

# SAS GRADED PROJECT

## DOMAIN: INSURANCE

**NAME: PANAGAM MOHITHA**

*/\*1. Import all the 4 files in SAS data environment\*/*

```
FILENAME REFFILE '/home/u61856037/sasuser.v94/Third_Party.csv';
```

```
PROC IMPORT DATAFILE=REFFILE DBMS=CSV OUT=Third_party;
```

```
    GETNAMES=YES;
```

```
RUN;
```

```
PROC CONTENTS DATA=Third_party;
```

```
RUN;
```

```
FILENAME REFFILE '/home/u61856037/sasuser.v94/Online.csv';
```

```
PROC IMPORT DATAFILE=REFFILE DBMS=CSV OUT=ONLINE;
```

```
    GETNAMES=YES;
```

```
RUN;
```

```
PROC CONTENTS DATA=ONLINE;
```

```
RUN;
```

```
FILENAME REFFILE '/home/u61856037/sasuser.v94/Roll_Agent.csv';
```

```
PROC IMPORT DATAFILE=REFFILE DBMS=CSV OUT=Roll_agent;
```

```
        GETNAMES=YES;

RUN;

PROC CONTENTS DATA=Roll_agent;

RUN;

FILENAME REFFILE '/home/u61856037/sasuser.v94/Agent_Score.csv';
```

```
PROC IMPORT DATAFILE=REFFILE DBMS=CSV OUT=Agent_score;

        GETNAMES=YES;

RUN;
```

```
PROC CONTENTS DATA=Agent_score;

RUN;
```

```
/*-----*/
```

```
/*2. Create one dataset from all the 4 dataset?*/
```

```
data Agents_data;

        set project.Roll_agent project.online project.third_party;

run;
```

```
proc contents data=Agent_score varnum;

run;
```

```
proc sort data=Agents_data;
```

```

        by agentid;

run;

proc sort data=Agent_score;

        by AgentID;

run;

proc print data=Agent_score;

run;

data agents_base;

        merge Agents_data(in=a) Agent_score(in=b);

        by AgentID;

        if a;

run;

/*-----*/

/*3. Remove all unwanted ID variables?*/

data Agents_data (drop=hhid proposal_num policy_num);

        set Agents_data;

run;

/*-----*/

/*4. Calculate annual premium for all customers?*/

data agents_base;

```

```

set agents_base;

if payment_mode="Annual" then
    Total_premium=(premium);
else if payment_mode="Semi Annual" then
    Total_premium=(premium*2);
else if payment_mode="Quarterly" then
    Total_premium=(premium*4);
else
    Total_premium=(premium*12);
run;

```

```

/*-----*/

```

***/\*5. Calculate age and tenure as of 31 July 2020 for all customers?\*/***

```

data agents_base;

    set agents_base;

    customer_age=intck('year', dob, '31jul2020'd);

run;

```

```

/*-----*/

```

***/\*6. Create a product name by using both level of product information.And product name should be representable i.e. no code should be present in final product name?\*/***

```

data agents_base;

    set agents_base;

    Extracted_Product_name=substr(product_lvl2, 5);

run;

```

```
data agents_base;
```

```
    set agents_base;
```

```
    Final_product_name=CAT(product_lvl1);
```

```
run;
```

```
/*-----*/
```

***/\*7. After doing clean up in your data, you have to calculate the distribution of customers across product and policy status***

```
proc sql;
```

```
    select policy_status, Final_product_name, count(custid) as no_of_customers
```

```
        from agents_base group by policy_status, Final_product_name order by
```

```
        no_of_customers;
```

```
quit;
```

```
/*-----*/
```

***/\*8. Calculate Average annual premium for different payment mode and interpret the result?\*/***

```
proc sql;
```

```
    select payment_mode, avg(total_premium) as Average_annual_premium from
```

```
        agents_base group by payment_mode order by Average_annual_premium desc;
```

```
quit;
```

```
/*-----*/
```

***/\*9. Calculate Average persistency score, no fraud score and tenure of customers across product and policy status, and interpret the result?\*/***

```
proc sql;
```

```
    select Final_product_name, policy_status, avg(Persistency_Score) as
```

```
        Average_Persistency_Score, avg(Tenure) as Average_Tenure from agents_base group by
```

```
Final_product_name, policy_status order by Average_Persistency_Score desc,  
Average_Tenure desc;
```

```
quit;
```

```
/*-----*/
```

```
/*10. Calculate Average age of customer across acquisition channel and policy status, and interpret  
the result?*/
```

```
proc sql;
```

```
select acq_chnl, policy_status, avg(customer_age) as Average_Customer_age from  
agents_base group by acq_chnl desc;
```

```
quit;
```

```
/*-----*/
```