

How can I use the COMPRESS function in SAS and what are some examples of its usage?

Authored by
stats writer

June 26, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I use the COMPRESS function in SAS and what are some examples of its usage?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=154192>

The COMPRESS function in SAS is a powerful tool that allows users to manipulate character strings by removing or replacing specific characters. It can be used to create cleaner and more efficient data sets, as well as to perform data cleaning and data validation tasks. The function takes in two arguments: the string to be compressed and a list of characters to be removed or replaced. By default, it compresses all blank spaces, but users can specify specific characters or strings to be removed. This function is particularly useful when dealing with large datasets with messy or inconsistent data. Some examples of its usage include removing special characters, compressing multiple blanks into single blanks, and removing punctuation marks from text. Overall, the COMPRESS function in SAS is a valuable tool for data manipulation and can greatly improve the efficiency and accuracy of data analysis.

Use the COMPRESS Function in SAS (With Examples)

You can use the COMPRESS function in SAS to remove specific characters from a string.

This function uses the following basic syntax:

COMPRESS(String, characters to be removed)

where:

String: The string to analyze
characters to be removed:
One or more specific characters to remove from string

Here are the four most common ways to use this function:

Method 1: Remove All Blank Spaces from String

```
data new_data;  
set original_data;  
compressed_string = compress(string_variable);  
run;
```

Method 2: Remove Specific Characters from String

```
data new_data;  
set original_data;  
compressed_string = compress(string_variable, '!?@#');  
run;
```

Method 3: Remove All Alphabetical Characters from String

```
data new_data;  
set original_data;  
compressed_string = compress(string_variable, ", 'a');  
run;
```

Method 4: Remove All Numeric Values from String

```
data new_data;  
set original_data;
```

```
compressed_string = compress(string_variable, ", 'd');  
run;
```

The following examples show how to use each method with the following dataset in SAS:

```
/*create dataset*/  
data original_data;  
input name $25.;  
datalines;  
Andy Lincoln4 Bernard!  
Barren Michael55 Smith!  
Chad Simpson7 Arnolds?  
Derrick Parson2 Henry  
Eric Miller2 Johansen!  
Frank Giovanni5 Goode  
;  
run;  
  
/*view dataset*/  
proc printdata=original_data;
```

Obs	name
1	Andy Lincoln4 Bernard!
2	Barren Michael55 Smith!
3	Chad Simpson7 Arnolds?
4	Derrick Parson2 Henry
5	Eric Miller2 Johansen!
6	Frank Giovanni5 Goode

Example 1: Remove All Blank Spaces from String

The following code shows how to remove all blank spaces from each string in the name column:

```
/*remove blank spaces from each string in name  
column*/  
data new_data;  
set original_data;  
compressed_name = compress(name);  
run;  
  
/*view results*/  
proc printdata=new_data;
```

Obs	name	compressed_name
1	Andy Lincoln4 Bernard!	AndyLincoln4Bernard!
2	Barren Michael55 Smith!	BarrenMichael55Smith!
3	Chad Simpson7 Arnolds?	ChadSimpson7Arnolds?
4	Derrick Parson2 Henry	DerrickParson2Henry
5	Eric Miller2 Johansen!	EricMiller2Johansen!
6	Frank Giovanni5 Goode	FrankGiovanni5Goode

Notice that all blank spaces have been removed from each string in the new column called `compressed_name`.

Example 2: Remove Specific Characters from String

The following code shows how to remove all question marks and exclamation points from each string in the name column:

```
/*remove question marks and exclamation points from  
each string in name column*/
```

```
data new_data;
```

```
set original_data;
```

```
compressed_name = compress(name, '?!');
```

```
run;
```

```
/*view results*/
```

```
proc printdata=new_data;
```

Obs	name	compressed_name
1	Andy Lincoln4 Bernard!	Andy Lincoln4 Bernard
2	Barren Michael55 Smith!	Barren Michael55 Smith
3	Chad Simpson7 Arnolds?	Chad Simpson7 Arnolds
4	Derrick Parson2 Henry	Derrick Parson2 Henry
5	Eric Miller2 Johansen!	Eric Miller2 Johansen
6	Frank Giovanni5 Goode	Frank Giovanni5 Goode

Notice that all question marks and exclamation points have been removed from each string in the new column called compressed_name.

Example 3: Remove All Alphabetical Characters from String

The following code shows how to remove all alphabetical characters from each string in the name column:

```
/*remove all alphabetical characters from each string in  
name column*/  
data new_data;  
set original_data;  
compressed_name = compress(name, ", 'a');  
run;
```

```
/*view results*/
```

```
proc printdata=new_data;
```

Obs	name	compressed_name
1	Andy Lincoln4 Bernard!	4!
2	Barren Michael55 Smith!	55!
3	Chad Simpson7 Arnolds?	7?
4	Derrick Parson2 Henry	2
5	Eric Miller2 Johansen!	2!
6	Frank Giovanni5 Goode	5

Notice that all all alphabetical characters have been removed from each string in the new column called `compressed_name`.

Example 4: Remove All Numeric Values from String

The following code shows how to remove all numeric values from each string in the name column:

```
/*remove all numeric values from each string in name  
column*/
```

```
data new_data;
```

```
set original_data;
```

```
compressed_name = compress(name, ", 'd');
```

```
run;
```

```
/*view results*/
```

```
proc printdata=new_data;
```

Obs	name	compressed_name
1	Andy Lincoln4 Bernard!	AndyLincolnBernard!
2	Barren Michael55 Smith!	BarrenMichaelSmith!
3	Chad Simpson7 Arnolds?	ChadSimpsonArnolds?
4	Derrick Parson2 Henry	DerrickParsonHenry
5	Eric Miller2 Johansen!	EricMillerJohansen!
6	Frank Giovanni5 Goode	FrankGiovanniGoode

Notice that all all numeric values have been removed from each string in the new column called compressed_name.

The following tutorials explain how to perform other common tasks in SAS: