

Exercises

- 1) Create a lambda function that adds two numbers.
- 2) Create a lambda function that multiplies two numbers.
- 3) Create a lambda function that subtracts two numbers.
- 4) Create a lambda function that divides two numbers.
- 5) Create a lambda function that returns the square of a number.
- 6) Create a lambda function that returns the cube of a number.
- 7) Create a lambda function that checks if a number is even.
- 8) Create a lambda function that checks if a number is odd.
- 9) Create a lambda function that returns the maximum of two numbers.
- 10) Create a lambda function that returns the minimum of two numbers.

Exercises and solution

- 1) Create a lambda function that adds two numbers.

```
add = lambda x, y: x + y
```

- 2) Create a lambda function that multiplies two numbers.

```
multiply = lambda x, y: x * y
```

- 3) Create a lambda function that subtracts two numbers.

```
subtract = lambda x, y: x - y
```

4) Create a lambda function that divides two numbers.

```
divide = lambda x, y: x / y
```

5) Create a lambda function that returns the square of a number.

```
square = lambda x: x**2
```

6) Create a lambda function that returns the cube of a number.

```
cube = lambda x: x**3
```

7) Create a lambda function that checks if a number is even.

```
is_even = lambda x: x % 2 == 0
```

8) Create a lambda function that checks if a number is odd.

```
is_odd = lambda x: x % 2 != 0
```

9) Create a lambda function that returns the maximum of two numbers.

```
maximum = lambda x, y: x if x > y else y
```

10) Create a lambda function that returns the minimum of two numbers.

```
minimum = lambda x, y: x if x < y else y
```