Primjena makro naredbi i varijabli u stočarstvu koristeći SAS program

Lovrić, Mateja

Master's thesis / Diplomski rad

2019

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Agriculture / Sveučilište u Zagrebu, Agronomski fakultet**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:204:706896

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-18**

Repository / Repozitorij:

Repository Faculty of Agriculture University of Zagreb

Sveučilište u Zagrebu
Agronomski fakultet
University of Zagreb
Faculty of Agriculture

Primjena makro naredbi i varijabli u stočarstvu koristeći SAS program

DIPLOMSKI RAD

Mateja Lovrić

Zagreb, rujan, 2019.

Sveučilište u Zagrebu
Agronomski fakultet
University of Zagreb
Faculty of Agriculture

Diplomski studij:

Genetika i oplemenjivanje domaćih životinja

Primjena makro naredbi i varijabli u stočarstvu koristeći SAS program

DIPLOMSKI RAD

Mateja Lovrić

Mentor:
prof. dr. sc. Alen Džidić

Zagreb,rujan, 2019.

Sveučilište u Zagrebu
Agronomski fakultet

University of Zagreb
Faculty of Agriculture

IZJAVA STUDENTA

O AKADEMSKOJ ČESTITOSTI

Ja, **Mateja Lovrić**, JMBAG 0178095688, rođen/a 06.10.1994. u Zagrebu, izjavljujem da sam samostalno izradila/izradio diplomski rad pod naslovom:

Primjena makro naredbi i varijabli u stočarstvu koristeći SAS program

Svojim potpisom jamčim:

- da sam jedina autorica/jedini autor ovoga diplomskog rada;
- da su svi korišteni izvori literature, kako objavljeni tako i neobjavljeni, adekvatno citirani ili parafrazirani, te popisani u literaturi na kraju rada;
- da ovaj diplomski rad ne sadrži dijelove radova predanih na Agronomskom fakultetu ili drugim ustanovama visokog obrazovanja radi završetka sveučilišnog ili stručnog studija;
- da je elektronička verzija ovoga diplomskog rada identična tiskanoj koju je odobrio mentor;
- da sam upoznata/upoznat s odredbama Etičkog kodeksa Sveučilišta u Zagrebu (Čl. 19).

U Zagrebu, dana _____

_____

Potpis studenta / studentice

Sveučilište u Zagrebu
Agronomski fakultet

University of Zagreb
Faculty of Agriculture

IZVJEŠĆE

O OCJENI I OBRANI DIPLOMSKOG RADA

Diplomski rad studenta/ice **Mateja Lovrić**, JMBAG 0178095688, naslova

Primjena makro naredbi i varijabli u stočarstvu koristeći SAS program

obranjen je i ocijenjen ocjenom _____ , dana _____ .

Povjerenstvo:		potpisi:
1.	prof. dr. sc. Alen Džidić mentor	_____
2.	doc. dr. sc. Dragica Šalamon član	_____
3.	doc. dr. sc. Maja Ferenčaković član	_____

Zahvala

Prije svega htjela bih se zahvaliti prof. dr. sc. Alenu Džidiću koji je svojim entuzijazmom u meni probudi želju za novim znanjima u području bioinformatike. Prihvatio biti mi mentor te mi je svojim savjetima pomogao u izradi ovog diplomskog rada.

Također, zahvaljujem se svim prijateljima i kolegama, koji su bili uz mene tokom ovog studija te, te mi ga učinili lakšim i zabavnijim.

Zahvaljujem se sestrama Biljani i Marini koje su uvijek bile uz mene kada mi je to trebalo. Zahvaljujem se i šogorima Josipu i Andriji te nećacima Ivanu, Ani, Kaji i Ivi.
Želim se zahvaliti dečku Marku na razumijevanju i potpori.

Na kraju se želim zahvaliti roditeljima Slavici i Željku koji su me uvijek poticali da napredujem i ne odustajem od svojih ciljeva. Oni su uvelike zaslužni za ovaj moj uspjeh i bez njihove potpore, kada se radilo o teškim ili sretnim trenutcima, ovaj moj uspjeh ne bi bio moguć.

Sadržaj

1. Uvod 1

2. SAS-makro varijable i naredbe 2

2.1. Makro varijable 2

2.1.1. Automatske makro varijable 3
2.1.2. Makro varijable koje kreiraju korisnici 6
2.1.2.1. Kreiranje makro varijabli i dodjeljivanje vrijednosti 6

2.2. Makro naredbe 7

3. Prikaz podataka 8

4. Upotreba makro varijabli i naredbi 9

4.1. LET funkcija 9

4.2. SYMBOLGEN sistemska opcija 15

4.3. PUT funkcija 17

4.4. Kreiranje makro naredbe 18

4.5. WINDOW i DISPLAY funkcije 20

4.6. DO petlja i IF THEN ELSE petlja 21

4.7. Iterativna DO petlja 23

4.8. DO UNTIL petlja i EVAL funkcija 24

4.9. DO WHILE petlja i SYSEVALF funkcija 26

4.10. GLOBAL i LOCAL funkcije 27

4.11. STR i UPCASE funkcije 29

4.12. LENGTH i SUBSTR funkcije 30

5. Usporedba DATA koraka i makro programa 33

6. Zaključak 39

7. Popis literature 40

Sažetak

Diplomskog rada studenta/ice **Mateja Lovrić**, naslova

Primjena makro naredbi i varijabli u stočarstvu koristeći SAS program

SAS makro programi se sastoje od SAS makro naredbi i varijabli. Oni se najviše koriste u Base SAS-u i to za prilagođavanje korisnikovim potrebama: procesiranja datoteka, koraka u procedurama, u kreiranju izvješća, te u prebacivanju podataka između koraka u SAS programu. SAS makro programi smanjuju potrebnu količinu napisanog koda i zato su puno učinkovitiji od običnih SAS programa. U današnje vrijeme sve je više empirijskih podataka u stočarstvu, te je uz pomoć SAS makro programa moguće obraditi empirijske podatke puno učinkovitije. Stoga će se u ovom radu objasniti korištenje SAS makro naredbi i varijabli na primjerima iz područja stočarstva.

Ključne riječi: SAS program, SAS makro varijable, SAS makro naredbe

Summary

Of the master's thesis – student **Mateja Lovrić,** entitled

Application of macro commands and variables in animal husbandry using SAS program

SAS macro programs consist of SAS macro commands and variables. They have been used mostly in BASE SAS program to meet user need such as: data set processing, steps in procedures, report writing and to move data from different steps in SAS program. SAS macro programs reduce amount of written code and therefore they are much more efficient than regular SAS programs. Today a lot of empirical data exist in the animal husbandry and therefore by using macro programs much more data can be processed efficiently. Therefore, this thesis will explain benefits of using SAS macro programs in the field of animal husbandry efficiently.

Keywords: SAS program, SAS macro variables, SAS macro commands

1. Uvod

Makro programi ili makro jezik sadrže makro naredbe, makro varijable i makro funkcije pomoću kojih komuniciraju sa makro procesorom unutar SAS programa. Vrlo je važno poznavati makro naredbe jer su one dosta slične običnim naredbama koje se koriste unutar DATA ili PROC koraka u običnom SAS programu. A kako bi učinkovito koristili SAS makro programe, važno je znati da se makro programi izvode prije običnog koda u DATA i PROC koraku SAS programa. S obzirom da makro program ispisuje novi program nekad se popularno takav način pisanja programa zove i meta-programiranje. U stočarskoj praksi vrlo je malo prikaza o korisnosti i prednostima koje SAS makro programi imaju u odnosu na uobičajene SAS programe, te je stoga ovaj diplomski rad prilog boljem poznavanju navedenog. Alati u makro jeziku nam omogućavaju prolazak informacija između SAS koraka, dinamičko kreiranje kodova, uvjetno ispunjavanje DATA ili PROC koraka te kreiranje fleksibilnog koda. Makro jezik se sastoji od vlastitog seta opcija, sintakse i funkcija.

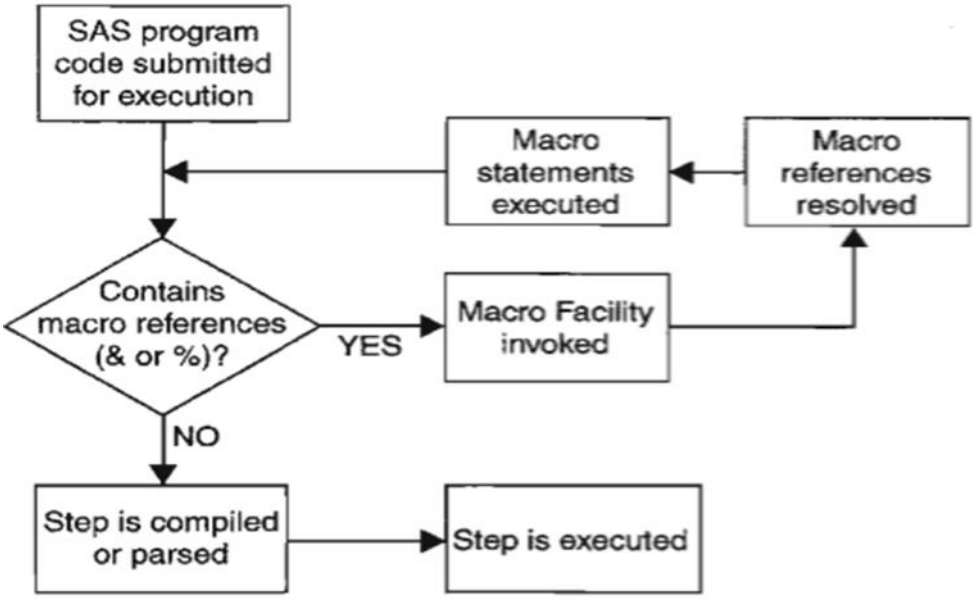
Cilj ovog rada je pokazati na primjerima kako bi upotreba SAS makro programa odnosno njegovih marko varijabli, naredbi i funkcija mogla koristiti u obradi podataka u poljoprivredi točnije stočarstvu.

2. SAS-makro varijable i naredbe

Prilikom kreiranja svakog makro programa koristimo dva osnovna konstitutivna dijela, a to su makro naredbe i makro varijable. Vrlo ih je lagano razlikovati jer su makro varijable uvijek označene sa znakom (&), dok su makro naredbe označene sa znakom (%). Makro varijable i naredbe su alati koji omogućavaju izmjenu teksta u SAS programu zamjenom simbola.

2.1. Makro varijable

Postoje makro varijable definirane od strane makro programera te se one nazivaju makro varijable koje definiraju korisnici. Makro varijable koje definira sam makro procesor nazivaju se automatske makro varijable. Makro varijable mogu se definirati i koristiti bilo gdje u SAS programima, osim unutar podatkovnih linija.



Slika 1. Prikaz kako radi makro procesor (Carpenter, 2004)

Na Slici 1. vidimo da SAS obrađuje makro varijable i naredbe prije DATA i PROC koraka. SAS makro procesor aktiviraju simboli % i &. Isto tako SAS makro procesor prvo obrađuje makro varijable i naredbe prije nego što počne provoditi običan SAS kod u SAS konzoli.

Kad definiramo makro varijablu, makro procesor je dodaje u jednu od tablica simbola makro varijabli programa. Kada je makro varijabla definirana u kodu koji je izvan definicije makro naredbe ili kada varijablu stvori makro procesor, varijabla se nalazi u tablici globalnih simbola, koju makro procesor stvara na početku SAS sesije. Kada se makro varijabla definira

unutar same makro naredbe i nije posebno definirana kao globalna, varijabla se obično nalazi u tablici lokalnih simbola makro naredbe, koju SAS stvara kada se makro naredba započne izvršavati.

Globalna makro varijabla je dostupna u bilo kojem dijelu SAS koda. Varijabla koja se nalazi u lokalnoj tablici simbola se naziva lokalna makro varijabla. Lokalna makro varijabla je dostupna samo za vrijeme izvođenja makro naredbe.

2.1.1. Automatske makro varijable

Prilikom pokretanja SAS-a makro procesor sam stvara automatske makro varijable koje daju informacije za tu SAS sesiju. Automatske makro varijable su globalne osim SYSPBUFF, koja je lokalna. Kako bi aktivirali automatsku makro varijablu prije nje se mora navesti simbol &, a nakon nje slijedi naziv automatske makro varijable (na primjer, & SYSJOBID).

Dio automatskih makro varijabli ima status da im se može dodijeliti nova vrijednosti. Sljedeća tablica sadrži automatske makro varijable koje stvara SAS makro procesor i njihov status unutar SAS sesije. Koristimo % PUT _AUTOMATIC_ za pregled svih dostupnih automatskih makro varijabli.

Tablica 1. Automatske makro varijable (SAS 9.4 Macro Language:Reference, Fifth Edition)

Status	Varijabla	Sadržaj
Automatske makro varijable kojima možemo mijenjati vrijednost	SYSBUFER	Nejednak tekst iz %INPUT
	SYSCC	Kod koji vraća SAS u radno okruženje
	SYSCMD	Zadnja neprepoznata naredba iz naredbenog retka
	SYSDEVIC	Naziv grafičkog uređaja
	SYSDMG	Povratni kod koji vrši radnju na oštećenom skupu podataka
	SYSDSN	Naziv najnovijeg SAS datoteke u dva polja
	SYSFILRC	Povratni kod postavljen FILENAME naredbom
	SYSLAST	Naziv najnovijeg SAS datoteke u jednom polju
	SYSLCKRC	Povratni kod postavljen naredbom LOCK
	SYSLIBRC	Povratni kod postavljen LIBNAME naredbom
	SYSLOGAPPLNAME	Vrijednost opcije LOGAPPLNAME
	SYSMSG	Poruka za prikaz u makro prozoru

	SYSPARM	Vrijednost navedena sa SYSPARM=opcija sistema
	SYSPBUFF	Tekstualni parametar makro varijable
	SYSRC	Razni povratni kodovi povezani sa sustavom
Automatske makro varijable kojima ne možemo mijenjat vrijednost	SYSADDRBITS	Broj bitova adrese
	SYSHARWIDTH	Vrijednost duljine znaka
	SYSDATASTEPPHASE	Vrijednost trenutne faze pokretanja datoteke
	SYSDATE	Znakovna vrijednost koja predstavlja datum kad se SAS sesija započinje (dvoznamenkasta godina)
	SYSDATE9	Znakovna vrijednost koja predstavlja datum pošetka SAS sesije (četveroznamenkasta godina)
	SYSDAY	Dan u tjednu kad SAS sesija započinje
	SYSENCODING	Naziv SAS sesije kodiranja
	SYSENDIAN	Naznaka redoslijeda bitova u trenutnoj sesiji
	SYSENV	Prednji ili stražnji indikator
	SYSERR	Povratni kod postavljen SAS postupcima i DATA korakom
	SYSERRORTEXT	Tekst zadnje poruke o pogrešci formatirane za prikaz u SAS dnevniku
	SYSHOSTINFOLONG	Informacije o radnom okruženju ako je navedena opcija HOSTINFOLONG
	SYSHOSTNAME	Naziv radnog operativnog sustava
	SYSINCKUDEFILEDEVICE	Tip uređaja na kojem je %INCLUDE datoteka
	SYSINCLUDEFILEDIR	Direktorij u kojem je %INCLUDE datoteka pronađena
	SYSINCLUDEFILEFILEREF	Referenca o datoteci koja je povezana sa %INCLUDE datotekom ili prazninom
	SYSINCLUDEFILENAME	Ime trenutne %INCLUDE datoteke
	SYSINDEX	Broj makro varijabli koje su izvedene za vrijeme SAS sesije
	SYSINFO	Informacije o povratnom kodu
	SYSJOBID	Naziv trenutačnog posla ili ID korisnika
	SYSMACRONAME	Naziv trenutne makro varijable koja se izvršava

	SYSMAXLONG	Vraća najveću moguću brojčanu vrijednost koju mu Linux dopušta
	SYSMENV	Trenutno okruženje izvršenja naredbe
	SYSNCPU	Trenutni broj procesora koje bi SAS mogao koristiti u izračunu
	SYSNOBS	Broj opažanja u zadnjoj datoteci kreiranoj u DATA koraku ili procesu
	SYSODESCAPECHAR	Vrijednost od ODS ESCAPECHAR= iz programa
	SYSODSPATH	Vrijednost PATH varijable u ODS-u
	SYSPRINTTOLIST	Putanja LIST datoteke prema PRINTTO proceduri budućih preusmjeravanja
	SYSPRINTTOLOG	Putanja LOG datoteke prema PRINTTO proceduri budućih preusmjeravanja
	SYSPROCESSID	ID trenutnog SAS procesa
	SYSPROCESSMODE	Naziv trenutnog SAS načina rada ili tipa servera
	SYSPROCESSNAME	Naziv trenutnog SAS procesa
	SYSPROCNAME	Naziv trenutne procedure koja se izvodi
	SYSSCP	Kratica za operacijski sustav
	SYSSCPL	Naziv operacijskog sustava
	SYSSITE	Vrijednost pripisana vašoj web stranici
	SYSSIZEOFLONG	Dužina u bitovima najdužeg znaka u trenutnoj sesiji
	SYSSIZEOFPTR	Veličina pokazivača u bitovima
	SYSSIZEOFUNICODE	Dužina u bitovima „Unicode“ znaka u trenutnoj sesiji
	SYSSTARTID	ID generiran od zadnje STARTSAS izjave
	SYSTARTNAME	Naziv procesa generiranog od zadnje STARTSAS izjave
	SYSCPIPHOSTNAME	Nazivi lokalnih i udaljenih operativnih okruženja kad su podržani sa više TCP / IP
	SYSTIME	Znakovna vrijednost vremena u kojem SAS sesija započinje posao
	SYSTIMEZONE	Naziv vremenske zone po TIMEZONE opciji
	SYSTIMETONEIDENT	ID vremenske zone po TIMEZONE opciji
	SYSTIMEZONEOFFSET	Trenutni pomak vremenske zone na temelju TIMEZONE opcije
	SYSUSERID	ID korisnika ili prijava u trenutni SAS proces

	SYSVER	Broj verzije SAS programa
	SYSVLONG	Broj verzije SAS programa sa dvoznamenkastom godinom
	SYSVLONG4	Broj verzije SAS programa sa četveroznamenkastom godinom
	SYSWARNINGTEXT	Tekst zadnje poruke upozorenja formatiran za prikaz u SAS dnevniku

Postoje i makro varijable koje su specifične za neki sustav, a koje su stvorene samo na određenoj platformi operativnog sustava. SAS softverski proizvodi također pružaju makro varijable koje su opisane u dokumentaciji za proizvod koji ih koristi. Nijedna od tih vrsta makro varijabli ne smatra se automatskom makro varijablom.

2.1.2. Makro varijable koje kreiraju korisnici

Moguće je kreirati vlastite makro varijable, promijeniti njihove vrijednosti te ih definirati. Makro varijablu je moguće kreirati unutar makro naredbe i kao globalnu varijablu definirajući je naredbom GLOBAL, ali može se i posebno definirat. Sami nazivi makro varijabli moraju započeti slovom ili donjom crtom, a mogu sadržavati slova ili znamenke. Makro varijabli se može dodijeliti bilo koje ime sve dok ime nije rezervirana riječ, odnosno riječ koja se već koristi u SAS programu za određene funkcije. Prefiksi za naziv makro varijable AF, DMS, SQL i SYS se ne preporučuju jer se koriste u SAS-u za određene funkcije. Korištenje jednog od ovih prefiksa može uzrokovati probleme sa makro varijablama prilikom njihovog korištenja. Ako makro varijabli dodijelimo riječ koja se već koristi u SAS programu, kako funkcija, SAS dnevnik će nam ispisati pogrešku.

2.1.2.1. Kreiranje makro varijabli i dodjeljivanje vrijednosti

Makro varijable se najlakše kreiraju pomoću funkcije LET te im se tako dodjeljuje vrijednost. U primjeru %LET Grad=Zagreb; Grad je naziv makro varijable, a Zagreb je vrijednost makro varijable. Vrijednost makro varijable je niz znakova. Znakovi mogu biti slova, brojevi ili simboli za ispis koji se nalaze na tipkovnici , te praznine između znakova. Ako makro varijabla već postoji onda dodijeljena vrijednost zamjenjuje njezinu trenutnu vrijednost. Ako makro varijabla ili njena vrijednost sadrži znakove koji automatski prepoznaju makro naredbe (% ili &), simbol sa imenom naredbe se procjenjuje prije dodjeljivanja vrijednosti makro naredbi. Osim LET funkcije makro varijable moguće je definirati pomoću PUT funkcije. Kada koristimo funkciju PUT za opis makro varijabli, ona uključuje samo makro varijable koje postoje u vrijeme izvršenja SAS koda. Opis makro varijable sadrži definiciju,

naziv i vrijednost makro varijable. Vrijednosti makro varijable koje su preduge za trenutnu veličinu retka prelaze u sljedeći redak ili retke. Makro varijable (lokalne i globalne) su navedene u SAS dnevniku.

Općenito makro procesor tretira vrijednosti makro varijabli kao što su abecedni znakovi, znamenke i simbole kao znakove. SAS makro procesor može tretirati & i % kao znakove, ali pomoću STR funkcije. Makro procesor ne radi razliku između znakovnih i numeričkih vrijednosti kao što to općenito SAS radi. Vrijednost varijable može biti u rasponu od 0 do 65.534 znakova. Ako se izostavi argument vrijednosti, automatska dodijeljena vrijednost je nula.

2.2. Makro naredbe

Makro naredbe su skupina SAS izraza koje se koriste u SAS programu bilo gdje, koristeći to ime. Postoje tri vrste makro izraza, a to su tekstualni, logički i aritmetički. Tekstualni izraz je svaka kombinacija teksta, makro varijabli ili makro funkcija. Logički izrazi i aritmetički izrazi su nizovi operatora i operanada koji se procjenjuju kako bi se dobio rezultat. Aritmetički izraz sadrži aritmetički operator.

Osim izraza LET, postoje i druge značajke jezika makro naredbe koje stvaraju makro varijable unutar makro naredbe kao što su; iterativna DO petlja, GLOBAL naredba, INPUT naredba, INTO petlja, SELECT izraz u SQL-u, LOCAL naredba, MACRO naredba, SYMPUT rutina i SYMPUTN rutina u SCL-u te WINDOW naredba.

Sintaksa za kreiranje makro naredbe
% MAKRO <ime makronaredbe>;
Makro izjave;
%MEND;

Makro naredbe omogućuju izvršavanje SAS koda pozivanjem imena makro naredbe. Makro naredbe su još učinkovitije kada im dodamo parametre. Parametri daju makro naredbama fleksibilnost.

Definiranu makro naredbu, makro procesor sprema u trenutnoj SAS sesiji. Makro naredbe se sastoje od makro naredbi teksta i programskih izjava.

3. Prikaz podataka

Podaci korišteni u ovom radu sadrže četiri varijable; županija (County), općina (Township), broj isporučitelja (Number_of_suppliers) i količina mlijeka u kilogramima (Quantity_of_milk) koje nam označavaju geografsko područje s kojeg su prikupljeni podaci, broj isporučitelja mlijeka te količinu mlijeka u kilogramima. U samim primjerima koristit ćemo podatke preuzete sa stranica Hrvatske poljoprivredne agencije (http://hpa.mps.hr/) za brojčani prikaz količine proizvedenog mlijeka po županijama i općinama od 2015. godine do 2018. godine.

4. Upotreba makro varijabli i naredbi

U ovom poglavlju biti će prikazano nekoliko primjera kako upotreba SAS makro varijabli i naredbi se može koristiti u stočarstvu.

4.1. LET funkcija

Jedan od najjednostavnijih načina za kreiranje marko varijabli je LET funkcija. LET funkcija je popraćena sa nazivom makro varijable te znakom jednakosti poslije kojeg dolazi zadana vrijednost makro varijable. Vrijednost makro varijable nije spremljena kao karakterna ili numerička vrijednost nego kao tekst. Sve što je iza znaka jednakosti SAS uzima kao vrijednost makro varijable. Makro varijabla može biti definirana van makro naredbe te unutar makro naredbe. Ako je definirana van makro naredbe njezina vrijednost će biti dostupna kroz cijeli program, ali ako se nalazi unutar makro naredbe njezina vrijednost će biti dostupna samo unutar makro naredbe, tj. bit će lokalna.

Sintaksa za upotrebu LET funkcije

%LET makro varijabla =<vrijednost> ;

Automatske makro varijable koje će biti korištene u primjeru nalaze se u Tablici 1. ,a to su SYSINFO i SYSSIZEOFLONG.

