

## 2. Creating DataSets and Filtering Data

Daniel Alconchel Vázquez

**Exercise 1.** Notice, that we cannot create this data set properly (Straszliwy is too long). Fix it (use LENGTH).

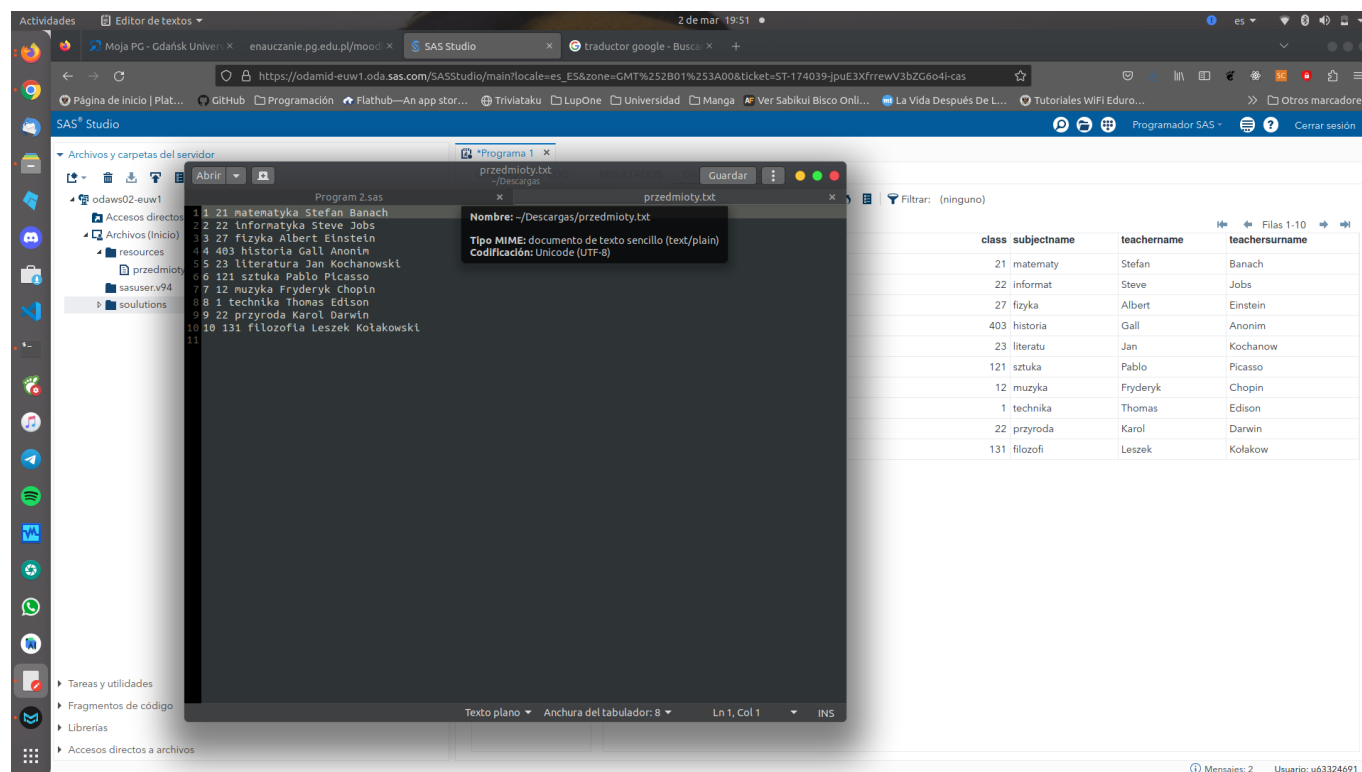
If we run the next code:

```
LIBNAME mylib "/home/u63324691/soulutions";

Filename plik "/home/u63324691/resources/przedmioty.txt";

data mylib.subjects;
    infile plik;
    input studentid class subjectname$ teachername$ teachersurname$;
run;
```

We will see that some datas are "cut or not complete", so we have to change the length of variables.



