* Overview of lambda functions
* Develop a function to compute some of integers in a list
* Develop a function to compute sum of squares of integers in a list
* Develop a function to compute sum of commission amounts from a list where each element is a tuple which contain **sale amount** and **commission %**.
* Switch to one function **my\_sum** and invoke using lambda functions.
* Advantages of lambda functions
* Usage of lambda functions
* Example of filter function
* Example of map function

1. What is a lambda function in Python?

A) A function without a name

B) A function that takes only one argument

C) A function that takes any number of arguments

D) A function that can only be used once

Answer: A) A function without a name

1. What is the syntax for defining a lambda function in Python?

A) lambda function\_name(arguments): expression

B) def lambda function\_name(arguments): expression

C) lambda (arguments): expression

D) def (arguments): expression lambda function\_name

Answer: A) lambda function\_name(arguments): expression

1. What is the purpose of a lambda function?

A) To reduce the number of lines of code needed to write a function

B) To create a function that can be used only once

C) To create a function that takes only one argument

D) To create a function that is more efficient than a regular function

Answer: B) To create a function that can be used only once

1. Write a lambda function to add two numbers.

A) lambda x, y: return x + y

B) lambda x, y: x + y

C) lambda add(x, y): x + y

D) lambda x + y

Answer: B) lambda x, y: x + y

1. Write a function using lambda to compute the sum of integers in a list.

A) my\_sum = lambda lst: sum(lst)

B) my\_sum = lambda lst: reduce(lambda x, y: x + y, lst)

C) my\_sum = lambda lst: [x for x in lst if type(x) == int]

D) my\_sum = lambda lst: [x for x in lst if x % 2 == 0]

Answer: A) my\_sum = lambda lst: sum(lst)

1. Write a function using lambda to compute the sum of squares of integers in a list.

A) my\_sum = lambda lst: sum(lst \*\* 2)

B) my\_sum = lambda lst: reduce(lambda x, y: x \*\* 2 + y \*\* 2, lst)

C) my\_sum = lambda lst: [x \*\* 2 for x in lst if type(x) == int]

D) my\_sum = lambda lst: [x \*\* 2 for x in lst if x % 2 == 0]

Answer: C) my\_sum = lambda lst: [x \*\* 2 for x in lst if type(x) == int]

1. Write a function using lambda to compute the sum of commission amounts from a list where each element is a tuple which contain sale amount and commission %.

A) my\_sum = lambda lst: sum(lst)

B) my\_sum = lambda lst: sum([x[0] \* x[1] for x in lst])

C) my\_sum = lambda lst: sum([x[0] + x[1] for x in lst])

D) my\_sum = lambda lst: sum([x[0] \* (1 + x[1]) for x in lst])

Answer: B) my\_sum = lambda lst: sum([x[0] \* x[1] for x in lst])

1. What are the advantages of using lambda functions in Python?

A) They are more efficient than regular functions

B) They allow for concise code with less typing

C) They are easier to debug than regular functions

D) They are more flexible than regular functions

Answer: B) They allow for concise code with less typing

1. In what situations are lambda functions particularly useful?

A) When working with large datasets

B) When performing complex mathematical operations

C) When creating small, one-time-use functions

D) When working with object-oriented programming

Answer: C) When creating small, one-time-use functions

1. What is the filter() function in Python used for?

A) To apply a function to each element in an iterable and return the result

B) To combine two iterables into a single iterable

C) To remove elements from an iterable based on a condition

D) To sort elements in an iterable

Answer: C) To remove elements from an iterable based on a condition