Objašnjenje sintakse za upotrebu automatskih makro varijabli

&SYSINFO-informacije o povratnom kodu
&SYSSIZEOFLONG-dužina u bitovima najdužeg znaka u trenutnoj sesiji

SAS primjer 1. za upotrebu LET funkcije i automatskih makro varijabli

```
%let podaci=prog.Podaci18;
title "Prikaz podataka";
data prog.prikaz;
set &podaci;
if County="GRAD ZAGREB";
run;
proc print data=prog.prikaz;
footnote1 &sysinfo;
footnote2 &syssizeoflong;
run;
```

U prvom primjeru je prikaz najjednostavnijeg formiranja makro varijable pomoću funkcije LET. U primjeru ćemo vidjeti kako u LET funkciji možemo koristiti podatke tako što ćemo ih

jednostavno označimo sa &podaci. U LET funkciji, koja se nalazi iznad samog koda, možemo pozvati drugi set podataka bez izmjene samog koda. Osim što smo pozvali taj set podataka sortirali smo ga IF funkcijom prema varijabli County gdje je njena vrijednost Grad Zagreb. U primjeru vidimo i automatske makro varijable SYSINFO i SYSSIZEOFLONG, koja nam daje informacije o kodu i dužini u bitovima najdužeg znaka u trenutnoj sesiji. Automatske makro varijable su nam prikazane u FOOTNOTE-ama ispod same tablice. Osim makro varijable dodali smo naziv prikazu te smo procedurom PROC PRINT prikazali podatke.

Prikaz podataka

Obs	County	Township	Number_of_suppliers	Quantity_of_milk
1	GRAD ZAGREB	Grad Zagreb	24	1.592.433

0
4

Slika 2. Prikaz rezultata korištenja LET funkcije

Na Slici 2. vidimo rezultate koje nam SAS ispisuje kao rezultat koda. Na slici vidimo da tablica ima naziv te da je on „Prikaz podatak“, vidimo da ispisuje koliko ima opažanja te ispisuje varijable koje ispunjavaju uvjete navedene u kodu. Jedini uvjet je bio da varijabla County koja sadrži naziv županije bude Grad Zagreb. Ispod same tablice imamo FOOTNOTE u koje smo stavili automatske makro varijable SYSINFO i SYSSIZEOFLONG. Vrijednost nula predstavlja vrijednost SYSINFO. SYSSIZEOFLONG ima vrijednost četiri te nam je to dužina u bitovima najdužeg znaka u trenutnoj sesiji.

PROC MEANS je osnovni korak unutar SAS programa koji se prvenstveno koristi za računanje opisne statistike. PROC koraci mogu se koristiti i za provođenje nekih osnovnih statističkih analiza. Uz pomoć LET funkcije izvest ćemo osnovnu statistiku za podatke iz 2015. godine.

SAS primjer 2. korištenje LET funkcije i PROC MEANS procedure

```
%let podaci=prog.Podaci15;
proc means data=&podaci mean;
class County;
var Quantity_of_milk;
run;
```

U Primjeru 2. vidimo proceduru PROC MEANS koja nam radi srednju vrijednost varijable Quantity_of_milk te ju klasificira po varijabli County koja sadrži županije iz kojih su podaci.

Automatske makro varijable koje smo u Primjeru 1. stavili u FOOTNOTE ako ih ne isključimo će se prikazivati dok ih ne isključimo ili ne zamijenimo.

The MEANS Procedure

Analysis Variable : Quantity_of_milk		
County	N Obs	Mean
BJELOVARSKO-BILOGORSKA ŽUPANIJA	23	3302228.04
BRODSKO-POSAVSKA ŽUPANIJA	23	627623.48
GRAD ZAGREB	1	1822055.00
ISTARSKA ŽUPANIJA	22	400219.45
KARLOVAČKA ŽUPANIJA	17	1072547.76
KOPRIVNIČKO-KRIŽEVAČKA ŽUPANIJA	25	2504327.56
KRAPINSKO-ZAGORSKA ŽUPANIJA	27	294818.07
LIČKO-SENJSKA ŽUPANIJA	9	620826.00

Slika 3. Prikaz rezultata PROC MEANS procedure iz Primjera 2.

Slika 3. nam prikazuje kako izgleda tablica nakon završetka rada koda iz drugog primjera. Možemo vidjeti da nam tablica nema naslov, nego nam je u prvom redu tablice naziv procedure. Možemo vidjeti da su na osnovu podataka za županije i količine mlijeka po županijama, napravljeni prosjeci za svaku od njih. Osim toga u tablici vidimo kolonu sa brojem opažanja koja se odnosi na količinu mlijeka po kojoj je napravljena srednja vrijednost za svaku županiju.

PROC COMPARE uspoređuje dva seta podataka. Postupak određuje podudaranje varijabli i podudaranja opažanja unutar dva seta podataka.

SAS primjer 3. za upotrebu LET funkcije, PROC MEANS i PROC COMPARE procedura

```
%let podaci=prog.Podaci15;
%let podaci2=prog.Podaci16;
%let naziv=Podaci15;
%let naziv2=Podaci16;
%let naziv3=Compare;
proc means data=&podaci mean max min;
class County;
var Quantity_of_milk;
title &naziv;
run;
proc means data=&podaci2 mean max min;
class County;
```

```
var Quantity_of_milk;
title &naziv2;
run;
proc compare base=&podaci
compare=&podaci2;
var Quantity_of_milk;
title &naziv3;
run;
```

U Primjeru 3. vidimo dva koda koja se odnose na obradu podataka te naredbe koje će se primijeniti unutar kodova, a to su max- maksimalna vrijednost, min- minimalna vrijednost te mean -srednja vrijednost varijable. Naredba PROC MEANS, sa dodatnim opcijama CLASS i VAR, se odnosi na varijablu količina mlijeka(Quantity_of_milk) te se ona klasificira(CLASS) po drugoj varijabli(VAR) koja sadrži naziv županije(County). Iznad kodova vidimo da su uvedene makro varijable podaci, podaci2, naziv, naziv2 i naziv3, te da se one odnose na setove podataka i na nazive na koje se funkcije izvršavaju. PROC MEANS ispisuje opisnu statistiku za podatke iz 2015. godine i 2016. godine dok PROC COMPARE uspoređuje podatke tih godina.

Podaci15

The MEANS Procedure

Analysis Variable : Quantity_of_milk				
County	N Obs	Mean	Maximum	Minimum
BJELOVARSKO-BILOGORSKA ŽUPANIJA	23	3302228.04	13052246.00	203154.00
BRODSKO-POSAVSKA ŽUPANIJA	23	627623.48	3045793.00	7446.00
GRAD ZAGREB	1	1822055.00	1822055.00	1822055.00
ISTARSKA ŽUPANIJA	22	400219.45	3590222.00	4727.00
KARLOVAČKA ŽUPANIJA	17	1072547.76	5132257.00	44362.00
KOPRIVNIČKO-KRIŽEVAČKA ŽUPANIJA	25	2504327.56	9432052.00	88009.00
KRAPINSKO-ZAGORSKA ŽUPANIJA	27	294818.07	1139946.00	2260.00
LIČKO-SENJSKA ŽUPANIJA	9	620826.00	2103314.00	12709.00
MEĐIMURSKA ŽUPANIJA	22	734692.64	1650318.00	5817.00
OSJEČKO-BARANJSKA ŽUPANIJA	39	3818727.33	37376179.00	27957.00
POŽEŠKO-SLAVONSKA ŽUPANIJA	10	1121652.50	2535533.00	614158.00
PRIMORSKO-GORANSKA ŽUPANIJA	6	36202.50	90010.00	2010.00
SISAČKO-MOSLAVAČKA ŽUPANIJA	18	959014.33	2971291.00	12951.00
SPLITSKO-DALMATINSKA ŽUPANIJA	10	88937.90	416297.00	348.0000000

Slika 4. Prikaz rezultata za podatke iz 2015.godine iz Primjera 3.

Podaci16

The MEANS Procedure

Analysis Variable : Quantity_of_milk				
County	N Obs	Mean	Maximum	Minimum
BJELOVARSKO-BILOGORSKA ŽUPANIJA	23	3271026.00	12623890.00	180340.00
BRODSKO-POSAVSKA ŽUPANIJA	22	596136.45	2894007.00	7233.00
GRAD ZAGREB	1	1819965.00	1819965.00	1819965.00
ISTARSKA ŽUPANIJA	21	267627.62	1044588.00	2973.00
KARLOVAČKA ŽUPANIJA	17	1007106.29	4662662.00	31354.00
KOPRIVNIČKO-KRIŽEVAČKA ŽUPANIJA	25	2394323.64	9177911.00	46922.00
KRAPINSKO-ZAGORSKA ŽUPANIJA	27	279865.11	978786.00	2389.00
LIČKO-SENJSKA ŽUPANIJA	9	604868.22	1878907.00	31891.00
MEĐIMURSKA ŽUPANIJA	22	721449.68	1735337.00	1277.00
OSJEČKO-BARANJSKA ŽUPANIJA	39	3765685.90	38180827.00	33666.00
POŽEŠKO-SLAVONSKA ŽUPANIJA	10	1005526.40	2358477.00	577434.00
PRIMORSKO-GORANSKA ŽUPANIJA	5	42622.20	102406.00	11374.00
SISAČKO-MOSLAVAČKA ŽUPANIJA	17	906265.47	2750090.00	10527.00

Slika 5. Prikaz rezultata iz 2016.godine iz Primjera 3.

Na Slikama 4. i 5. vidimo nazive tablica prema godini seta podataka. Opisnu statistiku je izradila PROC MEANS procedura. Na Slikama 4. i 5. prva kolona je varijabla koja sadrži županiju (County) a druga je broj opažanja. Dok se u trećoj, četvrtoj i petoj koloni nalaze prosječna, minimalna i maksimalna količina mlijeka po županijama.

Compare

The COMPARE Procedure
Comparison of PROG.PODACI15 with PROG.PODACI16
(Method=EXACT)

Data Set Summary

Dataset	Created	Modified	NVar	NObs
PROG.PODACI15	30AUG19:16:58:47	30AUG19:16:58:47	4	368
PROG.PODACI16	30AUG19:16:59:00	30AUG19:16:59:00	4	387

Variables Summary

Number of Variables in Common: 4.
Number of VAR Statement Variables: 1.

Slika 6. Prikaz rezultata PROC COMPARE procedure iz Primjera 3.

Na Slici 6. vidimo usporedbu između setova podataka iz 2015. godine i 2016. godine. Ispis procedure COMPARE prikazuje naziv datoteke, vrijeme njenog kreiranja, vrijeme zadnje promjene, ukupan broj varijabli, ukupan broj opažanja, broj zajedničkih varijabli i broj varijabli u VAR naredbi.

Observation Summary

Observation	Base	Compare
First Obs	1	1
First Unequal	1	1
Last Unequal	367	367
Last Match	368	368
Last Obs	.	387

Number of Observations in Common: 368.
Number of Observations in PROG.PODACI16 but not in PROG.PODACI15: 19.
Total Number of Observations Read from PROG.PODACI15: 368.
Total Number of Observations Read from PROG.PODACI16: 387.

Number of Observations with Some Compared Variables Unequal: 367.
Number of Observations with All Compared Variables Equal: 1.

Values Comparison Summary

Number of Variables Compared with All Observations Equal: 0.
Number of Variables Compared with Some Observations Unequal: 1.
Number of Variables with Missing Value Differences: 1.
Total Number of Values which Compare Unequal: 367.
Maximum Difference: 37749347.

Slika 7. Prikaz rezultata PROC COMPARE procedure iz Primjera 3.

Slika 7. Prikazuje usporedbu i broj opažanja varijable Quantity_of_milk (količina predanog mlijeka) iz podataka 2015. godine i 2016. godine. Vidimo broj zajedničkih opažanja između setova iz 2015. godine i 2016. godine koji je 368. Možemo vidjeti broj opažanja koji je zabilježen u podacima iz 2016. godine, ali nije zabilježen u podacima 2015. godine. Podaci iz 2016. godine imaju 19 opažanja više od podataka iz 2015. godine. Imamo ispisan ukupan broj opažanja prema svakoj godini posebno. Zabilježeno je 367 različitih opažanja te jedna vrijednost koja nedostaje.

