

How can I fix the TypeError “cannot perform reduce with flexible type” in my code?

Authored by
stats writer

July 2, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I fix the TypeError “cannot perform reduce with flexible type” in my code?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=165762>

The TypeError "cannot perform reduce with flexible type" typically occurs when there is inconsistency in the data types being used in a reduce function. To fix this error, one can ensure that all the data types being used are compatible, or convert them to a compatible type before performing the reduce operation. Another solution is to use a different function, such as map or filter, instead of reduce. It is important to carefully check the data types being used and make necessary adjustments to avoid this error.

Fix: TypeError: cannot perform reduce with flexible type

One error you may encounter when using Python is:

ValueError: cannot perform reduce with flexible type

This error occurs when you attempt to perform some calculation on an object in Python that is not numeric.

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

How to Reproduce the Error

Suppose we have the following NumPy array:

```
import numpy as np
```

```
#define NumPy array of values
```

```
data = np.array()
```

```
#attempt to calculate median of values
```

np.median(data)

TypeError: cannot perform reduce with flexible type

We receive a TypeError because we attempted to calculate the median of a list of string values.

How to Fix the Error

The easiest way to fix this error is to simply convert the NumPy array to a float object so that we can perform mathematical operations on it.

The following code shows how to do so:

```
#convert NumPy array of string values to float values  
data_new = data.astype(float)
```

```
#view updated NumPy array  
data_new  
array()
```

```
#check data type of array  
data_new.dtype
```

```
dtype('float64')
```

We can now perform mathematical operations on the NumPy array:

#calculate median value of array

np.median(data_new)

5.5

#calculate mean value of array

np.mean(data_new)

6.0

#calculate max value of array

np.max(data_new)

12.0

Notice that we don't receive any errors because the NumPy array is a float object, which means we can perform mathematical operations on it.

Additional Resources

The following tutorials explain how to fix other common errors in Python: