

# What is causing the type error of “expected string or bytes-like object” in this case?

Authored by  
**stats writer**

July 2, 2024

## RECOMMENDED CITATION

stats writer (2024). *What is causing the type error of “expected string or bytes-like object” in this case?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=165601>

The type error of "expected string or bytes-like object" is being caused by the fact that the code is expecting a specific type of variable (either a string or a bytes-like object) to be passed as an argument, but a different type of variable is being provided. This mismatch in variable types is causing the code to not be able to properly execute and results in the type error being raised.

## **Fix: TypeError: expected string or bytes-like object**

**One error you may encounter when using Python is:**

**TypeError: expected string or bytes-like object**

**This error typically occurs when you attempt to use the `re.sub()` function to replace certain patterns in an object but the object you're working with is not composed entirely of strings.**

**The following example shows how to fix this error in practice.**

**How to Reproduce the Error**

**Suppose we have the following list of values:**

**#define list of values**

**X =**

**Now suppose we attempt to replace each non-letter in**

**the list with an empty string:**

```
import re
```

```
#attempt to replace each non-letter with empty string
```

```
x = re.sub(" ", "", x)
```

**TypeError: expected string or bytes-like object**

**We receive an error because there are certain values in the list that are not strings.**

**How to Fix the Error**

**The easiest way to fix this error is to convert the list to a string object by wrapping it in the str() operator:**

```
import re
```

```
#replace each non-letter with empty string
```

```
x = re.sub(" ", "", str(x))
```

```
#display resultsprint(x)
```

**ABCDE**

**Notice that we don't receive an error because we used**

**str()** to first convert the list to a string object.

The result is the original list with each non-letter replaced with a blank.

**Note:** You can find the complete documentation for the `re.sub()` function .

**Additional Resources**

ARABPSYCHOLOGY.COM