The screenshot shows the SAS Studio interface. On the left, a file explorer shows the project structure with folders like 'resources' and 'solutions'. The main window displays a SAS program named 'Program 2.sas' with the following code:

```
1 1 21 matenatyka Stefan Banach
2 2 22 informatyka Steve Jobs
3 3 27 fizyka Albert Einstein
4 4 489 historia Gall Anonin
5 5 23 literatura Jan Kochanowski
6 6 121 sztuka Pablo Picasso
7 7 12 muzyka Fryderyk Chopin
8 8 1 technika Thomas Edison
9 9 22 przyroda Karol Darwin
10 10 131 filozofia Leszek Kolakowski
11
```

A tooltip for the file 'przedmioty.txt' is visible, showing its MIME type as 'documento de texto sencillo (text/plain)' and encoding as 'Unicode (UTF-8)'. On the right, the output window shows a table with the following data:

class	subjectname	teachername	teachersurname
21	matematyka	Stefan	Banach
22	informatyka	Steve	Jobs
27	fizyka	Albert	Einstein
403	historia	Gall	Anonim
23	literatura	Jan	Kochanowski
121	sztuka	Pablo	Picasso
12	muzyka	Fryderyk	Chopin
1	technika	Thomas	Edison
22	przyroda	Karol	Darwin
131	filozofia	Leszek	Kolakowski

We can fix this by adding the line:

```
data mylib.subjects;
  infile plik;
  Length subjectname$20. teachersurname$20.;
  input studentid class subjectname$ teachersurname$;
run;
```

The screenshot shows the SAS Studio web interface. On the left, a file explorer shows a project named 'Programa 1' with a file 'przedmioty.txt'. The main window displays the SAS program code in a dark-themed editor. The output window on the right shows the results of the program, which is a table with 10 rows and 5 columns: subjectname, teachersurname, studentid, class, and teachersurname. The data is as follows:

subjectname	teachersurname	studentid	class	teachersurname
matematyka	Banach	1	21	Stefan
informatyka	Jobs	2	22	Steve
fizyka	Einstein	3	27	Albert
historia	Anonim	4	403	Gall
literatura	Kochanowski	5	23	Jan
sztuka	Picasso	6	121	Pablo
muzyka	Chopin	7	12	Fryderyk
technika	Edison	8	1	Thomas
przyroda	Darwin	9	22	Karol
filozofia	Kolakowski	10	131	Leszek

Also, as it is specified in the laboratory document we can use this other method for fixing it:

```
data mylib.subjects;
  infile plik;
  input studentid class subjectname: $20. techarname$ teachersurname :$20.;
run;

/*or*/
```

```
data mylib.subjects;
  informat subjectname $10. teachersurname $15.;
  infile plik;
  input studentid class subjectname$ techarname$teachersurname$;
run;
```

**Exercise 2.** Clicking' choose all Land Rovers and Jeeps from table sashelp.cars. Read and analyse generated code.

We can do this by, in the data table, clicking in the filter option and using the sql notation (without indicate the property WHERE) the option. In our case `MAKE="Jeep"` or `MAKE=Land Rover`. Another option is to use right click in the attribute on the table, then go to filter and select by hand the values.

SAS Studio interface showing the SASHELP.CARS dataset. The filter applied is: (Make="Jeep" OR Make="Land Rover"). The table displays 6 rows of data.

Make	Model	Type	Origin	DriveTrain	MSRP	Invoice	EngineSize	Cylinders
1 Jeep	Grand Cherokee Laredo	SUV	USA	Front	\$27,905	\$25,686	4	6
2 Jeep	Liberty Sport	SUV	USA	All	\$20,130	\$18,973	2.4	4
3 Jeep	Wrangler Sahara convertible 2dr	SUV	USA	All	\$25,520	\$23,275	4	6
4 Land Rover	Range Rover HSE	SUV	Europe	All	\$72,250	\$65,807	4.4	8
5 Land Rover	Discovery SE	SUV	Europe	All	\$39,250	\$35,777	4.6	8
6 Land Rover	Freelander SE	SUV	Europe	All	\$25,995	\$23,969	2.5	6

The generated code is:

```
PROC SQL;
CREATE TABLE WORK.query AS
SELECT Make , Model , 'Type'n , Origin , DriveTrain , MSRP , Invoice ,
EngineSize , Cylinders , Horsepower , MPG_City , MPG_Highway , Weight ,
Wheelbase , 'Length'n FROM SASHELP.CARS WHERE (Make="Jeep" OR Make="Land
Rover");
RUN;
QUIT;

PROC DATASETS NOLIST NODETAILS;
CONTENTS DATA=WORK.query OUT=WORK.details;
RUN;

PROC PRINT DATA=WORK.details;
RUN;
```

**Exercise 3.** Choose the following observations from *sashelp.class*.

1. where age > 13.

