

# آشنایی با محیط نرم افزار SAS

نویسنده: نرگس سهرابی

SAS مخفف اصطلاح لاتین (Statistical Analysis System)؛ به معنای سیستم تحلیل آماری است که یک زبان سطح بالا می باشد. پس از اجرای برنامه SAS چهار پنجره اصلی برای شما نشان داده می شود که به ترتیب شامل پنجره log, editor, output, explore می باشد. علاوه بر این پنجره ها، پنجره های دیگری نیز هستند که از مسیر tools, view قابل دسترسی می باشند.

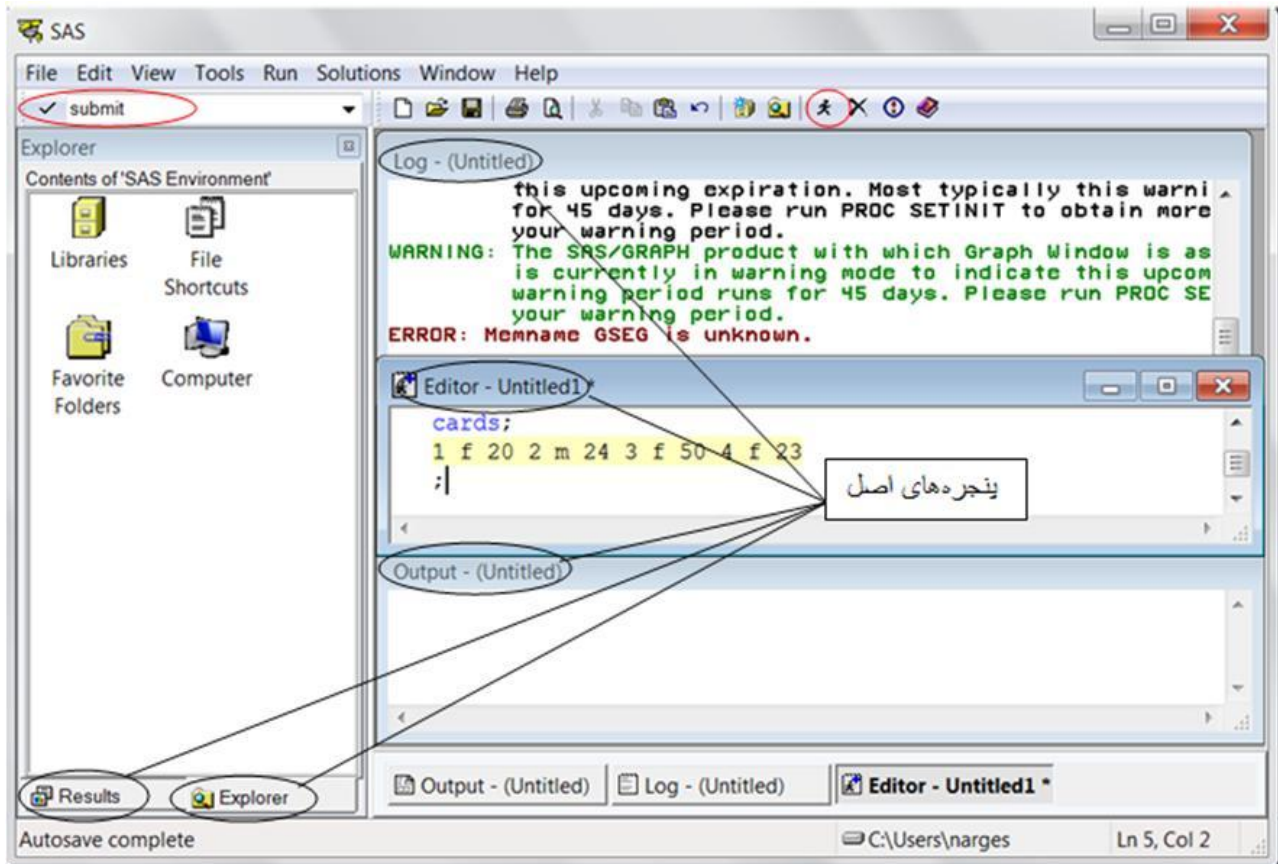
۱. پنجره Editor: پنجره ای است که در آن برنامه نویسی یا یک برنامه SAS را می شود فراخوانی و یا ویرایش کرد. این پنجره در هنگام اجرای برنامه بر روی صفحه اصلی نمایش داده می شود، و از مسیر `tools->text editor` نیز قابل دسترسی است. مانند مثال زیر برنامه را در این پنجره فراخوانی کرده یا می نویسیم و سپس با استفاده از کلید `f8`، یا نوشتن `submit` در نوار دستور یا کلیک به روی آیکن آدمک بر روی نوار ابزار برنامه را اجرا می کنیم. برای اجرای بخشی از برنامه فقط آن قسمت را انتخاب می کنیم و سپس از سه روش بالا اجرا می کنیم. بعد از اجرای این برنامه می توانید برنامه را با پسوند `.sas` ذخیره کنید. هر پوشه با این پسوند شامل یک برنامه SAS می باشد.

