SAS Lesson 08

Explicit Conversion

This demonstration illustrates the explicit conversion of values from numeric to character.

Converting a Variable to Another Data Type

```
proc contents data=hrdata;
run;
```

Partial PROC CONTENTS Output

```
Alphabetic List of Variables and Attributes

   #    Variable    Type    Len

   3    Bonus       Num       8
   2    EmpID       Num       8
   1    GrossPay    Char      6
   4    HireDate    Num       8
```

How can you convert **`GrossPay`** to a numeric variable with the same name?

Quiz

Will this statement convert **GrossPay** to numeric?

```
GrossPay=input(GrossPay,comma6.);
```

No, once data is read into PDV cannot change length or type of variable.

Explicit Conversion

This demonstration illustrates an attempt to change the variable type with a PUT function.

Quiz – Correct Answer

Will this statement convert **GrossPay** to numeric?

```
GrossPay=input(GrossPay,comma6.);
```

No, **GrossPay** remained a character variable. Why?

Converting a Variable to Another Data Type

```
GrossPay=input(GrossPay,comma6.);
```

This assignment statement does **not** change **`GrossPay`** from a character variable to a numeric variable.

A variable is character or numeric. After the variable's type is established, it cannot be changed.

By following three steps, you can create a new variable with the same name and a different type.

Converting a Variable to Another Data Type

Step 1: Use the RENAME= data set option to rename the variable that you want to convert.

```
data hrdata;
   set orion.convert(rename=(GrossPay=
                             CharGross));
run;
```

General form of the RENAME data set option:

SAS-data-set(RENAME=(*old-name*=*new-name*))

Converting a Variable to Another Data Type

Step 2: Use the INPUT function in an assignment statement to create a new variable with the original name of the variable that you renamed.

```
data hrdata;
   set orion.convert(rename=(GrossPay=
                             CharGross));
   GrossPay=input(CharGross,comma6.);
run;
```

Converting a Variable to Another Data Type

Step 3: Use a DROP= data set option in the DATA statement to exclude the original variable from the output SAS data set.

```
data hrdata;
  set orion.convert(rename=(GrossPay=
                            CharGross));
  drop CharGross;
  GrossPay=input(CharGross,comma6.);
run;
```

The compilation for this program shows the PDV being created with a numeric **GrossPay** variable.

Converting a Variable: Compilation

```
data hrdata;
   set orion.convert(rename=(GrossPay=
                             CharGross));
   GrossPay=input(CharGross,comma6.);
   drop CharGross;
run;
```

Partial PDV

ID	CharGross	Hired
$ 5	$ 6	$ 7

...

Converting a Variable: Compilation

```
data hrdata;
   set orion.convert(rename=(GrossPay=
                             CharGross));
   GrossPay=input(CharGross,comma6.);
   drop CharGross;
run;
```

Partial PDV

ID	CharGross	Hired	GrossPay
$ 5	$ 6	$ 7	N 8

...

Converting a Variable: Compilation

```
data hrdata;
   set orion.convert(rename=(GrossPay=
                             CharGross));
   GrossPay=input(CharGross,comma6.);
   drop CharGross;
run;
```

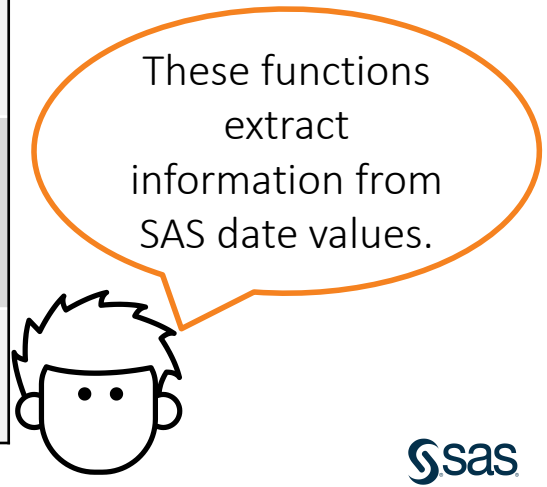
Partial PDV

ID	CharGross	Hired	GrossPay
$ 5	$ 6	$ 7	N 8

Date Functions

SAS Date is presents # of days since 1/1/1960

Function	What It Does
MONTH (*SAS-date*)	Returns a number from 1 through 12 that represents the month
YEAR (*SAS-date*)	Returns the four-digit year
DAY (*SAS-date*)	Returns a number from 1 through 31 that represents the day of the month
WEEKDAY (*SAS-date*)	Returns a number from 1 through 7 that represents the day of the week (Sunday=1)
QTR (*SAS-date*)	Returns a number from 1 through 4 that represents the quarter