Compare

The COMPARE Procedure
Comparison of PROG.PODACI15 with PROG.PODACI16
(Method=EXACT)

All Variables Compared have Unequal Values

Variable	Type	Len	Ndif	MaxDif	MissDif
Quantity_of_milk	NUM	8	367	3.775E7	6

Slika 8. Prikaz rezultata PROC COMPARE procedure iz Primjera 3.

Na Slici 8. možemo vidjeti kako uspoređeni setovi podataka za 2015. godinu i 2016. godinu imaju različite vrijednosti za varijablu Quantity_of_milk (količina predanog mlijeka). PROC COMPARE procedura usporedila je varijable Quantity_of_milk iz setova podataka za 2015. godinu i 2016. godinu. Varijabla Quantity_of_milk je numerička varijabla sa dužinom znakova 8. Ima 367 različitih opažanja za setove podataka iz 2015. i 2016. godine, s maksimalnom razlikom varijable između promatranih setova podataka, 3.775E7, te 6 vrijednosti koje nedostaju.

4.2. SYMBOLGEN sistemska opcija

Sistemska opcija SYMBOLGEN prikazuje status/vrijednost makro varijabli i naredbi u SAS dnevniku. Ova je opcija korisna za uklanjanje pogrešaka tijekom pisanja makro varijabli i naredbi. Isto tako ako ne želimo da nam se više prikazuje moramo je isključiti opcijom NOSYMBOLGEN.

Sintaksa za upotrebu opcije SYMBOLGEN
SYMBOLGEN/NOSYMBOLGEN

SAS primjer 4. za upotrebu opcije SYMBOLGEN

```
options symbolgen;
%let a=Number_of_suppliers;
%let b=Quantity_of_milk;
%let C=Township;
data new;
set prog.Podaci16;
if &b>&a then;
do &C =".";
end;
run;
```

U Primjeru 4. možemo vidjeti uvođenje opcije SYMBOLGEN koju ćemo objasniti ispod Slike 8. Vidimo kako imamo tri makro varijable te su one varijable iz seta podataka koji se naziva Podaci16, ti podaci se odnose na 2016. godinu. Ispod DATA koraka vidimo korištenje IF THEN DO petlje pomoću koje smo sortirali podatke prema odnosu (veće,manje ili jednako) između makro varijabli &b i &a odnosno između varijable sa količinom mlijeka (Quantity_of_mlik) te varijable sa brojem predavača mlijeka (Number_of_suppliers). Nakon toga smo pomoću IF THEN DO petlje kreirali naredbu koja će nam ako je izraz istinit dodijeliti vrijednost točka varijabli &C odnosno varijabli koja sadrži nazive općina (Township). Petlja IF THEN DO mora završiti sa END funkcijom.

```
230  options symbolgen;
231  %let a=Number_of_suppliers;
232  %let b=Quantity_of_milk;
233  %let C=Township;
234  data new;
235  set prog.Podaci16;
SYMBOLGEN:  Macro variable B resolves to Quantity_of_milk
236  if &b>&a then;
SYMBOLGEN:  Macro variable A resolves to Number_of_suppliers
SYMBOLGEN:  Macro variable C resolves to Township
237  do &C =".";
238  end;
239  run;

NOTE: There were 387 observations read from the data set PROG.PODACI16.
NOTE: The data set WORK.NEW has 387 observations and 4 variables.
NOTE: DATA statement used (Total process time):
      real time           0.06 seconds
      cpu time            0.04 seconds
```

Slika 9. Prikaz SAS dnevnika za upotrebu SYMBOLGEN opcije iz Primjera 4.

Na Slici 9. vidimo ispis iz SAS dnevnika (SAS log). Možemo vidjeti kako SAS kod dok ga čita SAS program nalazi vrijednosti za makro varijable koje se koriste. Vrijednosti makro varijabli se u Primjeru 4. nalaze uz LET funkciju.

	County	Township	Number_of_suppliers	Quantity_of_milk
1	BJELOVARSKO-BILOGORSKA ŽUPANIJA	.	30	1,245,346
2	BJELOVARSKO-BILOGORSKA ŽUPANIJA	.	136	7,575,229
3	BJELOVARSKO-BILOGORSKA ŽUPANIJA	.	43	4,682,655
4	BJELOVARSKO-BILOGORSKA ŽUPANIJA	.	26	836,371
5	BJELOVARSKO-BILOGORSKA ŽUPANIJA	.	137	4,261,226
6	BJELOVARSKO-BILOGORSKA ŽUPANIJA	.	48	2,354,136

Slika 10. Prikaz tablice dobivene pomoću IF THEN DO petlje iz Primjera 4.

Slika 10. prikazuje datoteku koja se nalazi u WORK direktoriju jer nije zadan poseban library za spremanje podataka. U varijabli Township vrijednosti jednake točki su one gdje je varijabla Quantity_of_milk (varijable koja sadrži količinu mlijeka) veća od vrijednosti varijable Number_of_suppliers (varijable koja sadrži broj predavača mlijeka).

4.3. PUT funkcija

PUT funkcija se koristi za definiciju makro varijabli. Uz PUT funkciju nalaze se definicija, naziv i vrijednost makro varijable.

Sintaksa za upotrebu PUT funkcije

%PUT <tekst | _ALL_ | _AUTOMATIC_ | _GLOBAL_ | _LOCAL_ | _READONLY_ | _USER_ | _WRITABLE_>;

SAS primjer 5. za upotrebu PUT funkcije

```
%put _automatic_;
```

U Primjeru 5. vidimo korištenje funkcije PUT pomoću kojega možemo kreirati makro varijable. Ovdje vidimo samo kratki SAS kod koji nam pokazuje sve automatske marko varijable. Osim automatic opcije još često korištena je i user opcija, koja ispisuje makro varijable kreirane od strane korisnika.

```
284  %put _automatic_;
AUTOMATIC AFDSID 0
AUTOMATIC AFDSNAME
AUTOMATIC AFLIB
AUTOMATIC AFSTR1
AUTOMATIC AFSTR2
AUTOMATIC FSPBDV
AUTOMATIC SYSADDRBITS 64
AUTOMATIC SYSBUFFR
AUTOMATIC SYSCC 3000
AUTOMATIC SYSCHARWIDTH 1
AUTOMATIC SYSCMD END
AUTOMATIC SYSDATASTEPPHASE
AUTOMATIC SYSDATE 09SEP19
AUTOMATIC SYSDATE9 09SEP2019
AUTOMATIC SYSDAY Monday
AUTOMATIC SYSDEVIC
AUTOMATIC SYSDMG 0
AUTOMATIC SYSDSN WORK    NEW
AUTOMATIC SYSENCODING wlatin2
AUTOMATIC SYSENDIAN LITTLE
AUTOMATIC SYSENV FORE
AUTOMATIC SYSERR 3000
```

Slika 11. Prikaz SAS dnevnika za upotrebu PUT funkcije iz Primjera 5.

Na Slici 11. vidimo dio automatskih makro varijabli koje nam se prikazuju u SAS dnevniku te njihove vrijednosti koje im dodjeljuje SAS za trenutnoj sesiji.

4.4. Kreiranje makro naredbe

Funkcija MACRO započinje definiciju makro naredbe, dodjeljuje makro ime i može sadržavati definiciju makro naredbe. Prije nego pozovemo makro naredbu istu moramo definirati. Funkcija MACRO može se pojaviti bilo gdje u SAS programu, osim u podatkovnim linijama.

Sintaksa za upotrebu MACRO funkcije

%MACRO makro ime <(lista parametara)></ opcija-1 <opcija-n>>;

U Tablici 1. nalaze se automatske makro varijable koje će biti korištene u Primjeru 6.,a to su SYSDATE i SYSINDEX.

Objašnjenje sintakse za upotrebu automatskih makro varijabli

&SYSDATE-znakovna vrijednost koja predstavlja datum kad se SAS sesija započinje (dvoznamenkasta godina)

&SYSINDEX- broj makro naredbi koje su izvedene za vrijeme SAS sesije

SAS primjer 6. za upotrebu MACRO funkcije i automatskih makro varijabli

```
%macro m(ime);
  proc print data=&ime;
  where County="ISTARSKA ŽUPANIJA";
    title "&ime, &sysdate, &sysindex";
    footnote "&ispod";
  run;
  %put _user_;
%mend m;
%let ispod=ISTARSKA ŽUPANIJA;
%m(PROG.Podaci18)
```

U Primjeru 6. možemo vidjeti kako se kreira marko naredba pomoću MACRO funkcije. Izraz ime u zagradi označava pozicijski parametar. Pozicijski parametar definiramo kada pozivamo makro naredbu. U redovima ispod samog naziva možemo vidjeti kako unutar naredbe m imamo opciju PROC PRINT što znači da kada pozovemo tu naredbu ona će prikazati trenutni set podataka. Osim opcije PROC PRINT imamo i uvjet, da varijabla County mora biti jednaka vrijednosti „ISTARSKA ŽUPANIJA“. Title nam označava naziv iznad tablice. SYSDATE nam označava datum trenutne SAS sesije te SYSINDEX broj makro naredbi koje su izvedene za vrijeme trajanja sesije. Naziv ispod se odnosi na FOOTNOTE, a njegova vrijednost je definirana ispod makro naredbe. Pomoću funkcije PUT makro naredbu smještamo u tablicu naredbi koju je napravio korisnik. Makro naredba se završava sa MEND funkcijom i samim nazivom makro naredbe.

PROG.Podaci18, 09SEP19, 32

Obs	County	Township	Number_of_suppliers	Quantity_of_milk
45	ISTARSKA ŽUPANIJA	Brtonigla	2	208,130
46	ISTARSKA ŽUPANIJA	Buje	2	478,730
47	ISTARSKA ŽUPANIJA	Buzet	9	173,415
48	ISTARSKA ŽUPANIJA	Cerovlje	12	907,810
49	ISTARSKA ŽUPANIJA	Gračišće	2	12,367
50	ISTARSKA ŽUPANIJA	Grožnjan	3	127,151
51	ISTARSKA ŽUPANIJA	Kršan	7	506,571
52	ISTARSKA ŽUPANIJA	Lanišće	3	16,031
53	ISTARSKA ŽUPANIJA	Ližnjan	1	13,833
54	ISTARSKA ŽUPANIJA	Lupoglav	7	536,192
55	ISTARSKA ŽUPANIJA	Novigrad	1	51,650
56	ISTARSKA ŽUPANIJA	Pazin	7	388,163
57	ISTARSKA ŽUPANIJA	Pula	1	934,641
58	ISTARSKA ŽUPANIJA	Tar-Vabriga	1	241,835
59	ISTARSKA ŽUPANIJA	Umag	2	23,113
60	ISTARSKA ŽUPANIJA	Vodnjan	6	389,925
61	ISTARSKA ŽUPANIJA	Vrsar	1	208
62	ISTARSKA ŽUPANIJA	Žminj	2	83,523

ISTARSKA ŽUPANIJA

Slika 12. Prikaz rezultata za kreiranje makro naredbe iz Primjera 6.

Na Slici 12. prvo možemo vidjeti naziv iznad tablice koji sadrži informacije poput ime seta podataka koji je korišten, datuma izvođenja koda te broj makro naredbi korištenih u sesiji. Osim toga vidimo tablicu koja sadrži samo vrijednosti za Istarsku županiju zbog uvjeta u samoj makro naredbi. U FOOTNOTE-u vidimo naziv koji je naveden u marko naredbi.

4.5. WINDOW i DISPLAY funkcije

Makro naredbu WINDOW koristimo za definiranje prilagođenih prozora a kontrolira ju makro procesor. Makro naredba WINDOW definira makro prozore dok makro naredba DISPLAY prikazuje makro prozore.