۲. پنجره log: در هنگام اجرای هر برنامه می توانید پیام های مربوط به اجرای هر خط برنامه را در این پنجره مشاهده کنید. پیام هایی که در این پنجره می توان مشاهده کرد شامل اجرای درست هر خط برنامه، خطاهای ساختاری یا نوشتاری، خطاهای منطقی و همچنین مدت زمان اجرای برنامه میشود. هر کدام از این پیام ها با رنگ های متفاوتی از هم تفکیک داده شده اند. پیام های معمولی (note) با رنگ آبی خطاهای نوشتاری (error) با رنگ قرمز و خطاهایی منطقی (warning) با رنگ سبز نشان داده شده اند. در صورتی که اجرای برنامه شما به مشکلی برخورد می کند می توانید در این پنجره مشکل برنامه را بررسی کنید و سپس برنامه را تصحیح کنید. با اجرای هر برنامه صفحه `log` شما طولانی و باعث سردرگمی میشود می توانید از دستور `Ctrl+E` برای پاک کردن صفحه `log` استفاده کنید. خروجی این پنجره با پسوند `log *` ذخیره می شود. این پنجره از مسیر `log->view` نیز قابل دسترسی است.

۳. پنجره output: اجرای بعضی از برنامه ها خروجی هایی را تولید میکنند که در پنجره `output` نمایش داده می شوند. خروجی برنامه را می توانید با پسوند `.lst *` ذخیره کنید.

۴. پنجره Explorer و results: پنجره Explorer همان طور که از اسمش پیداست پرونده ها و پوشه ها را جستجو می کند. این پنجره چهار تا پوشه دارد. `libraries, file shortcuts, favorite folder, my computer`. انتخاب `my computer` می توانید فایل های با پسوند های `log *`, `.lst *`, `.sas *` را فراخوانی و اجرا کنید. پوشه `libraries` محل ذخیره موقت خروجی برنامه ها است به طوریکه بعد از بستن نرم افزار تمام اطلاعات پاک میشود. مسیر ذخیره خروجی برنامه `libraries->work` است.

۵. علاوه بر آن، فایل برنامه و داده های مربوط به SAS در این پنجره قابل دستیابی است. پنجره results نتایج خروجی را برای دسترسی راحتتر به صورت دسته بندی نمایش می دهد.



قبل از ورود داده کمی در مورد اصطلاحات مربوط به برنامه نویسی در SAS بیان میشود.

در SAS دو دستور کلی دارد:

- (a) data step دستوری که میتوان در آن داده ها را وارد و مدیریت کرد
- (b) proc step مخفف procedure است و بعد از این دستور میتوان روش های آماری را در این دستور اجرا کرد.

هر جمله ای که به ( ; ) ختم شود را یک دستور (statement) میگویند.

### ایجاد مجموعه داده در sas

برای وارد کردن داده ها در SAS از دستور data با نام دلخواهی برای مجموعه داده باید شروع کرد. دستور (input) (statement) متغیرها را تعریف میکند. متغیرهایی که به صورت کاراکتر نشان داده میشوند را با علامت \$ نشان میدهیم. مثلا اگر متغیر ما نام

کشور باشد در اینصورت هر اسمی طول متفاوتی دارد و یا بینشان فاصله دارد، برای وارد کردن این داده طول بیشترین اسم را در جلوی متغیر مینویسیم. دستور بعدی `datalines` یا `cards` (هر دو دستور را میتوان به کاربرد) است. این دستور نشان می دهد مرحله `data step` کامل شده است و خط بعدی شامل دادهها میشود. میتوان (به دلخواه) بعد از ورود داده دستور `proc print` را نوشت. این دستور دادهها را در `output` نشان میدهد. در صورت نوشتن این دستور، جدول دادهها در مسیر

