第一章 sas是什么

1.SAS系统是一个模块化的集成软件系统；  
 —— 数据处理和统计领域的国际标准软件；  
        —— 世界领先的数据分析和信息系统；  
 SAS系统广泛应用于金融、医疗、运输、通迅、政府、科研和教育等领域；  
         
SAS含义  
        Statistical  Analysis  System

2.SAS系统的主要四大功能   
       数据访问  
       数据管理  
       数据分析  
       数据呈现

3.SAS系统对50多种数据源提供了引擎 ，如： DB2 和 Oracle

－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－

第二章 开始sas程序的讲解

1.sas程序的介绍  
 有两种程序步组成，数据步和过程步，每个步通常有若干个SAS语句组成；

        数据步：以data语句开始，用于创建和处理SAS数据集；  
        过程步：以proc语句开始，主要用户处理SAS数据集；

2.SAS数据集  
   通常分为两个部分：描述部分（包含数据属性的信息）和 数据部分（包含数值）；

                数据集的列称为变量(Variable),行称为观测(Observation)。

 查看数据集的描述部分：proc contents data=sas\_data\_set;run;  
        查看数据集的数据部分：proc print    data=sas\_data\_set;run;

4.SAS变量的类型   
     \*字符型变量 （Character Variable ）(1-32767字节),均以字母、下划线开头；字符型变量的缺省数据用空格表示；   
     \*数值型变量 （Numerical Variable ）默认为8个字节的长度，数值型变量的缺省数据用点（.）表示；

5.变量的命名规范：1－32个字符长度，不区分大小写，以下划线或字母开头

－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－

第三章  sas数据仓库

1.每次SAS启动都自动生成三个库标记：WORK、SASUSER和SASHELP；

2.库的分类  
     永久性库：sasuser、sashelp、自定义的库

     临时性库：只有一个，名为WORK，可以省略库标记；每次启动SAS自动生成，结束SAS后库中的数据被自动删除；

 用libname指定库标记，如：libname temp“e:\temp\data”；

3.使用关键词\_ALL\_列出数据仓库中所有的sas文件，使用NODS option来禁止对数据集的描述  
 PROC CONTENTS DATA=libref.\_ALL\_  NODS;  
 RUN;

 注意：NODS选项只能和\_ALL\_一起联用  
－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－

第四章 数据列表报表

1.print过程  
  语法格式：  
 proc print data=SAS数据集 noobs；  
  var 分析变量1 分析变量2 ... 分析变量n；  
  where 表达式；  
  sum 求和变量；    
 run;

     Noobs选项：在PRINT过程中可以用NOOBS选项去掉OBS列；  
                 
     VAR语句：控制变量的出现与否以及出现的顺序；

     WHERE语句：控制哪些观测将出现在报表中；它的表达式主要是操作数和操作符，

     SUM语句：计算变量的总合；

2.观测的排序和分组§(sort)和(by)

  对数据进行分组并求每组小计，用PRINT过程的BY语句，但必须先对相应的变量进行排序；  
  如：proc sort data=temp.empdata out=temp.empdata2;  
         By JobCode;  
      Run;

      proc print data=temp.empdata;  
      by JobCode;  
      sum Salary;  
      pageby JobCode; /\*使产生的报表按组分页\*/  
      run;

－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－

第五章：输出  
1.标题和脚注：  
  在所有的SAS报告中都可以加标题（Title）和脚注(Footnote):  
     
  语法格式：TITLEn ‘text’;  
            FOOTNOTEn ‘text’;

     特点：n 的取值范围是1-10；  
           标题出现在每页的顶部；  
           脚注出现在每页的底部；  
           如果没有定义标题，缺省的标题是：“The SAS System”;  
           如果没有脚注就不出现；  
           没有n的标题和脚注就是：TITLE1、FOOTNOTE1；  
           定义的标题和脚注一直有效，知道另一个语句被执行；  
           带n的标题或脚注被执行后，替代了原先具有同样号码的标题和脚注；  
           带n的标题或脚注被执行后，取消了更大号码的标题和脚注；  
        
2.LABEL语句：产生用户化和容易阅读的表头：  
      如：label 变量1=’标签’  变量2=’标签’;

      属性：是最大长度为256个字符串；  
             
      注意：在PRINT过程中必须用PRINT语句中的LABEL或SPLIT=选项才能被显示；  
            在过程步中定义只在该过程中有效；  
            在数据步中定义就被存在数据集的描述部分与数据集一直有效；