Sintaksa za upotrebu WINDOW funkcije

%WINDOW prozor-ime <prozor-opcije
grupa-definicija> polje definicije;

Sintaksa za upotrebu DISPLAY funkcije

%DISPLAY prozor<.grupa > <NOINPUT > <BLANK> <BELL > <DELETE>;

Objašnjenje sintakse za upotrebu automatskih makro varijabli

&SYSVER- broj verzije SAS programa
&SYSDAY- dan u tjednu koji predstavlja dan kad se SAS sesija započinje
&SYSDATE-znakovna vrijednost koja predstavlja datum kad SAS sesija započinje (dvoznamenkasta godina)

SAS primjer 7. za upotrebu WINDOW i DISPLAY funkcije

```
%window Prozor color=red
        #5 @28 'Prozor.' attr=blink
          color=yellow
        #7 @15
          "SAS verzija &sysver on &sysday, &sysdate.."
        #12 @29 'Boja.'
           color=green;
%display Prozor;
```

U Primjeru 7. možemo vidjeti upotrebe dvije makro funkcije WINDOW i DISPLAY. WINDOW funkcija se kreira pomoću znakova # i @. Broj koji se nalazi uz # nam označava red u kojem se nalazi vrijednost, dok broj koji se nalazi uz @ označava stupac unutar prozora. Isto tako uz koordinate se nalaze vrijednosti koje će nam se ispisati u rezultatima. Osim vrijednosti izraza te koordinata gdje će se on nalazit unutar zadanog prozora još možemo uočiti da je moguće promijeniti boju samog prozora. Boje su definirane SAS programom. DISPLAY makro funkcija nam prikazuje ono što smo u WINDOW-s funkciji definirali.

Slika 13. Prikaz rezultata WINDOWS i DISPLAY funkcije iz Primjera 7.

Na Slici 13. vidimo rezultate Primjera 7. u kojem se nalazi SAS makro funkcija WINDOW. Pozadina je crvene boje što možemo vidjeti iz koda. Isto tako u kodu upotrebljavamo automatske makro varijable. Automatske makro varijable koje se koriste su za SAS verziju, SYSVER, dan u tjednu, SYSDAY, te datum, SYSDATE. One pokazuju vrijednosti samo za tu SAS sesiju. DISPLAY marko funkciju koristimo kako bi prikazali makro naredbu koju smo kreirali u WINDOW funkciji.

4.6. DO petlja i IF THEN ELSE petlja

Petlja DO označava početak dijela makro naredbe koja se tretira kao jedinica dok se ne nađe odgovarajući END izraz.

Sintaksa za upotrebu DO petlje
%DO;
tekst i makro
%END;

IF THEN ELSE je petlja koja se koristi za uspoređivanje vrijednosti makro varijabli. Naredbu IF ne možemo koristit bez THEN izraza.

Sintaksa za IF THEN ELSE petlja
%IF izraz %THEN akcija;
< %ELSE akcija;>

SAS primjer 8. za upotrebu DO petlja i IF THEN ELSE petlja

```
%let podaci=prog.podaci16;
%macro milk(a);
data podaci;
set &podaci;
%if Quantity_of_milk < &a  %then %do;
%put mu=b&a;
%end;
%else %do;
%put m=a&a;
%end;
run;
%mend milk;
%milk(25);
```

U Primjeru 8., milk makro naredba definirana je pomoću IF THEN ELSE petlje. Izvan makro naredbe imamo definiranu makro varijablu podaci koja za vrijednost ima prog.podaci16, tj. set podataka iz 2016. godine. Uz naziv makro naredbe unutar zagrada nalazi se parametar makro naredbe. U makro naredbi nalazi se IF THEN ELSE petlja koja za uvjet ima ako je varijabla Quantity_of_milk (količina predanog mlijeka) manja od vrijednosti makro varijable a onda će nam se u SAS dnevnik ispisati vrijednost mu=b&a. Ako je vrijednost tog izraza veća u SAS dnevnik će nam se ispisati vrijednost izraza m=a&a. Za vrijednost od makro varijable a SAS makro procesor će uzeti vrijednost koja se nalazi unutar zagrade kada pozivamo makro naredbu.

```
113  %let podaci=prog.podaci16;
114  %macro milk(a);
115  data podaci;
116  set &podaci;
117  %if Quantity_of_milk < &a  %then %do;
118  %put mu=b&a;
119  %end;
120  %else %do;
121  %put m=a&a;
122  %end;
123  run;
124  %mend milk;
125  %milk(25);
m=a25
```

Slika 14. Prikaz ispisa iz SAS dnevnika za upotrebu DO i IF THEN ELSE petlje iz Primjera 8.

Na Slici 14. možemo vidjeti ispis iz SAS dnevnika. Možemo vidjeti kod iz Primjera 8. a nakon njega ide rješenje IF THEN ELSE petlje. Rješenje je m=a25 što znači da nam je izraz vrijednost od makro varijable a manji od vrijednosti varijable Quantity_of_milk.

4.7. Iterativna DO petlja

Izvodi odjeljak makro naredbe ponavljajući iteracije na temelju vrijednosti indeksne varijable.

Sintaksa za iterativnu DO PETLJU

%DO makro varijabla = start %TO stop <%BY povećanje> ;
tekst i izjava makro varijable
%END;

EVAL izraz procjenjuje aritmetičke ili logičke izraze. Izraz EVAL funkcionira tako da pretvara svoj argument iz vrijednosti znaka u numerički ili logički izraz. Zatim provodi evaluaciju koda.

Sintaksa za EVAL funkciju

%EVAL (aritmetički ili logički izraz)

Korištena je i automatska makro varijabla koja se nalazi u Tablici 1.

Objašnjenje sintakse za automatsku makro varijablu

&SYSDAY- dan u tjednu koji predstavlja dan kad se SAS sesija započinje

SAS primjer 9. upotrebu DO petlje te EVAL i SYSDAY funkcije

```
%macro create(broj);
  %do i=1 %to &broj %by 2;
%let  x=%eval(&broj+&i);
  %put datum=&sysday&x;
    run;
  %end;
%mend create;
%create(25)
```

U Primjeru 9. možemo vidjeti upotrebu create makro naredbe. Unutar same makro naredbe nalaze se i SAS makro funkcija. U Primjeru 8. pored imena naredbe ,unutar zagrade, nalazi se parametar. Unutar DO petlje definiran je broj ponavljanja. Inicijalni broj ponavljanja je 25, a broj iteracija je isto 25. Uz pomoć LET funkcije definirana je x makro varijabla koja za vrijednost ima rješenje EVAL funkcije. EVAL funkcija nam rješava izraz unutar zagrade. Unutar zagrade se nalazi aritmetički izraz zbrajanja dviju makro varijabli. U SAS dnevniku će nam se ispisati makro varijabla datum koja će za vrijednost imati automatsku makro varijablu SYSDAY te makro varijablu x, koju smo sami kreirali.

```
416  %macro create(broj);
417     %do i=1 %to &broj %by 2;
418  %let  x=%eval(&broj+&i);
419     %put datum=&sysday&x;
420        run;
421     %end;
422  %mend create;
423  %create(25)
datum=Friday26
datum=Friday28
datum=Friday30
datum=Friday32
datum=Friday34
datum=Friday36
datum=Friday38
datum=Friday40
datum=Friday42
datum=Friday44
datum=Friday46
datum=Friday48
datum=Friday50
```

Slika 15. Prikaz SAS dnevnika DO petlje i EVAL funkcije iz Primjera 9.

Na Slici 15. je prikaz SAS dnevnika na kojem se vidi rezultat koji smo dobili nakon rada makro naredbe. Rezultat je u obliku koji je definiran PUT funkcijom. DO petlja je napravila iteraciju od vrijednosti broja 25, koji je bio zadan, do broja 50. Broj iteracija je vrijednost makro varijable broj. SAS makro procesor riješio je izraz te za konačno rješenje ispisao vrijednost SYSDAY automatske makro varijable te vrijednost makro varijable broj s uvećanjem zbog iteracija.

4.8. DO UNTIL petlja i EVAL funkcija

DO UNTIL petlja radi ponavljanja dok ne dođe na vrijednost koja je definirana funkcijom. Nakon svakog ponavljanja provjerava vrijednost te radi iteracije dok ne postigne vrijednost definiranu funkcijom.

Sintaksa za DO UNTIL petlja

%DO %UNTIL (izjava);
tekst i definicija makro jezika
%END;

SAS primjer 10. za upotrebu DO UNTIL petlje i EVAL funkcije

```
%let podaci=prog.podaci16;
%let podaci1=prikaz16;
%macro test(i);
data &podaci1;
```

```
set &podaci;
%do %until (&i<Number_of_suppliers);
%put Vrijednost &i ;
%let i=%eval(&i+1);
%end;
run;
%mend test;
%test (58);
```

Pomoću LET funkcije smo definirali makro varijable koje pozivamo u DATA i SET koraku. Makro varijabla podaci se odnosi na podatke iz 2016. godine. U primjeru možemo vidjeti kako je uz pomoć MACRO funkcije kreirana makro naredba test. Makro naredba ima i parametar pored imena makro naredbe unutar zagrade. Unutar kreirane makro naredbe nalazi se njezina definicija koja je čini učinkovitijom. DO UNTIL petlja nam radi ponavljanja sve dok je vrijednost makro varijable i manja od vrijednosti varijable Number_of_suppliers (broj predavača mlijeka). Makro varijabla i nam je definirana unutar makro naredbe. Vrijednost makro varijable i je rješenje EVAL funkcije, izraza koji se nalazi unutar zagrade nakon EVAL funkcije. Rješenje izraza je uvećanje makro varijabla i za 1. Ona će se povećavati sve dok je sama vrijednost makro varijable i manja od vrijednosti varijable Number_of_suppliers.

```
131  %let podaci=prog.podaci16;
132  %let podaci1=prikaz16;
133  %macro test(i);
134  data &podaci1;
135  set &podaci;
136  %do %until (&i<Number_of_suppliers);
137  %put Vrijednost &i ;
138  %let i=%eval(&i+1);
139  %end;
140  run;
141  %mend test;
142  %test (58);
Vrijednost 58

NOTE: There were 387 observations read from the data set PROG.PODACI16.
NOTE: The data set WORK.PRIKAZ16 has 387 observations and 4 variables.
NOTE: DATA statement used (Total process time):
      real time           0.01 seconds
      cpu time            0.00 seconds
```

Slika 16.Prikaz SAS dnevnika za upotrebu DO UNTIL petlje i EVAL funkcije u Primjeru 10.

Iz ispisa SAS dnevnika na Slici 16. možemo vidjeti kako je vrijednost makro varijable i manja od vrijednosti varijable Number_of_suppliers. Zbog toga u SAS dnevniku nije ispisana nijedna iteracija. Ispisana je samo vrijednost makro varijable i. Ispis vrijednost makro varijable i je definiran pomoću PUT funkcije.

4.9. DO WHILE petlja i SYSEVALF funkcija

DO WHILE petlja radi iteracije dokle god je izraz uz nju istinit. Ako je izraz lažan kad ga prvi put testira makro procesor, petlja DO WHILE se više ne ponavlja.

Sintaksa za upotrebu DO WHILE petlje

```
%DO %WHILE (izjava);
tekst i izjava makro jezika
%END;
```

SYSEVALF je jedina makro funkcija koja može procijeniti logičke izraze koji sadrže vrijednosti s pomičnom točkom ili nedostajuće vrijednosti. Izraz SYSEVALF izvodi aritmetiku s pomičnim zarezom i vraća vrijednost koja je formatirana pomoću BEST32. formata.

Sintaksa za upotrebu SYSEVAL funkcije

```
&SYSEVALF(izraz<, tip pretvorbe>)
```

SAS primjer 11. za upotrebu DO WHILE petlje i SYSEVAL funkcije

```
%let podaci=prog.podaci16;
%let podaci1=prikaz16;
%macro iteracije(i);
data &podaci1;
set &podaci;
%do %while (&i<Number_of_suppliers);
%put Broj iteracija &i ;
%let i=%sysevalf(&i+50.25,ceil);
%end;
run;
%mend iteracije;
%iteracije (50);
```

U Primjeru 11. imamo kreiranu makro naredbu, a izvan nje imamo kreirane makro varijable. Makro varijable koje su kreirane izvan makro naredbe se odnose na nazive podataka koje koristimo u primjeru i na nazive podataka koje spremamo nakon rada makro naredbe. Makro varijabla podaci se odnosi na podatke za 2016. godinu. Makro naredba ima i parametar pored imena makro naredbe. Unutar makro naredbe imamo petlju DO WHILE koja radi iteracije sve dok je vrijednost izraza istinita. Izraz je u primjeru definiran unutar zagrada pored DO UNTIL petlje. Vrijednost izraza je da makro varijabla i mora biti manja od varijable Number_of_suppliers (broja predavača mlijeka). Vrijednost makro varijable i definirana je unutar same makro naredbe, tako što je stavljena za pozicijski parametar.

