A2=BEREICH.VERSCHIEBEN('octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Decks'::$A$5;0;MAX(C$1:C2))+SUMME(B$1:B2);BEREICH.VERSCHIEBEN('octup.us \| App: Hearthstone \| App.let: Deck Evolver \| Chapter: Tutorials \| D&L: Decks'::$A$5;0;MAX(C$1:C2));0)	=WENN(BEREICH.VERSCHIEBEN('octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Decks'::$A$5;0;MAX(C$1:C2))+SUMME(B$1:B2)<A3;MAX(C$1:C2)+1;0)	=WENNFEHLER(WENN(VERGLEICH(E3;E$1:E2;0)>0;;MAX(D$1:D2)+1);MAX(D$1:D2)+1)	=REGEX.EXTRAHIEREN(BEREICH.VERSCHIEBEN('octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Decks'::$A$4;0;MAX(C$1:C3));'octup.us \| App: Hearthstone \| App.let: Deck Evolver \| Chapter: Tutorials \| Configuration'::$C$2;A3−SUMME(B$1:B3))	Y:4
'3'	'0'	'0'	'3'	'Spreading Plague'	Y:5
'4'	'0'	'0'	'4'	'Malfurion the Pestilent'	Y:6
'5'	'4'	'1'	'0'	'Biology Project'	Y:7
'6'	'0'	'0'	'0'	'Druid of the Reef'	Y:8
'7'	'0'	'0'	'5'	'Jade Idol'	Y:9
'8'	'0'	'0'	'0'	'Malfurion the Pestilent'	Y:10
'9'	'4'	'2'	'0'	'Biology Project'	Y:11
'10'	'0'	'0'	'0'	'Druid of the Reef'	Y:12
'11'	'0'	'0'	'0'	'Jade Idol'	Y:13
'12'	'0'	'0'	'6'	'Lost in the Park'	Y:14
'13'	'0'	'0'	'7'	'Naturalize'	Y:15
'14'	'0'	'0'	'8'	'Secure the Deck'	Y:16
'15'	'0'	'0'	'9'	'Grizzled Wizard'	Y:17
'16'	'0'	'0'	'10'	'Jerry Rig Carpenter'	Y:18
'17'	'8'	'3'	'0'	'Biology Project'	Y:19
'18'	'0'	'0'	'0'	'Druid of the Reef'	Y:20
'19'	'0'	'0'	'0'	'Jade Idol'	Y:21
'20'	'0'	'0'	'0'	'Lost in the Park'	Y:22
'21'	'0'	'0'	'0'	'Secure the Deck'	Y:23
'22'	'0'	'0'	'11'	'Sir Finley Mrrgglton'	Y:24
'23'	'0'	'0'	'12'	'Untapped Potential'	Y:25

octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Card-o-mat | Paper Print Layout

octup	X:2	X:3	X:4	X:5	X:6 Y:1
'0'	'0'	'0'	'0'	'0'	Y:2
'1'	'0'	'0'	'1'	'Biology Project'	Y:3
'2'	'0'	'0'	'2'	'Druid of the Reef'	Y:4
'3'	'0'	'0'	'3'	'Spreading Plague'	Y:5
'4'	'0'	'0'	'4'	'Malfurion the Pestilent'	Y:6
'5'	'4'	'1'	'0'	'Biology Project'	Y:7
'6'	'0'	'0'	'0'	'Druid of the Reef'	Y:8
'7'	'0'	'0'	'5'	'Jade Idol'	Y:9
'8'	'0'	'0'	'0'	'Malfurion the Pestilent'	Y:10
'9'	'4'	'2'	'0'	'Biology Project'	Y:11
'10'	'0'	'0'	'0'	'Druid of the Reef'	Y:12
'11'	'0'	'0'	'0'	'Jade Idol'	Y:13
'12'	'0'	'0'	'6'	'Lost in the Park'	Y:14
'13'	'0'	'0'	'7'	'Naturalize'	Y:15
'14'	'0'	'0'	'8'	'Secure the Deck'	Y:16
'15'	'0'	'0'	'9'	'Grizzled Wizard'	Y:17
'16'	'0'	'0'	'10'	'Jerry Rig Carpenter'	Y:18
'17'	'8'	'3'	'0'	'Biology Project'	Y:19
'18'	'0'	'0'	'0'	'Druid of the Reef'	Y:20
'19'	'0'	'0'	'0'	'Jade Idol'	Y:21
'20'	'0'	'0'	'0'	'Lost in the Park'	Y:22
'21'	'0'	'0'	'0'	'Secure the Deck'	Y:23
'22'	'0'	'0'	'11'	'Sir Finley Mrrgglton'	Y:24
'23'	'0'	'0'	'12'	'Untapped Potential'	Y:25