2. where  $12 \leq \text{age} \leq 14$ .

```
data observations1;  
  set sashelp.class;  
  where age>13;  
run;
```

```
data observations1;  
  set sashelp.class;  
  where age>11 and age<15;  
run;
```

Nº total de filas: 9 Nº total de columnas: 5

	Name	Sex	Age	Height	Weight
1	Alfred	M	14	69	112.5
2	Carol	F	14	62.8	102.5
3	Henry	M	14	63.5	102.5
4	Janet	F	15	62.5	112.5
5	Judy	F	14	64.3	90
6	Mary	F	15	66.5	112
7	Philip	M	16	72	150
8	Ronald	M	15	67	133
9	William	M	15	66.5	112

Nº total de filas: 12 Nº total de columnas: 5

	Name	Sex	Age	
1	Alfred	M	14	
2	Alice	F	13	
3	Barbara	F	13	
4	Carol	F	14	
5	Henry	M	14	
6	James	M	12	
7	Jane	F	12	
8	Jeffrey	M	13	
9	John	M	12	
10	Judy	F	14	
11	Louise	F	12	
12	Robert	M	12	

**Exercise 4.** Using sashelp.shoes create:

- data set *Women*, which contains products for women,
- data set *WomenDresses*, which contain dresses for women,
- data set *Casual*, which contains casual products.

```
data women;  
  set sashelp.shoes;  
  where product like "%Women%";  
run;
```

SAS Studio interface showing the output of a SAS query. The query is: `data womandresses; set sashelp.shoes; where product like "%Women%Dress%"; run;`

The output table, titled "WORK.WOMEN", displays 24 rows of data. The columns are: Region, Product, Subsidiary, Stores, Sales, Inventory, and Returns. The data is filtered to show only products containing "Women" and "Dress".

Region	Product	Subsidiary	Stores	Sales	Inventory	Returns
1 Africa	Women's Casual	Addis Ababa	2	\$51,541	\$98,641	\$94
2 Africa	Women's Dress	Addis Ababa	12	\$108,942	\$311,017	\$3.23
3 Africa	Women's Dress	Algiers	12	\$90,648	\$266,805	\$2.69
4 Africa	Women's Casual	Cairo	14	\$328,474	\$940,851	\$10.12
5 Africa	Women's Dress	Cairo	3	\$14,095	\$51,145	\$74
6 Africa	Women's Dress	Johannesburg	4	\$42,682	\$120,127	\$96
7 Africa	Women's Casual	Khartoum	1	\$19,582	\$30,727	\$38
8 Africa	Women's Dress	Khartoum	6	\$48,031	\$132,679	\$1.37
9 Africa	Women's Casual	Kinshasa	1	\$17,919	\$21,363	\$40
10 Africa	Women's Dress	Kinshasa	3	\$32,928	\$87,149	\$71
11 Africa	Women's Dress	Luanda	1	\$8,467	\$47,387	\$21
12 Africa	Women's Dress	Nairobi	3	\$28,515	\$62,740	\$67
13 Asia	Women's Casual	Bangkok	1	\$5,389	\$16,251	\$18
14 Asia	Women's Casual	Seoul	2	\$20,448	\$36,576	\$79
15 Asia	Women's Dress	Seoul	7	\$78,234	\$140,628	\$1.89
16 Canada	Women's Dress	Calgary	2	\$12,601	\$54,677	\$48
17 Canada	Women's Casual	Montreal	2	\$24,497	\$34,808	\$83
18 Canada	Women's Dress	Montreal	12	\$132,638	\$483,637	\$2.88
19 Canada	Women's Casual	Ottawa	1	\$18,712	\$46,064	\$42
20 Canada	Women's Dress	Ottawa	4	\$33,824	\$84,378	\$93
21 Canada	Women's Casual	Toronto	4	\$63,492	\$149,239	\$1.70
22 Canada	Women's Dress	Toronto	7	\$53,940	\$165,711	\$1.96
23 Canada	Women's Casual	Vancouver	20	\$304,106	\$665,323	\$8.27
24 Canada	Women's Dress	Vancouver	21	\$756,347	\$2,503,387	\$19.37

```
data womandresses;  
set sashelp.shoes;  
where product like "%Women%Dress%";  
run;
```

SAS Studio interface showing the output of a SAS query. The query is: `data womandresses; set sashelp.shoes; where product like "%Women%Dress%"; run;`

The output table, titled "WORK.WOMANDRESSES", displays 24 rows of data. The columns are: Region, Product, Subsidiary, Stores, Sales, Inventory, and Returns. The data is filtered to show only products containing "Women" and "Dress".