SYSEVALF funkcija nam rješava izraz koji se nalazi unutar zagrade. Osim izraza u zagradi se nalazi i opcija same funkcije koja nam decimalne brojeve zaokruži na prvi veći cijeli broj.

```
126  %let podaci=prog.podaci16;
127  %let podaci1=prikaz16;
128  %macro iteracije(i);
129  data &podaci1;
130  set &podaci;
131  %do %while (&i<Number_of_suppliers);
132  %put Broj iteracija &i ;
133  %let i=%sysevalf(&i+50.25,ceil);
134  %end;
135  run;
136  %mend iteracije;

137  %iteracije (50);
Broj iteracija 50
Broj iteracija 101
Broj iteracija 152
Broj iteracija 203
Broj iteracija 254
Broj iteracija 305
Broj iteracija 356
Broj iteracija 407
Broj iteracija 458
Broj iteracija 509
Broj iteracija 560
Broj iteracija 611
Broj iteracija 662
Broj iteracija 713
Broj iteracija 764
Broj iteracija 815
Broj iteracija 866
Broj iteracija 917
Broj iteracija 968
Broj iteracija 1019
Broj iteracija 1070
Broj iteracija 1121
Broj iteracija 1172
Broj iteracija 1223
```

Slika 17. Ispis iz SAS dnevnika za upotrebu DO WHILE petlje i SYSEVALF funkcije iz Primjera 11.

Na Slici 17. se nalazi ispis iz SAS dnevnika za Primjer 11. na kojoj možemo vidjeti dio iteracija. Iteracije počinju od vrijednosti 50 jer smo to definirali u makro naredbi kao pozicijski parametar. SAS makro procesor će raditi iteracije sve dok je izraz istinit. Odnosno dok je vrijednost makro varijable i manja od vrijednosti varijable Number_of_suppliers (broj predavača mlijeka).

4.10. GLOBAL i LOCAL funkcije

Funkcija GLOBAL stvara jednu ili više globalnih makro varijabli. Globalne makro varijable su varijable koje su dostupne tijekom izvođenja SAS sesije.

Sintaksa za GLOBAL funkciju

%GLOBAL makro varijabla-1 <makro varijabla-n>;

Funkcija LOCAL stvara jednu ili više lokalnih makro varijabli. Lokalne makro varijable su varijable koje su dostupne samo tijekom izvršavanja makro naredbe u kojoj su definirane. Upotrebom funkcija LOCAL osiguravamo da se makro varijable kreirane ranije u programu ne bi zamijenile vrijednostima dodijeljenim za iste varijablama u trenutnoj makro naredbi.

Sintaksa za LOCAL funkciju

%LOCAL makro varijabla-1 <makro varijabla-n>;

SAS primjer 12. za upotrebu LOCAL funkcije

```
%let var=1;
%macro loglo;
  %put ***** Lokal *****;
  %local var;
  %let var=2;
  %put Vrijednost unutar loglo je &var;
  %put ***** Global *****;
%mend loglo;
%loglo
```

Na Primjeru 12. je objašnjeno djelovanje LOCAL funkcije. Izvan makro naredbe kreirana je makro varijabla var kojoj je dodijeljena vrijednost jedan. Unutar makro naredbe imamo definiranu makro varijablu var koja ima vrijednost dva. To je bitno jer kada unutar makro naredbe stavimo funkciju LOCAL to znači da je vrijednost makro varijabli definirana samo unutar te naredbe. Osim toga u makro naredbi imamo definiran izgled rješenja koje će biti ispisano u SAS dnevniku. Izgled rješenja je definiran pomoću PUT funkcije.

```
198  %macro loglo;
199      %put ***** Lokal *****;
200      %local var;
201      %let var=2;
202      %put Vrijednost unutar loglo je &var;
203      %put ***** Global *****;
204  %mend loglo;
205  %loglo
***** Lokal *****
Vrijednost unutar loglo je 2
***** Global *****
```

Slika 18. Ispis iz SAS dnevnika za upotrebu LOCAL funkcije iz Primjera 12.

Slika 18. prikazuje ispis iz SAS dnevnika nakon što smo učitali kod iz Primjera 12. Osim koda koji je učitan možemo vidjeti ispod njega rješenje tog koda. Rješenje se sastoji od izraza koji su definirani sa PUT funkcijom. Vidimo kako je unutar rješenja makro varijabla te da je njena vrijednost dva. Da je korištena GLOBAL funkcija vrijednost od makro varijable var bi bila dostupna i izvan makro naredbe.

4.11. STR i UPCASE funkcije

Funkcija STR maskira niz znakova tijekom definiranja makro naredbe.

Sintaksa za STR funkciju
%STR (niz znakova)

Funkcija UPCASE pretvara mala slova u velika.

Sintaksa za UPCASE funkciju
%UPCASE (niz znakova | tekstualni izraz)

SAS primjer 13. za upotrebu STR i UPCASE funkcije

```
%let varname=County;
%macro runn(varname);
data mu&varname. muu&varname.;
set prog.podaci16;
  %if %upcase(&varname.)= "ISTARSKA ŽUPANIJA"  %then
     %str(proc print data=mu&varname.; run;);
  %else %str(proc print data=muu&varname.; run;);
%mend runn;
%runn(varname=County)
```

U Primjeru 13. kreirana je makro varijabla varname a njezina vrijednost je county. Nakon makro varijable imamo makro naredbu koja je definirana IF THEN ELSE petljom. U IF THEN ELSE petlji imamo dvije dodatne funkcije UPCASE i STR. UPCASE funkcija se odnosi na sadržaj u zagradi, te makro varijablu varname koja ima vrijednost county. U IF THEN ELSE petlji se nalazi izraz sa UPCASE funkcijom. Ako je vrijednost makro varijable county jednaka „ISTARSKOJ ŽUPANIJI“, IF petlja će imati istinit izraz. STR funkcija maskira & simbol unutar zagrade te ga makro procesor ignorira. Podaci će biti spremljeni u datoteku sa nazivom mu&County ako je izraz istinit. Ako izraz nije istinit podaci će se spremiti u datoteku sa nazivom muu&County.

```
169  %let varname=County;
170  %macro runn(varname);
171  data mu&varname. muu&varname.;
172  set prog.podaci16;
173     %if %upcase(&varname.)= "ISTARSKA ŽUPANIJA"  %then
174         %str(proc print data=mu&varname.; run;);
175     %else %str(proc print data=muu&varname.; run;);
176  %mend runn;
177  %runn(varname=County)

NOTE: There were 387 observations read from the data set PROG.PODACI16.
NOTE: The data set WORK.MUCOUNTY has 387 observations and 4 variables.
NOTE: The data set WORK.MUUCOUNTY has 387 observations and 4 variables.
NOTE: DATA statement used (Total process time):
      real time           0.09 seconds
      cpu time            0.01 seconds


NOTE: There were 387 observations read from the data set WORK.MUUCOUNTY.
NOTE: PROCEDURE PRINT used (Total process time):
      real time           0.15 seconds
      cpu time            0.14 seconds
```

Slika 19. Ispis iz SAS dnevnika za upotrebu UPCASE i STR funkcije iz Primjera 13.

Slika 19. prikazuje nam ispis iz SAS dnevnika iz kojeg možemo vidjeti kako je SAS učitao makro naredbu te da nam je spremio dvije izlazne datoteke jer nam je izraz koji se nalazi uz UPCASE funkciju bio istinit.

4.12. LENGTH i SUBSTR funkcije

Funkcija LENGTH vraća duljinu niza, ako je argument niz znakova. Ako je argument tekstualni izraz, LENGTH vraća numeričku vrijednost izraza.

Sintaksa za LENGTH funkciju

%LENGTH (niz znakova | tekstualni izraz)

Funkcija SUBSTR skraćuje izraz, počevši od položaja, za dužinu broja znakova.

Sintaksa za SUBSTR funkciju

%SUBSTR (argument, pozicija<, dužina>)

PROC FREQ je procedura koja kreira tablice s frekvencijama podataka, postotkom, kumulativnom frekvencijom i kumulativnim postotkom.

SAS primjer 14. za upotrebu LENGTH i SUBSTR funkcije

```
%let varname=county;
%macro make(varname);
proc freq data=prog.podaci18;
```

```
tables &varname.;
  %put a=%length(&varname.);
   %put var = %substr(&varname.,1,3);
  run;
%mend make;
%make(varname=county)
```

U Primjeru 14. imamo makro naredbu make. Izvan makro naredbe definirana je makro varijabla varname koja ima vrijednost county. Unutar makro naredbe je procedura PROC FREQ koja nam na setu podataka za 2018. godinu kreira tablicu s obzirom na makro varijablu varname. Kada makro procesor riješi makro varijablu, umjesto izraza varname uzima vrijednost county. U SAS dnevniku ispisuju se vrijednosti za makro varijablu a i var. Makro varijabla a definirana je pomoću PUT funkcije, a vrijednost njezina izraza je definirana LENGTH funkcijom. LENGTH funkcija izražava numeričku vrijednost izraza u zagradi. Makro varijabla var ima vrijednost izraženu pomoću SUBSTR funkcije koja nam skraćuje riječ unutar zagrade. U zagradi osim riječi na koju se odnosi SUBSTR funkcija nalaze se i opcije koje nam označavaju od kojeg do kojeg znaka nam se treba skratiti riječ.

```
178  %let varname=county;
179  %macro make(varname);
180  proc freq data=prog.podaci18;
181  tables &varname.;
182     %put a=%length(&varname.);
183      %put var = %substr(&varname.,1,3);
184     run;
185  %mend make;
186  %make(varname=county)
a=6
var = cou

NOTE: There were 336 observations read from the data set PROG.PODACI18.
NOTE: PROCEDURE FREQ used (Total process time):
      real time           0.25 seconds
      cpu time            0.07 seconds
```

Slika 20. Ispis iz SAS dnevnika za upotrebu LENGTH i SUBSTR funkcije iz Primjera 14.

Slika 20. prikazuje ispis iz SAS dnevnika u kojem možemo vidjeti učitanu makro naredbu make. Osim makro naredbe vidimo i rješenja makro varijabli koja su definirana pomoću LENGTH i SUBSTR funkcije. Makro varijabla a koja je bila definirana pomoću LENGHT funkcije dobila je vrijednost 6, a makro varijabla koje je definirana pomoću SUBSTR funkcije dobila je vrijednost cou.

The FREQ Procedure

County	Frequency	Percent	Cumulative Frequency	Cumulative Percent
BJELOVARSKO-BILOGORSKA ŽUPANIJA	23	6.85	23	6.85
BRODSKO-POSAVSKA ŽUPANIJA	20	5.95	43	12.80
GRAD ZAGREB	1	0.30	44	13.10
ISTARSKA ŽUPANIJA	18	5.36	62	18.45
KARLOVAČKA ŽUPANIJA	17	5.06	79	23.51
KOPRIVNIČKO-KRIŽEVAČKA ŽUPANIJA	25	7.44	104	30.95
KRAPINSKO-ZAGORSKA ŽUPANIJA	24	7.14	128	38.10
LIČKO-SENJSKA ŽUPANIJA	8	2.38	136	40.48
MEĐIMURSKA ŽUPANIJA	19	5.65	155	46.13
OSJEČKO-BARANJSKA ŽUPANIJA	39	11.61	194	57.74
POŽEŠKO-SLAVONSKA ŽUPANIJA	10	2.98	204	60.71
PRIMORSKO-GORANSKA ŽUPANIJA	1	0.30	205	61.01
SISAČKO-MOSLAVAČKA ŽUPANIJA	16	4.76	221	65.77
SPLITSKO-DALMATINSKA ŽUPANIJA	8	2.38	229	68.15
VARAŽDINSKA ŽUPANIJA	25	7.44	254	75.60
VIROVITIČKO-PODRAVSKA ŽUPANIJA	16	4.76	270	80.36
VUKOVARSKO-SRIJEMSKA ŽUPANIJA	29	8.63	299	88.99

Slika 21. Prikaz rezultata PROC FREQ procedure iz Primjera 14.

Na Slici 21. vidimo prikaz PROC FREQ procedure. PROC FREQ procedura je kreirala tablicu na osnovu varijable koja sadrži imena županija. Unutar tablice se nalaze vrijednosti frekvencije, postotaka, kumulativnu frekvencije i kumulativnog postotak. Vrijednosti se odnose na broj opažanja za varijablu County.

5. Usporedba DATA koraka i makro programa

U ovom poglavlju prikazana je usporedba DATA koraka i makro programa. Prije svega važno je naglasiti da se makro varijable, naredbe i funkcije rješavaju prije samog pokretanja koda te nam to znači da se izvršavaju prije DATA i PROC koraka. Makro naredbe i varijable nam kreiraju dijelove programa koji se brzo i učinkovito ponavljaju. Modularna struktura programa omogućuje da se dio SAS koda koji se ponavlja može generirati i u SAS makro programu.