Date Functions

Function	What It Does
TODAY() or DATE()	Returns the current date as a numeric SAS date value
MDY (*month*, *day*, *year*)	Returns a SAS date value from month, day, and year values
YRDIF (*startdate*, *enddate*, 'AGE')	Calculates a precise difference in years between two dates
DATDIF(*startdate,enddate, basis*)	Returns number of days between two dates using selected count convention

Extracting Data from a Datetime Value

DATEPART(*datetime-value*)

TIMEPART(*datetime-value*)

```
data storm_detail2;
     set pg2.storm_detail;
     WindDate=datepart(ISO_Time);
     WindTime=timepart(ISO_Time);
     format WindDate date9. WindTime time.;
run;
```

PDV

ISO_Time	WindDate	WindTime
628192800	7270	21600

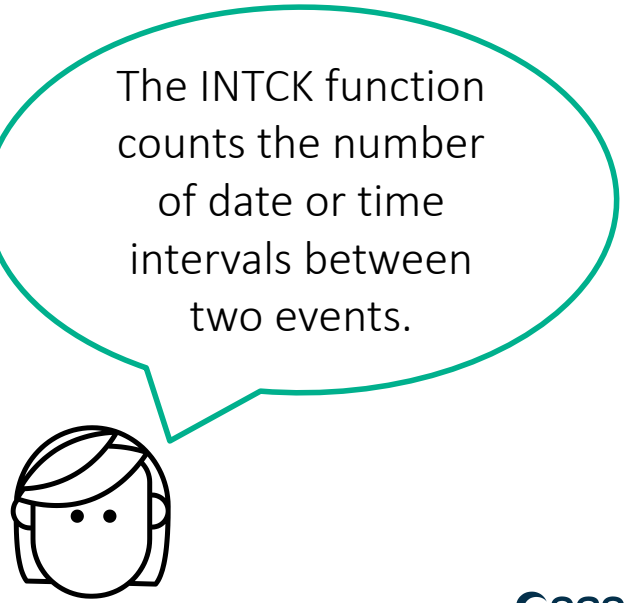
Name	ISO_time	WindDate	WindTime
ALBINE	27NOV1979:06:00:00.00	27NOV1979	6:00:00
ALBINE	27NOV1979:12:00:00.00	27NOV1979	12:00:00
ALBINE	27NOV1979:18:00:00.00	27NOV1979	18:00:00
ALBINE	28NOV1979:00:00:00.00	28NOV1979	0:00:00
ALBINE	28NOV1979:06:00:00.00	28NOV1979	6:00:00

Calculating Date Intervals

INTCK('*interval*', *start-date*, *end-date* <, '*method*'>)

- interval that you want to count
- SAS date columns
- method for calculating intervals

Possible intervals include week, month, year, weekday, or hour.