Region	Product	Subsidiary	Stores	Sales	Inventory	Returns
1 Africa	Women's Dress	Addis Ababa	12	\$108,942	\$311,017	\$3.23
2 Africa	Women's Dress	Algiers	12	\$90,648	\$266,805	\$2.69
3 Africa	Women's Dress	Cairo	3	\$14,095	\$51,145	\$74
4 Africa	Women's Dress	Johannesburg	4	\$42,682	\$120,127	\$96
5 Africa	Women's Dress	Khartoum	6	\$48,031	\$132,679	\$1.37
6 Africa	Women's Dress	Kinshasa	3	\$32,928	\$87,149	\$71
7 Africa	Women's Dress	Luanda	1	\$8,467	\$47,387	\$21
8 Africa	Women's Dress	Nairobi	3	\$28,515	\$62,740	\$67
9 Asia	Women's Dress	Seoul	7	\$78,234	\$140,628	\$1.89
10 Canada	Women's Dress	Calgary	2	\$12,601	\$54,677	\$48
11 Canada	Women's Dress	Montreal	12	\$132,638	\$483,637	\$2.88
12 Canada	Women's Dress	Ottawa	4	\$33,824	\$84,378	\$93
13 Canada	Women's Dress	Toronto	7	\$53,940	\$165,711	\$1.96
14 Canada	Women's Dress	Vancouver	21	\$756,347	\$2,503,387	\$19.37
15 Central America/Caribbean	Women's Dress	Kingston	41	\$375,817	\$1,147,300	\$11.65
16 Central America/Caribbean	Women's Dress	Managua	10	\$63,238	\$163,951	\$2.30
17 Central America/Caribbean	Women's Dress	Mexico City	7	\$69,553	\$257,437	\$1.61
18 Central America/Caribbean	Women's Dress	San Juan	16	\$109,110	\$336,513	\$3.77
19 Eastern Europe	Women's Dress	Budapest	6	\$52,022	\$144,360	\$1.45
20 Eastern Europe	Women's Dress	Moscow	10	\$72,671	\$196,973	\$2.22
21 Eastern Europe	Women's Dress	Prague	14	\$128,497	\$425,237	\$3.43
22 Eastern Europe	Women's Dress	Warsaw	19	\$108,936	\$425,390	\$4.46
23 Middle East	Women's Dress	Al-Khobar	11	\$241,820	\$611,686	\$6.16
24 Middle East	Women's Dress	Dubai	19	\$435,891	\$1,292,400	\$12.99

```
data casual;
  set sashelp.shoes;
  where product like "%Casual%";
run;
```

The screenshot shows the SAS Studio web interface. The main window displays the results of a SAS query. The query was executed on the WORK.CASUAL table, selecting all columns. The results table shows 24 rows of data. The columns are: Region, Product, Subsidiary, Stores, Sales, Inventory, and Returns. The data is filtered to show only rows where the product is 'Casual'.