3.format的使用  
 分类：系统format和用户自定义format

4.用户自定义format的使用  
  format变量的语法格式：<$>format<w>.<d>

  在VALUE语句中，格式可以赋予为：  
 A．单个数字：如：  
             Proc format;  
               Value gender 1=’Female’  
                          2=’Male’  
                          Other=’Miscoded’;  
             Run;

 B．某数字范围：如：  
             Proc format;  
                Value boadfmt  low-49=’Below’  
                              50-99=’Average’  
                              100-high=’Above Average’;  
             Run;

 C．字符或字符串：如：  
             Proc format;  
                Value  $grade  ‘A’=’GOOD’  
                              ‘B’-‘D’=’PAID’  
                              ‘I’,’W’=’POOR’  
                              ‘PILOT’=’pilot’  
                              Other=’Miscoded’;  
             Run;

    format的使用步骤：  
 第一步：用户创建format  
  PROC FORMAT;  
          VALUE format-name range1='label '   
       range2='label '  
       . . . ;  
  RUN;

 第二步：应用所创建的format  
  proc print data=ia.empdata;  
      format [$]varialble-name format-name;  
  run;

5.使用ODS创建html报表（利用ODS将SAS输出结果生成HTML格式文件）  
 ODS--Output Delivery System

 语法格式：ODS HTML FILE='HTML-file-specification' <options>;  
   产生输出的sas代码  
    ODS HTML CLOSE;

第六章 创建sas数据集  
1.列输入(column input)  
 \*此模式读入外部原始数据文件，适应文件为：  
         数据固定在某些列中；     
         数据只包含标准的数字和字符；

 \*过程:  
  a.开始一个数据步，并给数据步命名  
  b.用infile指明原始数据的存放位置  
  c.用input指明怎样读取原始数据

 \*格式：  
     data 库名.数据集名；  
  infile '文件名(路径)' <选项>;  
  input 变量名 <$> 起始列-结束列； ($用在变量是字符型)  
     run;

2.格式输入(formatted input)

 \*适合用格式输入的外部原始数据文件  
  数据是固定列；  
  但含有标准或者不标准字符以及数字的文件；

 \*语法格式：  
   data SAS数据集;  
            Infile ‘外部原始文件’;  
            INPUT 指针控制 变量名 <$> 格式名；($表示字符型变量)  
          Run;

 \*指针的控制：  
   @n 移动指针到第几列（绝对位置）  
   +n 把指针移动几个位置（相对位置）

3.输入格式informat  
 <$>informat-namew.<d>   
 说明:  
  $ 如果是字符型，使用$  
  informat-name是输入格式的格式名  
  w 是变量总长度  
  .  句点是必修的分隔符,不能缺少  
  d  如果是数值型的话, d指定了小数位的长度

4.分配变量属性  
  变量的临时属性和永久属性：  
   PROC步可赋予临时属性：其中的标签只在该步显示时有，并没存在数据集里；  
         如：proc print data=temp.dfwlax  label;  
                        Label Dest=’Destination’  
   FirstClass=’First Class Passengers’;  
                    Run;

   DATA步可赋予永久属性：其中的标签被存在数据的描述部分，与数据集一起存在；  
   如： data temp.dfwlax;  
            Infile ‘‘c:\course\tempdata.dat’;  
            Input  @12 Dest $3.  @15 FirstClass $3. ;  
            Label  Dest=’Destination’  FirstClass=’First Class Passengers’;  
        Run;

---------------------------------------------------------------------------------------

第七章 数据步程序设计  
1.读sas数据集以及创建变量  
   用DATA步产生SAS数据集的三种方法：  
      A.数据在作业流中：  
          DATA 语句；  
             INPUT 语句；  
             CARDS；  
             数据行；  
           ；  
           RUN；

       B.数据在磁盘上：  
           DATA 语句；  
              INFILE 语句；  
              INPUT 语句；  
           RUN；

        C.数据来自其它SAS数据集：  
            DATA 语句；  
               SET / MERGE / UPDATE / MODIFY语句；  
               <DATA步中的其它SAS语句>；  
            RUN；

2.用已有的数据集创建另一个数据集[set的使用]  
 DATA 新的数据集名;  
  SET input-SAS-data-set;  
  <additional SAS statements>  
 RUN;  
   
3.sas操作符和函数的使用  
 语法格式：function-name(argument1,argument2, . . .)  
   
