WHO Child Growth Standards
SPSS Syntax File (*igrowup_MICS.sps*)

The package igrowup_SPSS_MICS.zip contains the following items:

1. The syntax file **igrowup_MICS.sps**
2. Five permanent (read-only) SPSS data sets containing the WHO Child Growth Standards: **wazlms.sav, hazlms.sav, wfllms.sav, wfhlms.sav** and **bmilms.sav**
3. The file r**eadme_spss_MICS.pdf**
4. An example data set, **sample_MICS.sav**
5. The example syntax file, **sample_MICS.sps**
6. The example output file **sample_MICS.txt**

Contact for reporting bugs/ comments:
Should you encounter any problems with this macro, please send an e-mail with a clear description of the identified problem to "**anthro2005@who.int**", specifying in the subject line that it concerns the *igrowup_SPSS_MICS* package - also indicate which version of SPSS you are using. Thank you.

Syntax file functions

1. The SPSS syntax file (igrowup_mics.sps) calculates z-scores for the five anthropometric indicators, weight-for-age, length/height-for-age, weight-for-length, weight-for-height and body mass index (BMI)-for-age, based on the WHO Child Growth Standards. This syntax file was written to handle MICS SPSS files. If you are analyzing a file from DHS or any other spss survey data, there are specific syntax files which have the typical variable names already specified (see *igrowup_DHShh.sps, igrowup_DHSind.sps and igrowup.sps*) and which are found in their respective zip-files.

2. The syntax file flags any extreme (i.e. biologically implausible) z-scores for each indicator. The extreme z-scores are defined as follows:

Weight-for-age z-score (ZWAZ)	ZWAZ <-6 or ZWAZ >5
Length/height-for-age z-score (ZHAZ)	ZHAZ <-6 or ZHAZ >6
Weight-for-length/height z-score (ZWHZ)	ZWHZ<-5 or ZWHZ >5
BMI-for-age z-score (ZBMI)	ZBMI <-5 or ZBMI >5

3. The syntax file produces sex- and age-specific estimates for the prevalence of under/over nutrition and summary statistics (mean and SD) of the z-scores for each indicator, and if requested, stratified analysis by group such as urban/rural or regions. The first part of the output is in cross tabulation form, which some people prefer. The second part of the output uses REPORT format for the prevalences. Both give similar

results, but they may not match exactly. This is because SPSS crosstabs, particularly in the earlier versions, rounds to the nearest whole N on weighted crosstabs. The REPORT statistics at the end do not round to the nearest whole N on weighted samples. These REPORT statistics at the end are what WHO uses in the WHO Global Database on Child Growth and Malnutrition.

Syntax file requirements

Users should have good and reasonable knowledge of SPSS programming to run the syntax file properly; in addition, the following requirements must be met:

1. The Syntax file has been written to run in SPSS Version 6.0 or higher.

2. The Syntax file requires five permanent SPSS data sets containing the WHO Child Growths Standards.

Standards	**SPSS names**
Weight-for-age	wazlms.sav
Length/height-for-age	hazlms.sav
Weight-for-length	wfllms.sav
Weight-for-height	wfhlms.sav
BMI-for-age	bmilms.sav

   These files must be stored in the working directory, which is currently set to **D:\igrowup** by the program. This is also where the input dataset (*filename.sav*) is expected to be copied to. The location can easily be changed by the user, but if so, be sure to do multiple search and replaces for **D:\igrowup** so that all occurrences are found.

3. The syntax file requires a SPSS data set containing at least the variables: sex, age, weight and length/height. It must be stored in the same directory **D:\igrowup** or alternatively a user-defined directory (referred to in the syntax file as **D:\user**); the detailed requirements for its input variables are specified in the **Syntax File Parameters** section.

4. Avoid any variable names starting with underscore "_" or "$" in the input SPSS data set; they may conflict with SPSS system variables.

Syntax file outputs

The syntax file creates a new file based on the input file. The name of the new data set will be *filename_z.sav*. This data set comprises the key variables needed and the key computed variables at the end. These variables and their labels are listed below in case you need to do any debugging of the program.

Variables	Variable labels
casenum	case number
chclno	cluster number
chhhno	household number
chlnno	child's line number
wgting	weighting of sample, set to 1 if not needed (chweight)
sex	1 is male, 2 is female (hl3)
agemos	age in months (cage)
dateborn	Date child born; day,mo,yr. dmy(br3d,br3m,br3y)
datevis	Date of interview; day,mo,yr. dmy(hi3d,hi3m,hi3y)
agedays3	age in days *(datevis minus dateborn)*
agedays2	rounded (agedays3)
weight2	child's weight in kg (weight)
length	length of the child in cm (height)
lenhei2	converted length/ height according to age
lorh	Indicator variable measuring length or height (L=1, H=2) (an2a)
uselngth	Indicator variable (1=use Length tables, 2=use Height tables)
oedema	Indicator variable ("n" or "y", default = "n")
region	country region (hi7)
urbanr	urban or rural (hi6)
bmi	computed BMI
zhaz	Height-for-age z-score
zwaz	Weight-for-age z-score
zwhz	Weight-for-length/height z-score
zbmi	BMI-for-age z-score
wazflag	Flag for zwaz<-6 or zwaz>5
hazflag	Flag for zhaz<-6 or zhaz>6
whzflag	Flag for zwhz<-5 or zwhz>5
bmiflag	Flag for zbmi<-5 or zbmi>5
whznoage	Flag for no age given, but (zwhz $\geq$ -5 and zwhz $\leq$ 5)