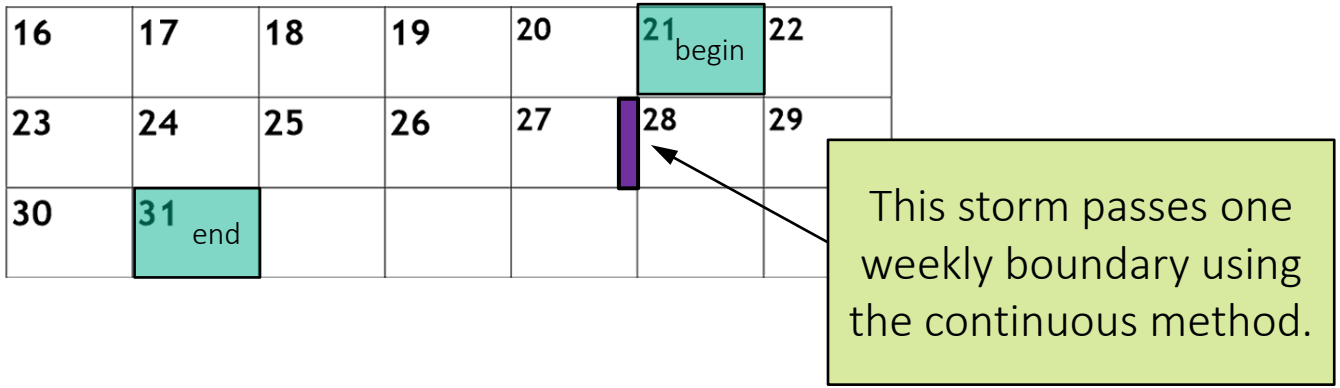
Calculating Date Intervals

Method	
'discrete' 'd'	Each interval has a fixed boundary. For example, a week ends after Saturday, or a year ends on December 31.

This storm passes two weekly boundaries using the default discrete method.

Calculating Date Intervals

Method	
'continuous' 'c'	Each interval is measured relative to the start date or time.



Question

What value would be assigned to **Months2Pay** for each expression?

ServiceDate	PayDate	Months2Pay
10JUL2018	05SEP2018	?

```
Months2Pay=intck('month', ServiceDate, PayDate);
```

```
Months2Pay=intck('month', ServiceDate, PayDate, 'c');
```

Question – Correct Answer

What value would be assigned to **Months2Pay** for each expression?

ServiceDate	PayDate	Months2Pay
10JUL2018	05SEP2018	?

```
Months2Pay=intck('month', ServiceDate, PayDate);
```

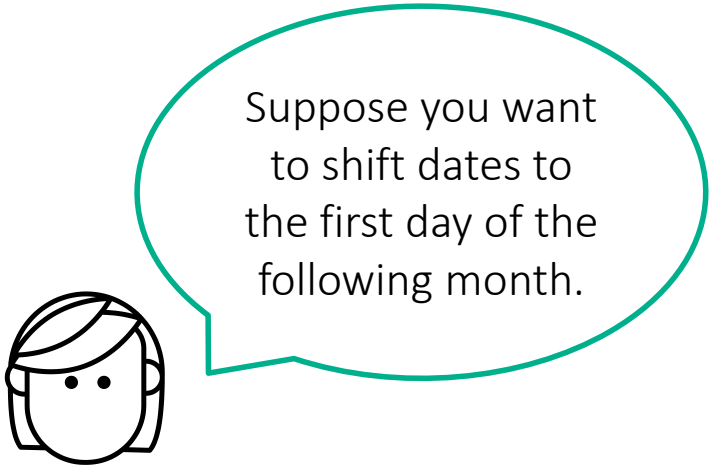
Two end-of-month boundaries were crossed at the end of July and August.

```
Months2Pay=intck('month', ServiceDate, PayDate, 'c');
```

One month boundary was crossed at August 10. The next boundary will not occur until September 10.

Shifting Date Values

Customer ID	SalesDate	BillingDate
12808	10JUL2018	01AUG2018
59601	17JUL2018	01AUG2018
42616	02AUG2018	01SEP2018



Shifting Date Values

INTNX(*interval*, *start*, *increment* <, '*alignment*'>)

- *interval* — interval that you want to shift
- *start* — SAS date column
- *increment* — number of intervals to shift
- *alignment* — position of SAS dates in the interval

Alignments
beginning (default)
middle
end
same

The INTNX function shifts dates or times based on an interval.

Working with Date Values

This demonstration illustrates using the date functions to create values from existing SAS date columns.

Business Scenario – Create a List of Charities

A manager in the Finance department asked for a list of all the charities that Orion Star contributes to. She would like to see the name of the charity as well as the ID code assigned to it.

Here is a sketch of the desired output:

Charity Names and ID Codes

ID	Name
AQI	Aquamissions International
CCI	Cancer Cures, Inc.
CNI	Conserve Nature, Inc.

Input Data

The **orion.biz_list** data set is extracted from the accounting system and contains the names of Orion Star's U.S. suppliers, charities, and consultants.

