In this Video, we are going to take a look at…

- What a lambda expression is
- Structure of a lambda expression
- How to write a lambda expression

What Is a Lambda Expression?

- An alternative way to write functions instead of using the traditional `def <function name>` syntax
- Generally used to create anonymous functions for one-time use only
- Can be combined with `map()`, `filter()`, `reduce()`, and so on to perform various operations on iterables

Structure of a Lambda Expression

- Lambda expressions take a comma-separated sequence of parameters as input
- The expression following the colon is the function body
- There is no explicit return statement

```
lambda par1, par2,... : expression
```

Structure of a Lambda Expression

- Lambda expressions take a comma-separated sequence of parameters as input
- The expression following the colon is the function body
- There is no explicit return statement
- Can be combined with `map()`, `filter()`, `reduce()`, and so on to perform various operations on iterables

```
lambda par1, par2,... : expression
```

Anonymous Functions Using Lambda

- When we don’t assign the lambda expression to a variable name, it’s called an anonymous function
- Generally used to create functions that are to be used once only

Summary

- Lambda expressions are an alternative to writing functions
- Consist of multiple parameters but only expressions
- When a lambda function is not assigned to a variable name, it is called an anonymous function
- Anonymous functions can be used as input for other functions