	Region	Product	Subsidiary	Stores	Sales	Inventory	Returns
1	Africa	Men's Casual	Addis Ababa	4	\$67,242	\$118,036	\$2,28
2	Africa	Women's Casual	Addis Ababa	2	\$51,541	\$98,641	\$94
3	Africa	Men's Casual	Algiers	4	\$63,206	\$100,982	\$2,22
4	Africa	Men's Casual	Cairo	25	\$360,209	\$1,063,251	\$9,42
5	Africa	Women's Casual	Cairo	14	\$328,474	\$940,851	\$10,12
6	Africa	Men's Casual	Khartoum	1	\$9,244	\$16,230	\$47
7	Africa	Women's Casual	Khartoum	1	\$19,582	\$30,727	\$38
8	Africa	Women's Casual	Kinshasa	1	\$17,919	\$21,363	\$40
9	Africa	Men's Casual	Luanda	3	\$62,893	\$149,864	\$1,70
10	Asia	Women's Casual	Bangkok	1	\$5,389	\$16,251	\$18
11	Asia	Men's Casual	Seoul	1	\$11,754	\$2,176	\$83
12	Asia	Women's Casual	Seoul	2	\$20,448	\$36,576	\$79
13	Canada	Men's Casual	Montreal	3	\$53,929	\$187,155	\$2,24
14	Canada	Women's Casual	Montreal	2	\$24,497	\$34,808	\$83
15	Canada	Men's Casual	Ottawa	1	\$19,210	\$29,249	\$57
16	Canada	Women's Casual	Ottawa	1	\$18,712	\$46,064	\$42
17	Canada	Men's Casual	Toronto	2	\$15,403	\$20,811	\$1,40
18	Canada	Women's Casual	Toronto	4	\$63,492	\$149,239	\$1,70
19	Canada	Men's Casual	Vancouver	25	\$353,361	\$671,837	\$20,47
20	Canada	Women's Casual	Vancouver	20	\$304,106	\$665,323	\$8,27
21	Central America/Caribbean	Men's Casual	Kingston	28	\$576,112	\$1,159,556	\$20,00
22	Central America/Caribbean	Women's Casual	Kingston	18	\$215,069	\$275,321	\$7,36
23	Central America/Caribbean	Men's Casual	Managua	3	\$69,276	\$79,666	\$1,86
24	Central America/Caribbean	Women's Casual	Managua	3	\$48,567	\$118,849	\$1,31

**Exercise 5.** Using `sashelp.heart` create table *DeathWithoutReason*, which contains observation with missing values in column *DeathCause*.

```
data DeathWithoutReason;
  set sashelp.heart;
  where deathcause is missing;
run;
```

Tabla: WORK.DEATHWITHOUTREASON Ver: Nombres de columna     Filtrar: (ninguno)

Ver: Nombres de columna      |  Filtrar: (ninguno)

Columnas 🕒 N° total de filas: 3218 N° total de columnas: 17 ⏪

Nº total de filas: 3218 Nº total de columnas: 17

<input checked="" type="checkbox"/>	Seleccionar todo		Status	DeathCause	AgeCHDdiag	Sex	AgeAtStart
<input checked="" type="checkbox"/>	 Status	1	Alive			Female	57
<input checked="" type="checkbox"/>	 DeathCause	2	Alive			Female	39
<input checked="" type="checkbox"/>	  AgeCHDdiag	3	Alive			Male	42
<input checked="" type="checkbox"/>	 Sex	4	Alive			Female	58
<input checked="" type="checkbox"/>	  AgeAtStart	5	Alive			Female	36
<input checked="" type="checkbox"/>	  Height	6	Alive			Male	35
<input checked="" type="checkbox"/>	  Weight	7	Alive			Male	39
<input checked="" type="checkbox"/>	  Diastolic	8	Alive		57	Male	33
<input checked="" type="checkbox"/>	  Systolic	9	Alive		55	Male	33
<input checked="" type="checkbox"/>	  MRW	10	Alive		79	Male	57
<input checked="" type="checkbox"/>	  Smoking	11	Alive		66	Male	44
<input checked="" type="checkbox"/>	  AgeAtDeath	12	Alive			Female	37
<input checked="" type="checkbox"/>	  Cholesterol	13	Alive			Male	40
<input checked="" type="checkbox"/>	  Chol_Status	14	Alive			Female	42
		15	Alive			Female	37

Propiedad	Valor	16	Alive			Female	45
Etiqueta		17	Alive			Female	59
Nombre		18	Alive			Female	36
Longitud		19	Alive			Female	50
Tipo		20	Alive			Female	35
Formato		21	Alive			Male	42
Formato de lectura		22	Alive		68	Male	40

	Status	DeathCause	AgeCHDdiag	Sex	AgeAtStart
1	Alive			Female	57
2	Alive			Female	39
3	Alive			Male	42
4	Alive			Female	58
5	Alive			Female	36
6	Alive			Male	35
7	Alive			Male	39
8	Alive		57	Male	33
9	Alive		55	Male	33
10	Alive		79	Male	57
11	Alive		66	Male	44
12	Alive			Female	37
13	Alive			Male	40
14	Alive			Female	42
15	Alive			Female	37
16	Alive			Female	45
17	Alive			Female	59
18	Alive			Female	36
19	Alive			Female	50
20	Alive			Female	35
21	Alive			Male	42
22	Alive		68	Male	40
23	Alive			Female	41
24	Alive		68	Male	40