Partial Listing of **orion.biz_list**

Acct_ Code	Name
AEK3	ANGELA E. KEARNEY
AQI2	AQUAMISSIONS INTERNATIONAL
ATS1	A TEAM SPORTS
CBO3	CLAIRE B. OWENS
CCI2	CANCER CURES, INC.
CNI2	CONSERVE NATURE, INC.
CS1	CAROLINA SPORTS

Input Data – Details

Acct_Code is a character variable defined as length 6. Its last digit represents the type of organization: **1** denotes a supplier, **2** a charity, and **3** a consultant.

The other characters in the **Acct_Code** variable represent the ID for the organization, so the **ID** value can have as many as five characters.

Example:

Acct_Code $6	ID $5
AQI2	AQI

- 2 denotes a charity.
- AQI is the ID.

Input Data – Details

The name of the organization is stored as all capital letters. In the desired output, only the first letter of each word is capitalized.

Example:

Name
AQUAMISSIONS INTERNATIONAL

Change to:

Name
Aquamissions International

Business Scenario – Desired Results

Create a new data set, **charities**, that has the information that the finance manager would like to see.

Partial Listing of **charities**

ID	Acct_ Code	Name
AQI	AQI2	Aquamissions International
CCI	CCI2	Cancer Cures, Inc.
CNI	CNI2	Conserve Nature, Inc.
CS	CS2	Child Survivors
CU	CU2	Cuidadores Ltd.
DAI	DAI2	Disaster Assist, Inc.

This data set can then be used to create the manager's report.

Create the List of Charities – Step 1

The first step is to subset the data based on the last character of **`Acct_Code`**.

Partial Listing of **`orion.biz_list`**

Acct_ Code	Name
AEK3	ANGELA E. KEARNEY
AQI2	AQUAMISSIONS INTERNATIONAL
ATS1	A TEAM SPORTS
CBO3	CLAIRE B. OWENS
CCI2	CANCER CURES, INC.
CNI2	CONSERVE NATURE, INC.
CS1	CAROLINA SPORTS

SAS character functions make this task easy.

The SUBSTR Function (Right Side)

The SUBSTR function on the right side of an assignment statement is used to extract characters.

General form of the SUBSTR function:

NewVar=SUBSTR(*string*,*start*<,*length*>);

string	can be a character constant, variable, or expression.
start	specifies the starting position.
length	specifies the number of characters to extract. If omitted, the substring consists of the remainder of string.
NewVar	If *NewVar* is a new variable it will be created with the same length as *string*. To set a different length for *NewVar*, use a LENGTH statement prior to the assignment statement.

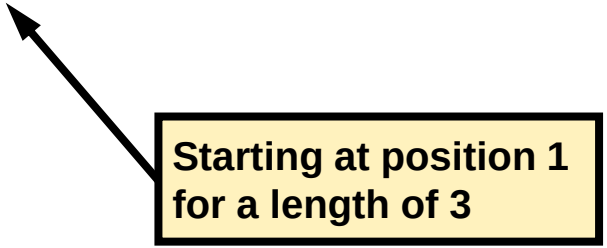
The SUBSTR Function – Example

Extract the first three characters from the value in the **Item_Code** variable and store them in **Item_Type**.

```
Item_Type=substr(Item_Code,1,3);
```

PDV

Item_Code $ 20	Item_Type $ 20
978-1-59994-397-8	978



Setup for the Poll

This is the current value of **`Item_Code`**:

PDV

Item_Code
$ 20
978-1-59994-397-8

The SUBSTR function is a good method to extract the highlighted digits.

Multiple Choice Poll

Which SUBSTR function can extract the group of five numbers from the middle of the **Item_Code** value?

a. `substr(Item_Code,5,7)`

b. `substr(Item_Code,5)`

c. `substr(Item_Code,7,5)`

d. `substr(Item_Code,'mid',5)`

PDV

Item_Code
$ 20
978-1-59994-397-8

Multiple Choice Poll – Correct Answer

Which SUBSTR function can extract the group of five numbers from the middle of the **`Item_Code`** value?