This Deck has
Y:1 running i (Hardcoded)
Y:2 Counter that WHEN ALL Cards of a given Deckcode are evaluated RESULTS in the Count of how many Cards this have been so far. Or 0 otherwise. (thatisimportant).
Y:3 running n for Deckcodes (dynamic)
Y:4 Counter of UNIQUE Cards. That one is key for further functionality!
Y:5 Cardname as String

octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Card-o-mat | Print: Input

octup	X:2	X:3	X:4	X:5	X:6 Y:1
0	0	0	0	0	Y:2
1	0	0	1	Biology Project	Y:3
2	0	0	2	Druid of the Reef	Y:4
3	0	0	3	Spreading Plague	Y:5
4	0	0	4	Malfurion the Pestilent	Y:6
5	4	1	0	Biology Project	Y:7
6	0	0	0	Druid of the Reef	Y:8
7	0	0	5	Jade Idol	Y:9
8	0	0	0	Malfurion the Pestilent	Y:10
9	4	2	0	Biology Project	Y:11
10	0	0	0	Druid of the Reef	Y:12
11	0	0	0	Jade Idol	Y:13
12	0	0	6	Lost in the Park	Y:14
13	0	0	7	Naturalize	Y:15
14	0	0	8	Secure the Deck	Y:16
15	0	0	9	Grizzled Wizard	Y:17
16	0	0	10	Jerry Rig Carpenter	Y:18
17	8	3	0	Biology Project	Y:19
18	0	0	0	Druid of the Reef	Y:20
19	0	0	0	Jade Idol	Y:21
20	0	0	0	Lost in the Park	Y:22
21	0	0	0	Secure the Deck	Y:23
22	0	0	11	Sir Finley Mrrgglton	Y:24
23	0	0	12	Untapped Potential	Y:25

octup.us | Chapter: Examples | Set: Hello World & Co. |

octup	B	C	D	E	EOL_1
0	0	0	0	0	EOL_2
1	0	0	1	Biology Project	EOL_3
2	0	0	2	Druid of the Reef	EOL_4
3	0	0	3	Spreading Plague	EOL_5
4	0	0	4	Malfurion the Pestilent	EOL_6
5	4	1	0	Biology Project	EOL_7
6	0	0	0	Druid of the Reef	EOL_8
7	0	0	5	Jade Idol	EOL_9
8	0	0	0	Malfurion the Pestilent	EOL_10
9	4	2	0	Biology Project	EOL_11
10	0	0	0	Druid of the Reef	EOL_12
11	0	0	0	Jade Idol	EOL_13
12	0	0	6	Lost in the Park	EOL_14
13	0	0	7	Naturalize	EOL_15
14	0	0	8	Secure the Deck	EOL_16
15	0	0	9	Grizzled Wizard	EOL_17
16	0	0	10	Jerry Rig Carpenter	EOL_18
17	8	3	0	Biology Project	EOL_19
18	0	0	0	Druid of the Reef	EOL_20
19	0	0	0	Jade Idol	EOL_21
20	0	0	0	Lost in the Park	EOL_22
21	0	0	0	Secure the Deck	EOL_23
22	0	0	11	Sir Finley Mrrgglton	EOL_24
23	0	0	12	Untapped Potential	EOL_25
24	0	0	13	Astalor Bloodsworn	EOF

octup.us | App: Hearthstone | App.let: Deck Evolver | Chapter: Tutorials | D&L: Card-o-mat-1

