```sas

ods <destination> <options>;

```

```sas

ods html file='output.html' style=htmlblue;

proc print data=sashelp.class;

run;

ods html close;

```

```sas

ods pdf file='output.pdf';

ods rtf file='output.rtf';

proc print data=sashelp.class;

run;

ods pdf close;

ods rtf close;

```

```sas

ods pdf file='output.pdf' style=journal;

proc print data=sashelp.class;

run;

ods pdf close;

```

```sas

proc print data=mydata;

var var1-var5;

run;

```

```sas

proc report data=mydata nowd;

column region product sales;

define region / group;

define product / group;

define sales / analysis sum;

run;

```

```sas

proc tabulate data=mydata;

class region product;

var sales;

table region\*product, sales\*(sum mean);

run;

```

```sas

proc sgplot data=mydata;

scatter x=var1 y=var2;

run;

```

```sas

proc report data=mydata nowd;

column category variable1 variable2 variable3;

define category / group;

define variable1 / analysis sum 'Variable 1';

define variable2 / analysis mean 'Variable 2';

define variable3 / analysis min 'Variable 3';

run;

```

```sas

/\* Generate reports on patient outcomes \*/

proc lifetest data=HealthcareData;

time survival\_time\*censor(0);

strata treatment;

run;

```

```sas

/\* Join transactional data with product information \*/

proc sql;

create table SalesData as

select t.\*, p.category

from Transactions t

inner join Products p

on t.product\_id = p.product\_id;

quit;

/\* Generate tabular reports \*/

proc report data=SalesData;

column category region sales\_date sales\_amount;

define category / group;

define region / group;

define sales\_date / across;

define sales\_amount / analysis sum;

run;

/\* Create graphical reports \*/

proc sgplot data=SalesData;

vbar sales\_date / response=sales\_amount group=category;

run;

```

```sas

/\* Perform customer segmentation using k-means clustering \*/

proc fastclus data=CustomerData out=Segments maxclusters=5;

var age income purchase\_frequency;

run;

/\* Generate reports on customer segments \*/

proc freq data=Segments;

tables segment;

run;

```

```sas

/\* Build predictive models for campaign response \*/

proc logistic data=CampaignData;

model response(event='1') = campaign\_channel age income / link=logit;

run;

/\* Generate reports on campaign performance \*/

proc means data=CampaignData;

var response\_rate conversion\_rate roi;

run;

```

```sas

/\* Calculate portfolio metrics \*/

proc summary data=PortfolioData;

var return volatility sharpe\_ratio;

run;

/\* Generate reports on portfolio performance \*/

proc means data=PortfolioData;

var return volatility sharpe\_ratio;

run;

```

```sas

/\* Perform survival analysis \*/

proc phreg data=HealthcareData;

model survival\_time\*censor(0) = treatment age gender / dist=exponential;

run;

/\* Generate reports on patient outcomes \*/

proc lifetest data=HealthcareData;

time survival\_time\*censor(0);

strata treatment;

run;

```

```sas

/\* Build anomaly detection models \*/

proc hpforest data=TransactionData;

input transaction\_id fraud\_flag variables;

target fraud\_flag / level=nominal;

run;

/\* Generate reports on fraud detection \*/

proc freq data=TransactionData;

tables fraud\_flag / nocum;

run;

```

```sas

/\* Optimize supply chain network \*/

proc optmodel;

/\* Define optimization model \*/

...

run;

/\* Generate reports on supply chain optimization \*/

proc print data=OptimizationResults;

run;

```

```sas

/\* Perform sentiment analysis \*/

proc textmining data=SocialMediaData;

sentiment sentiment\_score;

run;

/\* Generate reports on sentiment analysis \*/

proc sgplot data=SocialMediaData;

vbar sentiment / response=sentiment\_score;

run;

```