a. **`substr(Item_Code,5,7)`**

b. **`substr(Item_Code,5)`**

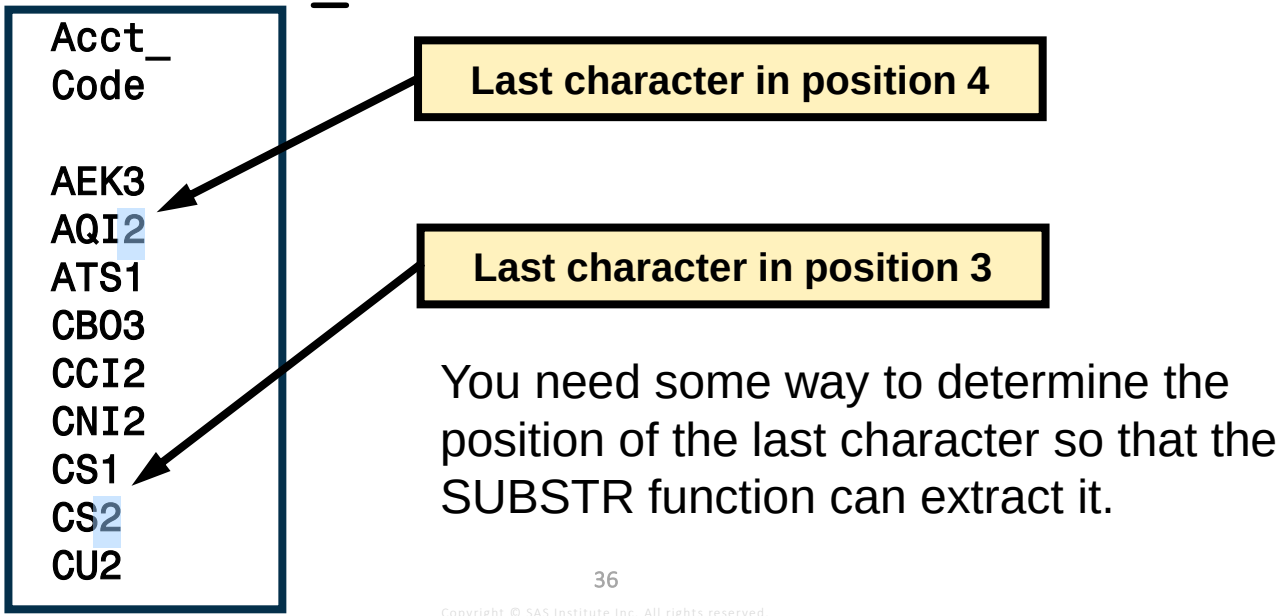
c. **`substr(Item_Code,7,5)`**

d. **`substr(Item_Code,'mid',5)`**

Create the List of Charities – Step 1

The last non-blank character in the **Acct_Code** value occurs in different positions for different observations.

Partial Listing of
orion.biz_list



You need some way to determine the position of the last character so that the SUBSTR function can extract it.

The LENGTH Function

The LENGTH function returns the length of a non-blank character string, excluding trailing blanks.

General form of the LENGTH function:

NewVar=LENGTH(*argument*);

Example:

```
Code = 'ABCD   ';
Last_NonBlank=length(Code);
```

PDV

Code $ 6	Last_NonBlank N 8
ABCD	4

Create the List of Charities – Step 1

This program uses the SUBSTR and LENGTH functions to create the **charities** data set.

The LENGTH function is *nested,* or used as an argument to the SUBSTR function.

```
data charities;
  length ID $ 5;
  set orion.biz_list;
  if substr(Acct_Code,length(Acct_Code),1)='2';
  ID=substr(Acct_Code,1,length(Acct_Code)-1);
run;
```

Partially stepping through the execution for the first charity observation shows how the functions transform the data.

Execution: Step 1

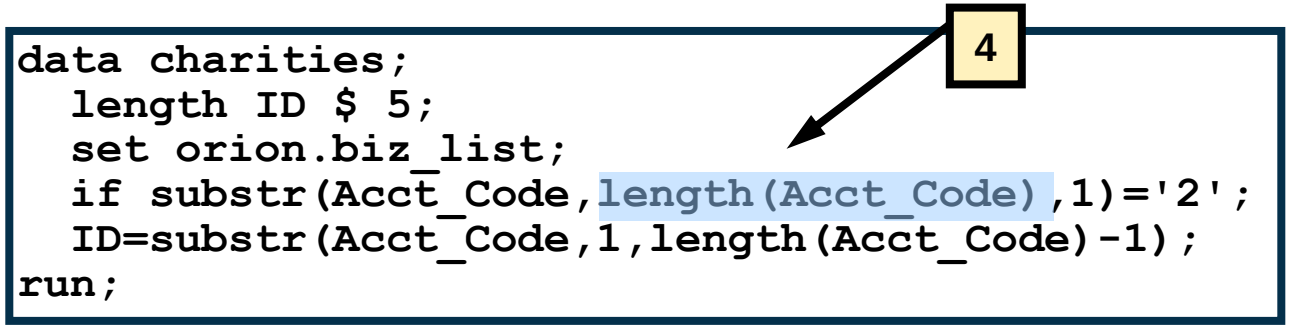
Read the first charity observation.

```
data charities;
  length ID $ 5;
  set orion.biz_list;
  if substr(Acct_Code,length(Acct_Code),1)='2';
  ID=substr(Acct_Code,1,length(Acct_Code)-1);
run;
```