SAS primjer 15. koda sa PROC korakom

```
proc means data=prog.podaci15;
title "Proizvodnja mlijeka za 15";
class County;
var Number_of_suppliers Quantity_of_milk;
run;

proc means data=prog.podaci16;
title "Proizvodnja mlijeka za 16";
class County;
var Number_of_suppliers Quantity_of_milk;
run;

proc means data=prog.podaci17;
title "Proizvodnja mlijeka za 17";
class County;
var Number_of_suppliers Quantity_of_milk;
run;

proc means data=prog.podaci18;
title "Proizvodnja mlijeka za 18";
class County;
var Number_of_suppliers Quantity_of_milk;
run;
```

U Primjeru 15. možemo vidjeti četiri koda. Kodovi se koriste za opisnu statistiku, tj. naredbe mean, max, min i std. PROC MEANS procedura će podatke klasificirati s obzirom na varijablu County (županija) te napraviti analizu podataka za varijablu Number_of_suppliers (broj predavača mlijeka) i Quantity_of_milk (količine mlijeka). Tablice sa rezultatima će imati naziv s obzirom na godinu iz koje su podaci.

SAS primjer 16. za prikaz upotrebe makro naredbe

```
%macro podaci;
%do godine=15 %to 18;
proc means data=prog.podaci&godine;
title "Proizvodnja mlijeka za &godine";
class County;
var Number_of_suppliers Quantity_of_milk;
run;
%end;
%mend podaci;
%podaci
```

Kod iz Primjer 16. daje iste rezultate kako kod u Primjeru 15. Ali kod u Primjeru 16. je kraći te jednostavniji. Umjesto da za svaku PROC MEANS proceduru radimo poseban kod moguće je kreirati jedan kod kao u Primjeru 16. U makro naredbi možemo vidjeti makro varijablu godine koja sa DO petljom ima definiranu vrijednost od 15 do 18. Kada je definirana makro varijabla, ista je pozvana u PROC MEANS koraku. PROC MEANS procedura radi statističku analizu na podacima prema klasifikaciji s obzirom na županiju, a analizu radi na varijablama koje sadrže vrijednosti za broj predavača mlijeka i količinu mlijeka koja je predana. Za svaki set podataka SAS će kreirati poseban ispis. Svaki ispis imat će naziv s obzirom iz koje su godine podaci korišteni u analizi. Važno je napomenuti da je bitna upotreba dvostrukih navodnika kod kreiranja naslova kako bi SAS makro procesor prepoznao simbol makro varijable.

Proizvodnja mlijeka za 15 — 15
08:10 Friday, September 27, 2019

The MEANS Procedure

County	N Obs	Variable	Minimum	Maximum
OSJEČKO-BARANJSKA ŽUPANIJA	39	Quantity_of_milk	27957.00	37376179.00
POŽEŠKO-SLAVONSKA ŽUPANIJA	10	Number_of_suppliers	8.0000000	36.0000000
		Quantity_of_milk	614158.00	2535533.00
PRIMORSKO-GORANSKA ŽUPANIJA	6	Number_of_suppliers	1.0000000	5.0000000
		Quantity_of_milk	2010.00	90010.00
SISAČKO-MOSLAVAČKA ŽUPANIJA	18	Number_of_suppliers	1.0000000	110.0000000
		Quantity_of_milk	12951.00	2971291.00
SPLITSKO-DALMATINSKA ŽUPANIJA	10	Number_of_suppliers	1.0000000	8.0000000
		Quantity_of_milk	348.0000000	416297.00
VARAŽDINSKA ŽUPANIJA	25	Number_of_suppliers	1.0000000	44.0000000
		Quantity_of_milk	2193.00	1310197.00
VIROVITIČKO-PODRAVSKA ŽUPANIJA	16	Number_of_suppliers	1.0000000	111.0000000
		Quantity_of_milk	86308.00	3841106.00
VUKOVARSKO-SRIJEMSKA ŽUPANIJA	29	Number_of_suppliers	1.0000000	100.0000000
		Quantity_of_milk	19716.00	17383802.00
ZADARSKA ŽUPANIJA	5	Number_of_suppliers	1.0000000	5.0000000
		Quantity_of_milk	31068.00	3172447.00
ZAGREBAČKA ŽUPANIJA	32	Number_of_suppliers	1.0000000	108.0000000
		Quantity_of_milk	17339.00	4795432.00
ŠIBENSKO-KNINSKA ŽUPANIJA	8	Number_of_suppliers	1.0000000	18.0000000
		Quantity_of_milk	32649.00	245838.00

Slika 22.Prikaz rezultata PROC MEANS procedure za podatke iz 2015. godine iz Primjera 15. i Primjera 16.

Proizvodnja mlijeka za 16 16
08:10 Friday, September 27, 2019

The MEANS Procedure

County	N Obs	Variable	N	Mean	Std Dev
BJELOVARSKO-BILOGORSKA ŽUPANIJA	23	Number_of_suppliers	23	73.0000000	45.4882602
		Quantity_of_milk	23	3271026.00	2679248.08
BRODSKO-POSAVSKA ŽUPANIJA	22	Number_of_suppliers	22	11.0000000	12.5508490
		Quantity_of_milk	22	596136.45	829923.10
GRAD ZAGREB	1	Number_of_suppliers	1	33.0000000	.
		Quantity_of_milk	1	1819965.00	.
ISTARSKA ŽUPANIJA	21	Number_of_suppliers	21	4.4761905	4.1063250
		Quantity_of_milk	21	267627.62	301067.10
KARLOVAČKA ŽUPANIJA	17	Number_of_suppliers	17	21.1764706	27.0768243
		Quantity_of_milk	17	1007106.29	1155780.91
KOPRIVNIČKO-KRIŽEVAČKA ŽUPANIJA	25	Number_of_suppliers	25	72.0400000	72.8339893
		Quantity_of_milk	25	2394323.64	2625074.50
KRAPINSKO-ZAGORSKA ŽUPANIJA	27	Number_of_suppliers	27	12.2962963	14.8345097
		Quantity_of_milk	27	279865.11	294493.56
LIČKO-SENJSKA ŽUPANIJA	9	Number_of_suppliers	9	56.1111111	70.0507752
		Quantity_of_milk	9	604868.22	616327.57
MEĐIMURSKA ŽUPANIJA	22	Number_of_suppliers	22	10.2727273	10.5567934
		Quantity_of_milk	22	721449.68	518535.82
OSJEČKO-BARANJSKA ŽUPANIJA	39	Number_of_suppliers	39	14.5128205	10.3029679
		Quantity_of_milk	39	3765685.90	6919124.09
POŽEŠKO-SLAVONSKA ŽUPANIJA	10	Number_of_suppliers	10	20.7000000	8.1928153
		Quantity_of_milk	10	1005526.40	517122.82
PRIMORSKO-GORANSKA ŽUPANIJA	5	Number_of_suppliers	5	3.2000000	2.0493902
		Quantity_of_milk	5	42622.20	35523.38

Slika 23.Prikaz rezultata PROC MEANS procedure za podatke iz 2016. godine iz Primjera 15. i Primjera 16.

Proizvodnja mlijeka za 17 19
08:10 Friday, September 27, 2019

The MEANS Procedure

County	N Obs	Variable	N	Mean	Std Dev
BJELOVARSKO-BILOGORSKA ŽUPANIJA	23	Number_of_suppliers	23	61.3478261	39.0077017
		Quantity_of_milk	23	3361378.78	2714533.31
BRODSKO-POSAVSKA ŽUPANIJA	20	Number_of_suppliers	20	10.1500000	12.1537302
		Quantity_of_milk	20	645630.75	854145.56
GRAD ZAGREB	1	Number_of_suppliers	1	30.0000000	.
		Quantity_of_milk	1	1641627.00	.
ISTARSKA ŽUPANIJA	19	Number_of_suppliers	19	4.1052632	3.6952633
		Quantity_of_milk	19	310868.68	391184.21
KARLOVAČKA ŽUPANIJA	17	Number_of_suppliers	17	18.0588235	22.2442987
		Quantity_of_milk	17	943409.47	1040378.16
KOPRIVNIČKO-KRIŽEVAČKA ŽUPANIJA	25	Number_of_suppliers	25	63.8400000	68.4931140
		Quantity_of_milk	25	2379458.00	2691121.03
KRAPINSKO-ZAGORSKA ŽUPANIJA	24	Number_of_suppliers	24	9.5000000	10.3587812
		Quantity_of_milk	24	296879.33	300361.85
LIČKO-SENJSKA ŽUPANIJA	9	Number_of_suppliers	9	47.8888889	60.2296531
		Quantity_of_milk	9	546675.78	618163.47
MEĐIMURSKA ŽUPANIJA	21	Number_of_suppliers	21	8.8571429	8.8050310
		Quantity_of_milk	21	732336.86	509167.24
OSJEČKO-BARANJSKA ŽUPANIJA	39	Number_of_suppliers	39	12.7179487	9.1391256
		Quantity_of_milk	39	3660966.21	6826420.69
POŽEŠKO-SLAVONSKA ŽUPANIJA	10	Number_of_suppliers	10	16.5000000	7.1995370
		Quantity_of_milk	10	945618.90	540628.27
PRIMORSKO-GORANSKA ŽUPANIJA	5	Number_of_suppliers	5	1.8000000	1.3038405
		Quantity_of_milk	5	23761.00	21478.09

Slika 24. Prikaz rezultata PROC MEANS procedure za podatke iz 2017. godine iz Primjera 15. i Primjera 16.

Proizvodnja mlijeka za 18 22
08:10 Friday, September 27, 2019

The MEANS Procedure

County	N Obs	Variable	N	Mean	Std Dev
BJELOVARSKO-BILOGORSKA ŽUPANIJA	23	Number_of_suppliers	23	54.7391304	34.6163616
		Quantity_of_milk	23	3230729.13	2705801.38
BRODSKO-POSAVSKA ŽUPANIJA	20	Number_of_suppliers	20	8.9500000	10.8796043
		Quantity_of_milk	20	603114.10	810611.17
GRAD ZAGREB	1	Number_of_suppliers	1	24.0000000	.
		Quantity_of_milk	1	1592433.00	.
ISTARSKA ŽUPANIJA	18	Number_of_suppliers	18	3.8333333	3.3121878
		Quantity_of_milk	18	282960.44	296182.18
KARLOVAČKA ŽUPANIJA	17	Number_of_suppliers	17	16.4117647	19.3909090
		Quantity_of_milk	17	938498.82	1114508.90
KOPRIVNIČKO-KRIŽEVAČKA ŽUPANIJA	25	Number_of_suppliers	25	56.7600000	63.7908562
		Quantity_of_milk	25	2302918.16	2639496.99
KRAPINSKO-ZAGORSKA ŽUPANIJA	24	Number_of_suppliers	24	8.6250000	9.0880356
		Quantity_of_milk	24	283271.21	308803.70
LIČKO-SENJSKA ŽUPANIJA	8	Number_of_suppliers	8	49.0000000	56.5331761
		Quantity_of_milk	8	611329.75	650925.55
MEĐIMURSKA ŽUPANIJA	19	Number_of_suppliers	19	8.7368421	8.1774901
		Quantity_of_milk	19	706851.89	517928.70
OSJEČKO-BARANJSKA ŽUPANIJA	39	Number_of_suppliers	39	11.2564103	8.3594194
		Quantity_of_milk	39	3424691.28	6445975.31
POŽEŠKO-SLAVONSKA ŽUPANIJA	10	Number_of_suppliers	10	14.0000000	6.9761498
		Quantity_of_milk	10	931373.00	515696.15
PRIMORSKO-GORANSKA ŽUPANIJA	1	Number_of_suppliers	1	1.0000000	.
		Quantity_of_milk	1	25457.00	.

Slika 25. Prikaz rezultata PROC MEANS procedure za podatke iz 2018. godine iz Primjera 15. i Primjera 16.