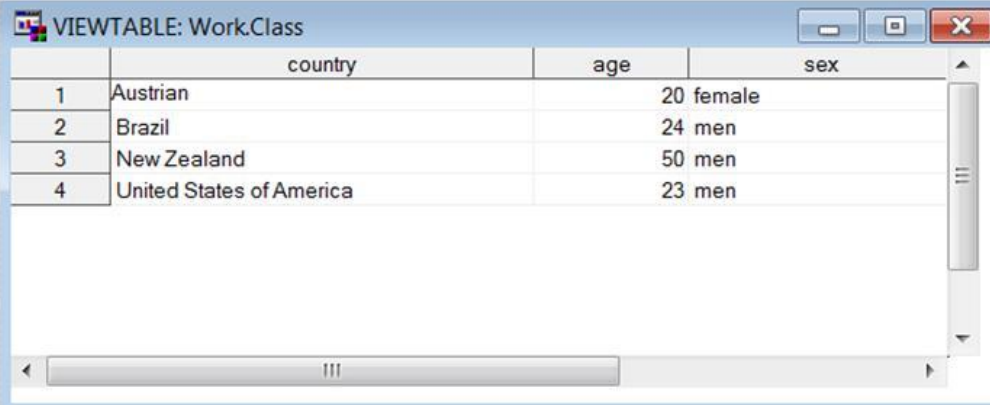
libraries-> work

به صورت موقت ذخیره میشود و برای دیدن جدول داده باید به مسیر بالا رفت.

جدول موقت دادهها که در مسیر

->class libraries->work (نام مجموعه داده)

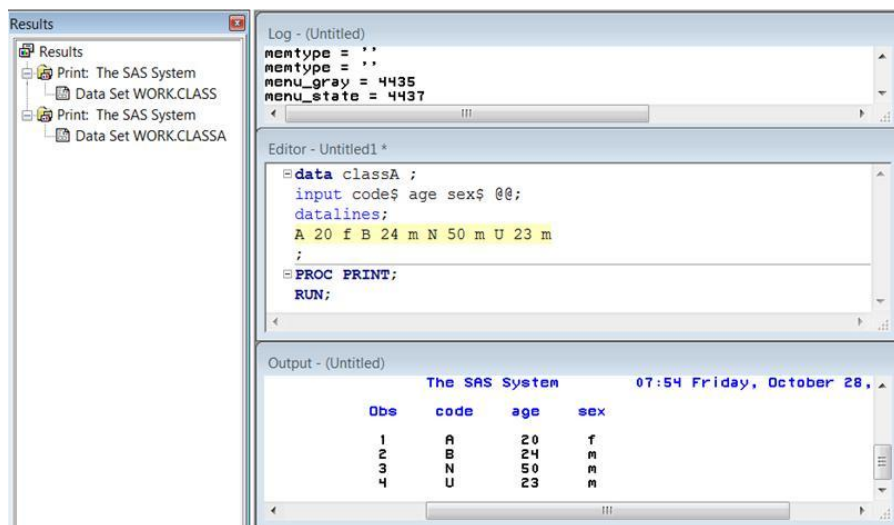
ذخیره شده است.



	country	age	sex
1	Austrian	20	female
2	Brazil	24	men
3	New Zealand	50	men
4	United States of America	23	men

توجه کنید که در جدول بالا متغیرهایی که کاراکتری هستند در سمت چپ نوشته شدهاند اگر به جای حروف از عدد برای کشور استفاده میکردیم به علت علامت \$ عددها هم در چپ نمایش داده میشدند. ولی متغیرهایی که بدون علامت \$ تعریف میشوند در طرف راست نوشته میشوند که نشانه محاسباتی بودن هست. در اینجا ما نشان ندادیم ولی شما می توانید امتحان کنید.

در مثال بالا دادهها به صورت ستونی وارد شده است. میتوان دادهها را به صورت پشت سر هم در سطر نیز وارد کرد فقط کافی است در پایان دستور `input` علامت @@ را بنویسیم.