PDV

ID	Acct_Code	Name
$ 5	$ 6	$ 30
	AQI2	AQUAMISSIONS INTERNATIONAL

Execution: Step 1

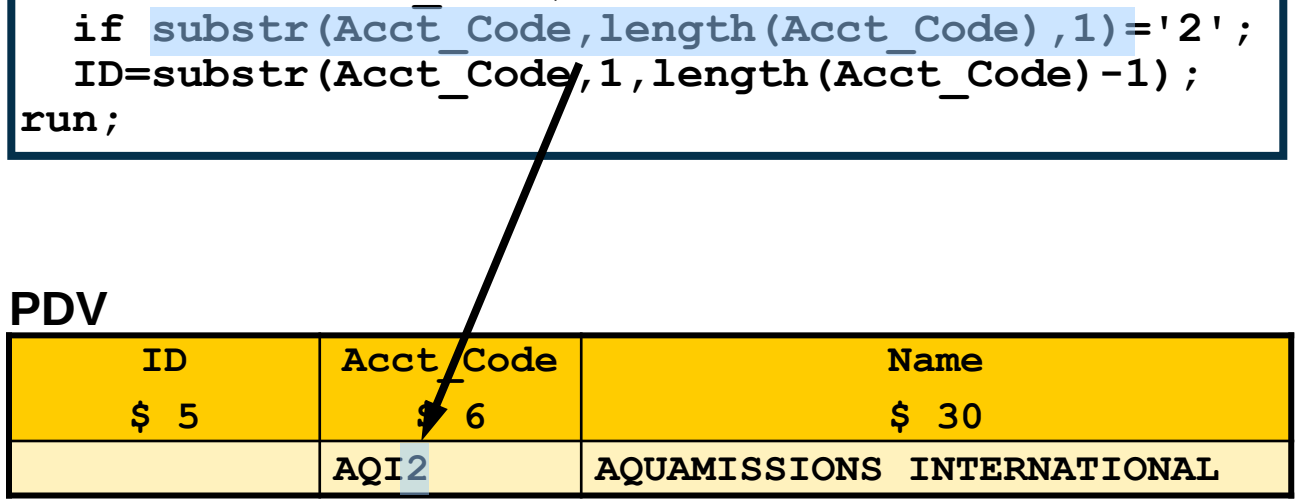


PDV

ID	Acct_Code	Name
$ 5	$ 6	$ 30
	AQI2	AQUAMISSIONS INTERNATIONAL

Execution: Step 1

```
data charities;
  length ID $ 5;
  set orion.biz_list;
  if substr(Acct_Code,length(Acct_Code),1)='2';
  ID=substr(Acct_Code,1,length(Acct_Code)-1);
run;
```



Execution: Step 1

```
data charities;
  length ID $ 5;
  set orion.biz_list;
  if substr(Acct_Code,length(Acct_Code),1)='2';
  ID=substr(Acct_Code,1,length(Acct_Code)-1);
run;
```