 函数：sum(argument1,argument2, . . .);  
       TODAY();  
       MDY(month,day,year);  
       QTR(SAS-date);  
       MONTH(SAS-date);  
       WEEKDAY(SAS-date);

4.有条件的程序

语法结构：  
简单if语句  
 IF expression THEN statement;  
 ELSE statement;

复杂if语句  
 IF expression THEN DO;  
      executable statements  
 END;  
 ELSE DO;  
      executable statements  
 END;

设置变量长度  
 LENGTH variable(s) $ length;

取数据集子集  
 a.WHERE语句

 b.DELETE语句  
  IF expression THEN DELETE;

 c.子集IF语句  
  IF expression;

使用sas日期常数  
 格式： 'ddMMMyyyy'd   
 例如：(example: '14dec2000'd)   
 说明：'d是必须的，用来把引号里的字符串转换成sas日期

--------------------------------------------------------------------------------------------

第八章 数据拼接

1.使用set连接sas数据集  
  语法格式：  
 DATA SAS-data-set ;  
  SET SAS-data-set1 SAS-data-set2 . . . ;  
  <additional SAS statements>  
 RUN;

  set中变量重命名  
 语法格式：  
 SAS-data-set(RENAME=(old-name-1=new-name-1  
                             old-name-2=new-name-2                                                            .   
                                          .  
                                          .                 
                             old-name-n=new-name-n))；

  交叉sas数据集，使用by语句  
  BY语句：使用BY语句可使生成的数据集按某变量排序，但输入数据集必先按该变量排序过；  
 语法格式：  
 DATA SAS-data-set;  
  SET SAS-data-set1 SAS-data-set2 . . . ;  
  BY BY-variable;  
       <other SAS statements>  
 RUN;

2.MERGE sas数据集(必先排序)

  MERGE语法格式：  
 DATA SAS-data-set;  
  MERGE SAS-data-sets;  
  BY BY-variable(s);  
  <additional SAS statements>  
 RUN;

  IN= 选项  
 格式:SAS-data-set(IN=variable)  
 解释:一个临时的数字类型的变量，其值为0或者1  
  IN选项，当读入多个SAS数据集时，用IN选项可确定本观测来自哪个数据集；  
  variable＝0表示观测不是来自本数据集  
  variable＝1表示观测是来自本数据集

－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－

第九章 制作汇总报表

1.基本的汇总报表(freq、mean)  
  freq报表默认的情况下：  
      分析每一个变量,显示出每一个数据值,计算出数字类型的每列的百分比,指出每一个变量有多少条观测中有缺失值

  用此过程一般有两个目的：   
    1：描述过程：产生频数表和交叉表，可简洁的描述数据；  
    2：统计过程：产生各种统计量(频数、百分比)，分析变量间关系；

  使用：  
   A.单项频数表：PROC FREQ DATA=SAS数据集;  
                 TABLES 变量;  
               RUN;

   B.双向交叉表：PROC FREQ DATA=SAS数据集;  
                 TABLES 行变量\*列变量;  
               RUN;

   C.n向交叉表:PROC FREQ DATA=SAS数据集;  
                 TABLES a\*b\*c\*d;  
               RUN;

 如果要一张三向（或n向）交叉表 ，只要在TABELS语句中用星号将3个（或n个）变量名连接起来。  
 最后一个变量的值形成列，倒数第二变量的值形成行，其余变量的每一水平（或水平的组合）形成一层.

2.MEAN过程  
 对数据集中的数值变量计算简单描述统计量（个数、均值、标准差、最大值、最小值）；  
                     如：proc means data=temp.crew;  
                             Var Salary;  
                             Class JobCode;  
                         Run;

3.REPORT 程序  
 REPORT过程将PRINT、MEANS、和TABULATE过程与DATA步报表写法的特点结合起来成为一个强有力的编写报表的过程；它的一般形式：

        格式：  
 PROC REPORT DATA=SAS数据集 [选项];  
      BREAK BEFORE|AFTER break变量 [/选项];  
               COLUMN 显示变量;  
               DIFINE 处理变量  [/选项];  
               RBREAK AFTER|BEFORE [/选项];  
        Run;

 例子:  
 proc report data = temp.crew nowd headline headskip ;    
   column JobCode Location Salary;      /\*column: 显示的列\*/  
   define Location / order 'Home Base'; /\*order表示Location是排序变量，引号中的是列抬头\*/  
   define JobCode / group width=8  'Job Code'; /\*group说明JobCode是分组变量、width指明变量宽度 \*/  
   define Salary / format = dollar10.;  /\*format: 输出格式\*/  
   rbreak after /summarize dol;         /\*结尾产生一总汇总值，其中 dol是两虚线\*/  
 run;