

How can the SAS programming language be used to add leading zeros in Z format to values?

Authored by
stats writer

June 23, 2024

RECOMMENDED CITATION

stats writer (2024). *How can the SAS programming language be used to add leading zeros in Z format to values?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=148069>

The SAS programming language can be utilized to add leading zeros in Z format to values by using the "Z" format modifier. This modifier instructs SAS to display numeric values with leading zeros, making the output appear in a consistent and visually appealing manner. By incorporating the "Z" format modifier in a SAS program, users can ensure that their data is correctly formatted and aligned, making it easier to read and understand. This feature is particularly useful when dealing with numerical data that requires a fixed number of digits, such as account numbers or identification numbers. Overall, the use of the "Z" format modifier in SAS allows for more efficient and accurate data manipulation and presentation.

SAS: Use Z Format to Add Leading Zeros to Values

You can use the Z format option in SAS to add leading zeros to numeric values.

The following examples show how to use the Z format option in practice with the following dataset in SAS that shows the total sales made by various employees at some company:

```
/*create dataset*/  
data my_data;  
input employee $ sales;  
datalines;  
A 32  
B 10  
C 24  
D 40  
E 138
```

F 42

G 54

H 9

I 38

J 22

K 18.5

;

run;

/*view dataset*/

proc printdata=my_data;

Obs	employee	sales
1	A	32.0
2	B	10.0
3	C	24.0
4	D	40.0
5	E	138.0
6	F	42.0
7	G	54.0
8	H	9.0
9	I	38.0
10	J	22.0
11	K	18.5

Example 1: Use Z Format with No Decimal Places

We can use the following Z format option to add as

many leading zeros as necessary to make each value in the sales column have a length of 6:

```
/*use Z format to add leading zeros to values in sales column*/
```

```
proc printdata=my_data;
```

```
format sales z6.;
```

```
run;
```

Obs	employee	sales
1	A	000032
2	B	000010
3	C	000024
4	D	000040
5	E	000138
6	F	000042
7	G	000054
8	H	000009
9	I	000038
10	J	000022
11	K	000019

Each value in the sales column now has as many leading zeros as necessary for it to have a length of 6.

Since we didn't specify any value after the decimal place in z6. we told SAS not to display any values after

the decimal place and to simply round each value to the nearest integer.

For example, the last value in the sales column had a value of 18.5 but was rounded to 19 and then had leading zeros added to it in order to create the final value of 000019, which has total length of 6.

Example 2: Use Z Format with Decimal Places

We can use the following Z format option to add as many leading zeros as necessary to make each value in the sales column have a length of 10, including 1 decimal place:

```
/*use Z format to add leading zeros to values in sales  
column*/  
proc printdata=my_data;  
format sales z10.1;  
run;
```

Obs	employee	sales
1	A	00000032.0
2	B	00000010.0
3	C	00000024.0
4	D	00000040.0
5	E	00000138.0
6	F	00000042.0
7	G	00000054.0
8	H	00000009.0
9	I	00000038.0
10	J	00000022.0
11	K	00000018.5

Each value in the sales column now has as many leading zeros as necessary for it to have a length of 10.

Since we used z10.1 we told SAS to display one value after the decimal place of each value.

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