True

PDV

ID	Acct_Code	Name
$ 5	$ 6	$ 30
	AQI2	AQUAMISSIONS INTERNATIONAL

Execution: Step 1

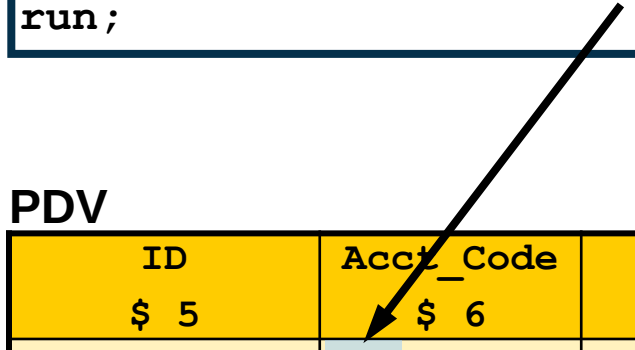
```
data charities;
  length ID $ 5;
  set orion.biz_list;
  if substr(Acct_Code,length(Acct_Code),1)='2';
  ID=substr(Acct_Code,1,length(Acct_Code)-1);
run;
```

PDV

ID	Acct_Code	Name
$ 5	$ 6	$ 30
	AQI2	AQUAMISSIONS INTERNATIONAL

Execution: Step 1

```
data charities;
  length ID $ 5;
  set orion.biz_list;
  if substr(Acct_Code,length(Acct_Code),1)='2';
  ID=substr(Acct_Code,1,length(Acct_Code)-1);
run;
```



PDV

ID	Acct_Code	Name
$ 5	$ 6	$ 30
	AQI2	AQUAMISSIONS INTERNATIONAL

Execution: Step 1

```
data charities;
  length ID $ 5;
  set orion.biz_list;
  if substr(Acct_Code,length(Acct_Code),1)='2';
  ID=substr(Acct_Code,1,length(Acct_Code)-1);
run;
```

PDV

ID $ 5	Acct_Code $ 6	Name $ 30
AQI	AQI2	AQUAMISSIONS INTERNATIONAL

Execution: Step 1

```
data charities;
  length ID $ 5;
  set orion.biz_list;
  if substr(Acct_Code,length(Acct_Code),1)='2';
  ID=substr(Acct_Code,1,length(Acct_Code)-1);
run;
```

Implicit OUTPUT;
Implicit RETURN;

PDV

ID $ 5	Acct_Code $ 6	Name $ 30
AQI	AQI2	AQUAMISSIONS INTERNATIONAL

Create the List of Charities – Step 1 Complete

Listing of **charities**

ID	Acct_ Code	Name
AQI	AQI2	AQUAMISSIONS INTERNATIONAL
CCI	CCI2	CANCER CURES, INC.
CNI	CNI2	CONSERVE NATURE, INC.
CS	CS2	CHILD SURVIVORS
CU	CU2	CUIDADORES LTD.
DAI	DAI2	DISASTER ASSIST, INC.
ES	ES2	EARTHSALVORS
FFC	FFC2	FARMING FOR COMMUNITIES
MI	MI2	MITLEID INTERNATIONAL
SBA	SBA2	SAVE THE BABY ANIMALS
V2	V22	VOX VICTIMAS
YYCR	YYCR2	YES, YOU CAN RECYCLE

Step 2 is to transform the values in **Name** to a mix of uppercase and lowercase.

The PROPCASE Function

The PROPCASE function converts all words in an argument to *proper case*, in which the first letter is uppercase and the remaining letters are lowercase.

General form for the PROPCASE function:

NewVar=PROPCASE(*argument <,delimiter(s)>*);

argument	can be a character constant, variable, or expression.
delimiter(s)	delimiters are characters which separate words. If omitted, the default delimiters are the blank, /, - , (, ., and tab characters.
NewVar	If *NewVar* is a new variable, it is created with the same length as *argument*.

The PROPCASE Function

Example:

```
Name = 'SURF&LINK SPORTS';
Pname = propcase(Name);
Pname2 = propcase(Name,' &');
```

PDV

Name $ 16	Pname $ 16
SURF&LINK SPORTS	Surf&link Sports

Pname2 $ 16
Surf&Link Sports

Quiz

This PDV shows the current value of **Name**:

Name
HEATH*BARR*LITTLE EQUIPMENT SALES

Write an assignment statement that converts the value of **Name** to this:

Name
Heath*Barr*Little Equipment Sales

Quiz – Correct Answer

This PDV shows the current value of **Name**:

Name
HEATH*BARR*LITTLE EQUIPMENT SALES

Write an assignment statement that will convert the value of **Name** to this:

Name
Heath*Barr*Little Equipment Sales

```
Name = propcase(Name,' *');
```

The second argument to the PROPCASE function must list all the characters to use as delimiters. In this example, the space and * both need to be listed.

