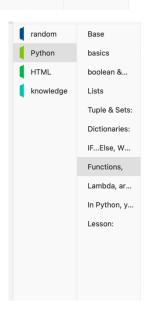


A lambda function can take any number of arguments, but can only have one expression. x = lambda a : a + 10 print(x(5)) Pl ez 15 lesz Barmennyi szammal lehet mukodtetni x = lambda a, b : a * b
print(x(5, 6)) 30 lesz Osszeadast lehet vele csinalni x = lambda a, b, c: a + b + cprint(x(5, 6, 2))The power of lambda is better shown when you use them as an anonymous function inside another function. Say you have a function definition that takes one argument, and that argument will be multiplied with an unknown number: def myfunc(n): return lambda a: a * n Use that function definition to make a function that always doubles or triples the number you send in: def myfunc(n): return lambda a : a * n mydoubler = myfunc(2)
print(mydoubler(11)) Or, use the same function definition to make both functions, in the same program: def mvfunc(n): return lambda a : a * n mydoubler = myfunc(2) mytripler = myfunc(3)



A function is a block of code which only runs when it is called You can pass data, known as parameters, into a function A function can return data as a result. In Python a function is defined using the def keyword: def my_function():
print("Hello from a function") Hogy elohivj egy functiont name() def my_function():
 print("Hello from a function")
my_function() Arguments/parameters - Information can be passed into functions as arguments. Arguments are specified after the function name, inside the parentheses. You can add as many arguments as you want, just separate them with a comma. def my_function(fname):
 print(fname + " Refsr my_function("Emil")
my_function("Tobias")
my_function("Linus") def add_five(num1): print(num1 + 5)userInput = int(input("Enter a number")) add_five(userInput) Ez peldaul minden vegere odarakja a Refsnes nevet Szamit az hogy ha a function pl. 2 args-ra szamit akkor 2 argsnak kell lennie. ha nem tudod