Exercises

1) Create a lambda function that checks if a number is prime.
2) Create a lambda function that checks if a string is a palindrome.
3) Create a lambda function that reverses a string.
4) Create a lambda function that converts a string to uppercase.
5) Create a lambda function that converts a string to lowercase.
6) Create a lambda function that returns the length of a string.
7) Create a lambda function that returns the first letter of a string.
8) Create a lambda function that returns the last letter of a string.
9) Create a lambda function that returns the first n elements of a list.
10) Create a lambda function that returns the last n elements of a list.

Exercises and solution

1) Check if a number is prime:

```
is_prime = lambda num: all(num % i != 0 for i in range(2, int(num ** 0.5) + 1)) and num > 1
```

2) Check if a string is a palindrome:

```
is_palindrome = lambda s: s == s[::-1]
```

3) Reverse a string:

```
reverse_string = lambda s: s[::-1]
```

4) Convert a string to uppercase:

```
to_uppercase = lambda s: s.upper()
```

5) Convert a string to lowercase:

```
to_lowercase = lambda s: s.lower()
```

6) Return the length of a string:

```
string_length = lambda s: len(s)
```

7) Return the first letter of a string:

```
first_letter = lambda s: s[0]
```

8) Return the last letter of a string:

```
last_letter = lambda s: s[-1]
```

9) Return the first n elements of a list:

```
first_n = lambda lst, n: lst[:n]
```

10) Return the last n elements of a list:

```
last_n = lambda lst, n: lst[-n:]
```