Create the List of Charities – Step 2

Adding an assignment statement to convert **Name** to proper case completes the **charities** data set.

```
data charities;
  length ID $ 5;
  set orion.biz_list;
  if substr(Acct_Code,length(Acct_Code),1)='2';
  ID=substr(Acct_Code,1,length(Acct_Code)-1);
  Name = propcase(Name);
run;
```

Create the List of Charities – Complete

Listing of **charities**

ID	Acct_ Code	Name
AQI	AQI2	Aquamissions International
CCI	CCI2	Cancer Cures, Inc.
CNI	CNI2	Conserve Nature, Inc.
CS	CS2	Child Survivors
CU	CU2	Cuidadores Ltd.
DAI	DAI2	Disaster Assist, Inc.
ES	ES2	Earthsalvors
FFC	FFC2	Farming For Communities
MI	MI2	Mitleid International
SBA	SBA2	Save The Baby Animals
V2	V22	Vox Victimas
YYCR	YYCR2	Yes, You Can Recycle

Other Useful Character Functions

Function	Purpose
RIGHT(*string*)	right-aligns a character expression.
LEFT(*string*)	left-aligns a character expression.
UPCASE(*string*)	converts all letters in an argument to uppercase.
LOWCASE(*string*)	converts all letters in an argument to lowercase.
CHAR(*string*,*position*)	returns a single character from a specified *position* in a character *string*.

Quiz

Find the syntax error in the code below. Product_Name has a length of 45.

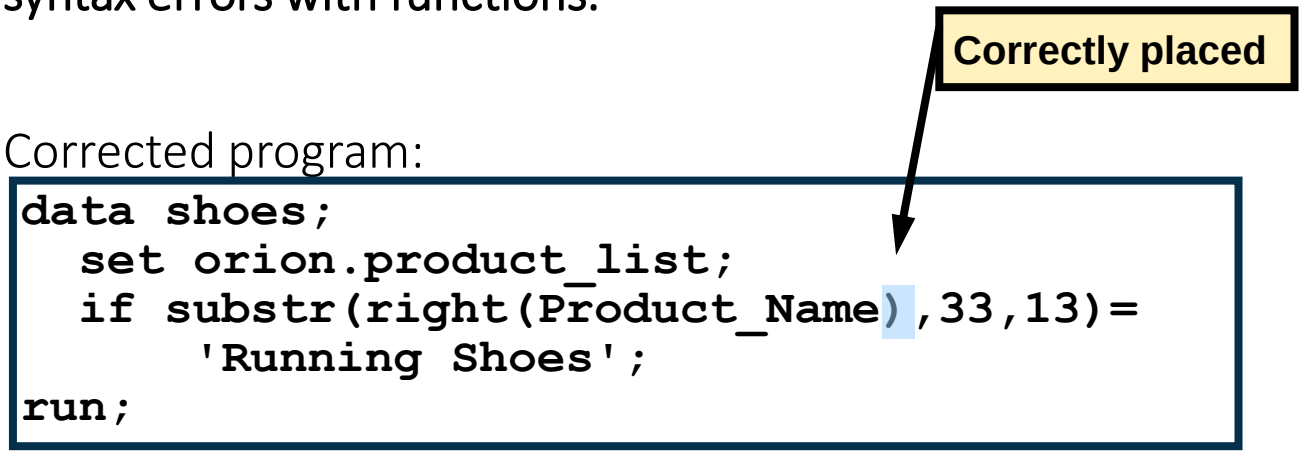
Partial listing of product_list:

Product_ID	Product_Name	Supplier_ID	Product_Level	Product_Ref_ID
220100700023	Armadillo Road Dmx Men's Running Shoes	16733	1	220100700000
220100700024	Armadillo Road Dmx Women's Running Shoes	16733	1	220100700000
220100700046	Tcp 6 Men's Running Shoes	16733	1	220100700000

```
data shoes;
   set orion.product_list;
   if substr(right(Product_Name,33,13))=
      'Running Shoes';
run;
```

Quiz – Correct Answer

Misplaced parentheses are some of the most common syntax errors with functions.



Corrected program:

```
data shoes;
  set orion.product_list;
  if substr(right(Product_Name),33,13)=
      'Running Shoes';
run;
```