## خواندن داده‌ها از یک فایل متنی

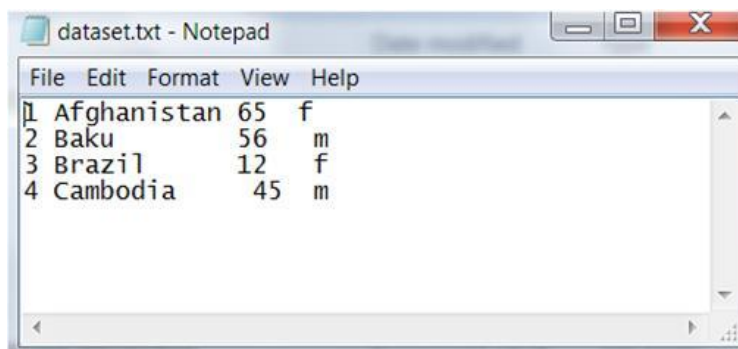
برای خواندن داده‌ها از فایل متن قبل از دستور `input` دستور

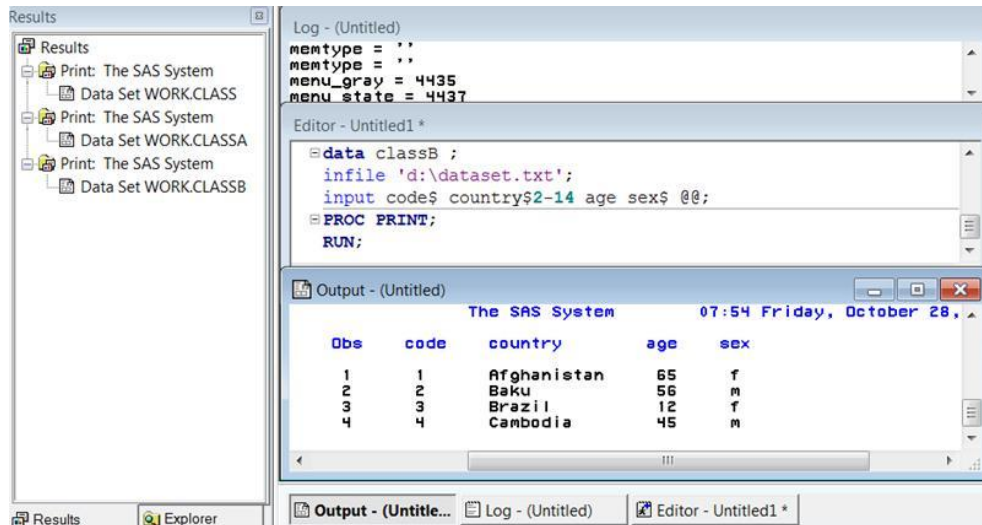
```
infile 'file path(c;\data) ' ;
```

را به کار می‌بریم. برای استفاده از دستور `infile` از دو روش می‌توان استفاده کرد.

در روش اول `infile 'file path(c;\data) ' ;` را قبل از دستور `input` قرار می‌دهیم.

داده‌های موجود در فایل `txt`.





در روش دوم ابتدا از دستور `filename` استفاده می کنیم. این دستور یک فایل میانبر را با اسمی که نوشتیم در برنامه SAS ایجاد می کند. دقت کنید که این دستور در خارج از دستور `DATA step` نوشته میشود.

```
Filename toxic (نام دلخواه) 'D:\Research\Toxin\Data1998\DON.txt';
Data readasc;
  Infile toxic;
  Input location truck probe DON;
run;
```

در اینجا دستور `infile` که در `DATA step` قرار دارد. برای فراخوانی داده‌ها به فایل میانبری که در SAS درست شده است ارجاع داده میشود. در بعضی از فایل‌ها ستون‌های داده با فاصله از هم جدا نمیشوند (برای مثال فایل `*.prn` داده‌ها با ، (کاما) از هم جدا میشوند) بنابراین می توان گزینه `delimiter` برای مشخص کردن نوع جدا کننده‌ها در دستور `input` به کاربرد. همچنین بعضی از فایل داده‌ها دارای `header` (نام متغیرها در سطر اول ) هستند در صورتی که نخواهیم سطر اول خوانده شود از گزینه `firstobs` استفاده میکنیم. و برای فراخوانی کردن تعداد محدودی از سطرها از دستور `obs` استفاده میکنیم.

```
Filename toxic 'D:\Research\Toxin\Data1998\DON.txt' ;
data readasc;
  infile toxic delimiter=', ' firstobs=12 obs=100 ;
  input location truck probe DON ;
run;
```