Na Slikama od 22. do 25. je prikaz dijela rezultata iz Primjera 15. i 16. Kada definiramo kod iz Primjera 16. mi ga iznova možemo koristiti ako učestalo trebamo raditi statističku analizu na sličnim podacima. Ako dobijemo nove podatke kod možemo generirati i iznova koristiti. U Primjeru 15. za svaki novi set podataka moramo iznova pisati kod. Na slikama vidimo PROC MEANS proceduru i opisnu statistiku. Klasifikacija podataka je napravljena na osnovu varijable sa županijama. Statistička analiza je napravljena na varijablama sa brojem predavača mlijeka i količine predanog mlijeka. Analiza sadrži vrijednosti srednje, maksimalne, minimalne i standardne devijacije za količinu mlijeka i broj predavača mlijeka po županijama za godine od 2015. do 2018. godine.

U nastavku teksta prikazan je još jedan primjer usporedbe DATA koraka i makro programa. Prije izvođenja samih primjera važno je napomenuti da je podacima dodana varijabla countycode koja nam ima vrijednost od 1 do 20. Countycode za svaku vrijednost županije ima dodijeljen jedan broj od 1 do 20.

SAS primjer 17. za prikaz DATA koraka

```
data Countycode1 Countycode2 Countycode3 Countycode4 Countycode5 Countycode6
Countycode7 Countycode8 Countycode9 Countycode10 Countycode11 Countycode12
Countycode13 Countycode14 Countycode15 Countycode16 Countycode17 Countycode18
```

```
Countycode19 Countycode20
;
set prog.Podaci15n;

if Countycode=1 then output Countycode1;
else if Countycode=2 then output Countycode2;
else if Countycode=3 then output Countycode3;
else if Countycode=4 then output Countycode4;
else if Countycode=5 then output Countycode5;
else if Countycode=6 then output Countycode6;
else if Countycode=7 then output Countycode7;
else if Countycode=8 then output Countycode8;
else if Countycode=9 then output Countycode9;
else if Countycode=10 then output Countycode10;
else if Countycode=11 then output Countycode11;
else if Countycode=12 then output Countycode12;
else if Countycode=13 then output Countycode13;
else if Countycode=14 then output Countycode14;
else if Countycode=15 then output Countycode15;
else if Countycode=16 then output Countycode16;
else if Countycode=17 then output Countycode17;
else if Countycode=18 then output Countycode18;
else if Countycode=19 then output Countycode19;
else if Countycode=20 then output Countycode20;

run;
```

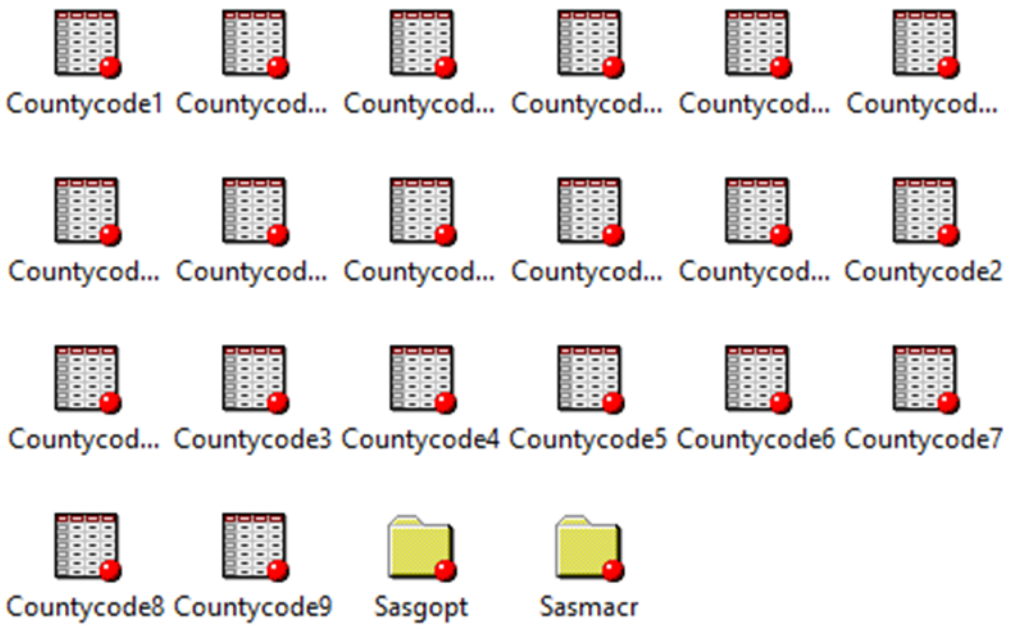
U Primjeru 17. imamo prikaz koda koji nam je definiran pomoću IF THEN ELSE petlje. Kod nam radi izlazne datoteke s obzirom koju vrijednost varijabla Countycode ima. S obzirom na vrijednost Countycode varijable IF THEN ELSE petlja kreira ime datoteci. Što znači ako je vrijednost Countycoda jednaka 1 izlazna datoteka će nam biti countycode1. Na isti način kreirat će se i datoteke od broja 2 do 20.

SAS primjer 18. za prikaz upotrebe makro naredbe

```
%macro Countycoden;
data
%do i=1 %to 20;
Countycode&i
%end;
;
set prog.Podaci15n;
if Countycode=1 then output Countycode1;
```

```
%do i=2 %to 20;
else if Countycode=&i then output Countycode&i;
%end;
run;
%mend Countycoden;
%Countycoden
```

Primjer 18. nam kreira izlazne datoteke koje su iste kao i u Primjer 17. Kada kreiramo makro naredbu možemo je iznova koristit i prilagođavati. U Primjeru 18. možemo vidjeti kreiranu makro varijablu i, kojoj smo dodijelili vrijednost od 1 do 20. Vrijednost makro varijable i stavili smo uz naziv varijable Countycode. Spojili smo vrijednost makro varijable i te naziv Countycode. Definirali smo unutar makro naredbe da ako Counticode ima vrijednost makro varijable i jedan da nam napravi izlaznu datoteku Countycode1. Za sve ostale Countycode vrijednost makro varijabla i će imati vrijednost od 2 do 20. Unutar ELSE petlje nalazi nam se definicija ostalih izlaznih datoteka, koje su definirane pomoću makro varijable i. Definicija ostalih izlaznih datoteka glasi ako je Countycode&i onda neka izlazna datoteka bude Countycode&i, tako će ispisati za sve preostale vrijednosti makro varijable i od 2 do 20.



Slika 26. Prikaz izlaznih datoteka iz Primjera 17. i Primjera 18.

Na Slici 26. imamo dvadeset izlaznih datoteka koje su stvorene nakon primjene kodova iz Primjera 17. i Primjera 18.

6. Zaključak

Makro programi ili makro jezik sadrže makro naredbe, makro varijable i makro funkcije. Makro naredbe, makro varijable i makro funkcije komuniciraju sa makro procesorom unutar SAS programa. Makro programi izvode se prije običnog koda u DATA i PROC koraku SAS programa. Simbol koji se koristi za pozivanje makro varijable je &, a makro naredbe je %. Automatske makro varijable kreira sam SAS makro procesor koji daje vrijednosti za tu SAS sesiju. Moguće je i kreiranje vlastitih makro varijabli. Kreiranim makro varijablama moguće je i promijeniti njihove vrijednosti. Makro varijablu je moguće kreirati unutar makro naredbe i izvan makro naredbe, osim u DATALINES koraku. Makro naredbe se kreiraju pomoću MACRO i MEND funkcije. Kada kreiramo makro naredbu ona je definirana samo za trenutnu SAS sesiju. SAS makro naredbe nam ubrzavaju analize i procedure koje moramo učestalo izvoditi te su dobra alternativa klasičnom DATA koraku radi mogućnosti koje pružaju u vidu fleksibilnijih kodova koji se mogu generirati.

SAS makro programiranje uvelike smanjuje potreban trud za programiranje SAS kodova. SAS makro programiranje nam je korisno kod analiza i procedura koje trebamo učestalo izvoditi. Omogućava nam kreiranje fleksibilnog koda koji se može unutar makro programa sam generirati. U ovom radu prikazani su primjeri iz stočarske prakse u okviru kojih se mogu koristiti makro programi na učinkovit način.

7. Popis literature

1. Burlew M. M. (2006). SAS® Macro Programming Made Easy, Second Edition. Cary, NC: SAS Institute Inc.
2. Carpenter A. (1998). Carpenter's Complete Guide to the SAS Macro Language. SAS Institute, Cary, NC.
3. Delwiche L. D., Slaughter S. J. (2003). The Little SAS Book: A Primer, Third Edition. SAS Institute, Cary, NC.
4. Delwiche, Lora D. and Susan J. Slaughter (2004). SAS® Macro Programming for Beginners. Proceedings of the Twenty-Ninth Annual SAS® Users Group International Conference. Cary, NC: SAS Institute Inc. Paper 243-29, 9. – 12.05.2004, Montreal, Canada, str.1-13.
5. SAS Institute Inc. (2016). SAS® 9.4 Macro Language: Reference, Fifth Edition. Cary, NC: SAS Institute Inc.

Popis slika

Slika 1. Prikaz kako radi makro procesor (Carpenter, 2004) 2
Slika 2. Prikaz rezultata korištenja LET funkcije 10
Slika 3. Prikaz rezultata PROC MEANS procedure iz Primjera 2. 11
Slika 4. Prikaz rezultata za podatke iz 2015.godine iz Primjera 3. 12
Slika 5. Prikaz rezultata iz 2016.godine iz Primjera 3. 13
Slika 6. Prikaz rezultata PROC COMPARE procedure iz Primjera 3. 13
Slika 7. Prikaz rezultata PROC COMPARE procedure iz Primjera 3. 14
Slika 8. Prikaz rezultata PROC COMPARE procedure iz Primjera 3. 14
Slika 9. Prikaz SAS dnevnika za upotrebu SYMBOLGEN opcije iz Primjera 4. 16
Slika 10. Prikaz tablice dobivene pomoću IF THEN DO petlje iz Primjera 4. 16
Slika 11. Prikaz SAS dnevnika za upotrebu PUT funkcije iz Primjera 5. 17
Slika 12. Prikaz rezultata za kreiranje makro naredbe iz Primjera 6. 19
Slika 13. Prikaz rezultata WINDOWS i DISPLAY funkcije iz Primjera 7. 21
Slika 14. Prikaz ispisa iz SAS dnevnika za upotrebu DO i IF THEN ELSE petlje iz Primjera 8. 22
Slika 15. Prikaz SAS dnevnika DO petlje i EVAL funkcije iz Primjera 9. 24
Slika 16.Prikaz SAS dnevnika za upotrebu DO UNTIL petlje i EVAL funkcije u Primjeru 10. 25
Slika 17. Ispis iz SAS dnevnika za upotrebu DO WHILE petlje i SYSEVALF funkcije iz Primjera 11. 27
Slika 18. Ispis iz SAS dnevnika za upotrebu LOCAL funkcije iz Primjera 12. 28
Slika 19. Ispis iz SAS dnevnika za upotrebu UPCASE i STR funkcije iz Primjera 13. 30
Slika 20. Ispis iz SAS dnevnika za upotrebu LENGTH i SUBSTR funkcije iz Primjera 14. 31
Slika 21. Prikaz rezultata PROC FREQ procedure iz Primjera 14. 32
Slika 22.Prikaz rezultata PROC MEANS procedure za podatke iz 2015. godine iz Primjera 15. i Primjera 16. 34
Slika 23.Prikaz rezultata PROC MEANS procedure za podatke iz 2016. godine iz Primjera 15. i Primjera 16. 35
Slika 24. Prikaz rezultata PROC MEANS procedure za podatke iz 2017. godine iz Primjera 15. i Primjera 16. 35
Slika 25. Prikaz rezultata PROC MEANS procedure za podatke iz 2018. godine iz Primjera 15. i Primjera 16. 36
Slika 26. Prikaz izlaznih datoteka iz Primjera 17. i Primjera 18. 38

Popis tablica

Tablica 1. Automatske makro varijable (SAS 9.4 Macro Language:Reference, Fifth Edition) 3

Životopis

Lovrić Mateja rođena je 06.10.1994 godine u Zagrebu. Pohađala je osnovnu školu Dragutina Tadijanovića. Nastavila je svoje školovanje u općoj gimnaziji Lucijana Vranjanina. Nakon završetka gimnazije upisuje Agronomski fakultet u Zagrebu. Govori engleski jezik i poznaje osnove njemačkog jezika. Osim što studira još volontira u Zoološkom vrtu u Zagrebu. Nakon završene tri godine Agronomskog fakulteta upisala je diplomski studij Genetika i oplemenjivanje domaćih životinja.