1			1	Biology Project	Biology Project	
2			2	Druid of the Reef	Druid of the Reef	
3			3	Spreading Plague	Spreading Plague	
4			4	Malfurion the Pestilent	Malfurion the Pestilent	
5	4	1		Biology Project	Jade Idol	
6				Druid of the Reef	Lost in the Park	
7			5	Jade Idol	Naturalize	
8				Malfurion the Pestilent	Secure the Deck	
9	4	2		Biology Project	Grizzled Wizard	
84	22	6		Biology Project		
85				Druid of the Reef		
86				Jade Idol		
87				Lost in the Park		
88				Secure the Deck		
89				Untapped Potential		
90				Astalor Bloodsworn		
91				Capture Coldtooth Mine		
92				Grizzled Wizard		
93				Wrath		
94				Bad Luck Albatross		
95				Feral Rage		
96				Photographer Fizzle		
97			29	Wardruid Loti		
98			30	Flobbidinous Floop		
99				Poison Seeds		
100				Rhythm and Roots		
101				Widowbloom Seedsman		
102				Kobold Stickyfinger		
103				Nourish		
104				Wildheart Guff		
105				Malfurion the Pestilent		
106			31	Tyrantus		
107	23	7	32	jo		
108	1	8	33			
109		9	34			
110		10	35			
111			36			
112			37			

octup.us | App.let | Chapter: Examples | D&L: Classes | DEBUG

102				Kobold Stickyfinger			
103				Nourish			
104				Wildheart Guff			
105				Malfurion the Pestilent			
106			31	Tyrantus			
107	23	7	32	jo			
108	1	8	33	SEVERE ERROR			
109		9	34	SEVERE ERROR			
110		10	35	SEVERE ERROR			
111	SEVERE ERROR	SEVERE ERROR	36	SEVERE ERROR			
112	SEVERE ERROR	SEVERE ERROR	37	SEVERE ERROR			
113	SEVERE ERROR	SEVERE ERROR	38	SEVERE ERROR			

octup.us Scope | 1 0 1 0

octup.us | Executive

	Executive Summary:	
	octup.us is a unique and innovative technological platform that operates within the framework of Apple Numbers. It provides efficient solutions for managing and manipulating complex data structures. With potential applications like Hearthstone, octup.us can streamline the management of intricate gaming data. Moreover, it offers elegant solutions to common programming challenges such as handling object-like structures, creating Fibonacci sequences or calculating factorials.	octup.us is a unique method in providing App.lication Functionality. She is roughly comparable with cellular automata or assembler code, that comes with it’s dedicated memory for each instruction. Hence octup.us can manage manipulating complex data structures in an efficient way. There are a couple of examples, which showcase a broad range of Use Cases.
	Representing a multi-layered abacus, octup.us functions as a dynamic 'Deck of Gears'. The platform's design is grounded on individual elements known as 'Gears' nestled within data containers referred to as 'Decks'. Gears serve as the processing units, whereas Decks function as dynamic data containers. This interconnected design ensures fluid data manipulation, where changes in one element immediately influence and update all linked entities.	Organizing octup.us Code in a geometrical, multi-layered like abacus enables a fluid transition between Chapter of Gears and Decks across different Hardware implementations. You could have one octup.us, where certain Decks are implemented via QBits, but the octup.us Code is agnostic from Runtime or Hardware it runs on.
	The structure of octup.us, as showcased in the 'Generat-o-mat' deck, encompasses multiple layers such as Apps, App.lets, Chapters, Configurations, and Data & Logic (D&L), among others. These components work in harmony to create a comprehensive and efficient system. The Generat-o-mat deck provides a thorough explanation of the nomenclature and structure of the octup.us platform.	I’d like to emphasize in this context, that Realtime critical implementations are possible within the very basis of octup.us.
	A key objective for the octup.us team is to enable the octup.us code to run on different platforms like Apple Numbers, HTML / JavaScript, and even dedicated hardware configurations. The aim isn't just to translate the code to different languages, but to establish a system that, given the same octup.us source code, can generate hardware layouts dedicated to specific computations. Such a setup contrasts with the traditional CPU approach, which performs general computations based on loaded program code.	octup.us Code should be translatable, compiled to, run on, executed via multiple possibilities.
	Moreover, octup.us holds the potential to incorporate quantum phenomena. The goal is to explore possibilities of using gears that could exploit quantum properties, such as adding photon frequencies or interfering and/or entangling polarization states to generate a result in response to input data.	
	Embodying its mission in the phrase, "octup.us creates soft hard ware," octup.us highlights its ability to convert software solutions into hardware implementations, symbolizing software that possesses hardware capabilities. This slogan underscores its potential for a wide spectrum of applications, from contemporary digital platforms to future quantum technologies. The platform operates efficiently, reflecting its capacity to perform optimally with minimal resources. The work of octup.us is accessible on GitHub at https://github.com/0ctupus and on Patreon at https://www.patreon.com/octup_us.	