Syntax file parameters:

The Syntax file requires **7 parameters** that should be specified:

▪ *sex*: to specify the name of a variable containing sex information. In most MICS datafiles, this numeric variable is **hl3**, and the program assumes that. However, if this is not the case, then follow the following rules: If it is a numeric variable, its values must be 1 for males and 2 for females. If it is a character variable, e.g. "m" or "M" for males and "f" or "F" for females, an appropriate recode must be made to transform character variables into a numeric **1** and **2**. Users must code its missing value as no entry for either character or numeric variables, in which case no z-scores will be calculated. If the missing value is something else, such as 0,

then you must declare it as missing in the usual fashion: `missing values sex (0).`

- *agedays3*: to specify the name of a numeric variable containing age in days. In MICS datafiles, age in days should be calculated from the values given for the interview date (hi3y,hi3m,hi3d) minus the values for the birthdate (br3y,br3m,br3d). This syntax program does that calculation and will use the age in days. If available , however, the user should **also** specify the age in months **agemos,** a numeric variable containing age in months information. In MICS datafiles, this variable is typically named **cage**. Users must code missing values as **no entry or a user declared value**, in which case the age-related z-scores are not calculated.

- *weight2*: to specify the name of a numeric variable containing body weight information. In MICS datafiles, the weight variable is typically named **weight** and it is expressed in kg. Missing values are typically expressed as **no entry,** in which case weight-related z-scores are not calculated. It is good practice to examine the range of weight values in case there is an undeclared missing value such as 999.

- *length*: to specify the name of a numeric variable containing height or length information. In MICS datafiles, this variable is typically called **height** and it is expressed in cm. Missing values are typically expressed as **no entry**, in which case height-related z-scores are not calculated. For children aged below 24 months (<731 days) and measured standing, the syntax file converts the height to recumbent length by adding 0.7 cm; and for children aged 24 months and above who were measured in recumbent position, the syntax file converts the length to standing height by subtracting 0.7 cm. In other words, all the z-scores generated by the syntax file are length-based for children below 24 months, and height-based otherwise. The variable *lenhei2* is the converted length/ height according to age.

- *lorh*: to specify the name of a character variable indicating whether recumbent length or standing height was measured. In MICS data files, this variable is typically called **an2a**. The values of this variable should be 1 for length and 2 for height. Missing values are typically expressed as 0, 7, or **no entry**, and the syntax file imputes the missing value according to the following algorithm:

    a. If age is not missing, then it is recumbent length if age below 24 months (< 731 days), and standing height if age is 24 months or above.

    b. If age is missing, then it is recumbent length if height < 87 cm and standing height if height ≥ 87 cm.

- *oedema*: to specify the name of the character variable containing oedema information. The values of this variable must be character, i.e. "n" or "N" for non-oedema, and "y" or "Y" for oedema. The current program assumes there is no

oedema present in the dataset, so oedema is set to "n". If oedema is present, you must set oedema to the variable that contains those values. Users must code its missing value as **no entry**, and the syntax file assumes missing to be non-oedema.

- *wgting*: to specify the name of a numeric variable containing sampling weight. For MICS datafiles, this variable is typically named **chweight,** and weighted analysis is the default. If **"wgting=1"** for all records, the un-weighted analysis is performed. If not otherwise specified, observations with missing or negative values in sampling weight are excluded from the analysis.

Syntax file setup and run

1. Create a directory, assumed by the syntax file to be **D:\igrowup**, where you will place the 5 files containing the standards, i.e. wazlms.sav, hazlms.sav, wfllms.sav, wfhlms.sav, bmilms.sav.

2. Copy the file **igrowup_mics.sps** to this same directory. In older versions of SPSS you may want to set SPSS to open a syntax file at startup. This is done by checking the appropriate box under Edit/Preferences.

3. Copy the file containing the child data into this same directory. In the syntax file, change the name of "*filename.sav*" to the actual name.

4. If needed, edit the syntax file so that the key variables such as weight, sex, age, length, etc. are given the correct variable names as found in the child data file.

5. Open SPSS and run the entire syntax file. It will find the data file and create the proper statistics in the output file, which will appear onscreen as you run it. You must save this output file to whatever name you wish, *filename.lst* for older versions of SPSS and *filename.spo* for newer versions. It will also create a new data file, *filename_z.sav*, which will have the key variables from the original file, plus the key computed variables. If you do not wish to have this new file saved, you can simply delete this line near the end:

   XSAVE outfile="d:\igrowup\filename_z.sav" /keep=casenum to zbmi,zwaz,zwhz,whzflag to whznoage.

   The **Untitled** data file that remains on screen at the end of the run contains all of the variables mentioned above, plus intermediate calculation variables which might be of use to those involved in understanding or debugging the syntax file.

Note: To get started, the igrowup_SPSS_MICS.zip package provides the user with an example data set (sample_MICS.sav), the respective syntax (sample_MICS.sps) and the expected output (sample_MICS.txt).