

Documentation for all Data Sets

HEALTH ABC DATA ANALYSIS FILE.....	2
PARTICIPANT HISTORY FILE (PH).....	3
YEAR 13 ANNUAL TELEPHONE CONTACT DATA (Y13Phone).....	7
YEAR 13 PROXY CONTACT DATA (Y13Proxy).....	8
150-MONTH SEMI-ANNUAL TELEPHONE CONTACT DATA (SA150Mo)	9
150-MONTH PROXY CONTACT DATA (SA150Prox)	11
MISSED FOLLOW-UP CONTACT DATA (MissVis).....	12
SPECIAL MISSING VALUE CODES	14
DROPPED VARIABLES.....	15
LISTINGS	16
DATA DICTIONARY	16
Appendix I DROPPED VARIABLES AND ALTERNATES	17
Appendix II 150-MONTH CALCULATED VARIABLES	22

HEALTH ABC DATA ANALYSIS FILE

To use the data, please contact the PI at your site.

Contents:

SAS Datasets

Y13Phone	Year 13 Annual Telephone Interview data
Y13Proxy	Year 13 Proxy Interview data
SA150Mo	150-month Semi-Annual Telephone Contact Data
SA150Prox	150-month Proxy Interview data
MissVis	Missed Follow-up Contact data
Y13Calc	Year 13 calculated (derived) variables

In addition, the following files, not specific to any year but updated each time data are released, can be found at the top of the Current Datasets listing on the Health ABC website under the [Datasets & Documentation](#) link:

PH	Participant History File
Formats	SAS Format Library

PARTICIPANT HISTORY FILE (PH)

1. General description

The PH file contains general information about the participants enrolled in the study. Variables included are:

HABCID	Health ABC Enrollment ID# without the 2-letter prefix
HCFAID	HCFA Screening ID (as assigned by the Coordinating Center)
DOB	Date of Birth
DOD	Date of Death
GENDER	Gender (1=Male; 2=Female)
RACE	Race (1=White; 2=Black)
RACEGEN	Race-gender group (1=White male, 2=White female, 3=Black male, 4=Black female)
SITE	Clinic Site (1=Memphis, 2=Pittsburgh)
CV1AGE	Age at Year 1 Clinic Visit
CV1DATE	Year 1 Clinic Visit Date
SV06AGE	Age at 6-Month Contact
SV06DATE	6-Month Contact Date
VITAL06M	Vital Status at time of 6-month contact
VTYPE06	Visit Type at 6-month contact
CV2AGE	Age at Year 2 Clinic Visit
CV2DATE	Year 2 Clinic Visit Date
VITAL12M	Vital Status at time of Year 2 (12-month) contact
VTYPE12	Visit Type at Year 2 (12-month) contact
SV18AGE	Age at 18-Month Contact
SV18DATE	18-Month Contact Date
VITAL18M	Vital Status at time of 18-month contact
VTYPE18	Visit Type at 18-month contact
CV3AGE	Age at Year 3 Clinic Visit
CV3DATE	Year 3 Clinic Visit Date
VITAL24M	Vital Status at time of Year 3 (24-month) contact
VTYPE24	Visit Type at Year 3 (24-month) contact
SV30AGE	Age at 30-Month Contact
SV30DATE	30-Month Contact Date
VITAL30M	Vital Status at time of 30-month contact
VTYPE30	Visit Type at 30-month contact
CV4AGE	Age at Year 4 Clinic Visit
CV4DATE	Year 4 Clinic Visit Date
VITAL36M	Vital Status at time of Year 4 (36-month) contact
VTYPE36	Visit Type at Year 4 (36-month) contact
SV42AGE	Age at 42-Month Contact
SV42DATE	42-Month Contact Date
VITAL42M	Vital Status at time of 42-month contact
VTYPE42	Visit Type at 42-month contact
CV5AGE	Age at Year 5 Clinic Visit
CV5DATE	Year 5 Clinic Visit Date

VITAL48M	Vital Status at time of Year 5 (48-month) contact
VTYPER48	Visit Type at Year 5 (48-month) contact
SV54AGE	Age at 54-Month Contact
SV54DATE	54-Month Contact Date
VITAL54M	Vital Status at time of 54-month contact
VTYPER54	Visit Type at 54-month contact
CV6AGE	Age at Year 6 Clinic Visit
CV6DATE	Year 6 Clinic Visit Date
VITAL60M	Vital Status at time of Year 6 (60-month) contact
VTYPER60	Visit Type at 60-month contact
SV66AGE	Age at 66-Month Contact
SV66DATE	66-Month Contact Date
VITAL66M	Vital Status at time of 66-month contact
VTYPER66	Visit Type at 66-month contact
CV7AGE	Age at Year 7 Clinic Visit
CV7DATE	Year 7 Clinic Visit Date
VITAL72M	Vital Status at time of Year 7 (72-month) contact
VTYPER72	Visit Type at 72-month contact
SV78AGE	Age at 78-Month Contact
SV78DATE	78-Month Contact Date
VITAL78M	Vital Status at time of 78-month contact
VTYPER78	Visit Type at 78-month contact
CV8AGE	Age at Year 8 Clinic Visit
CV8DATE	Year 8 Clinic Visit Date
VITAL84M	Vital Status at time of Year 8 (84-month) contact
VTYPER84	Visit Type at 84-month contact
SV90AGE	Age at 90-Month Contact
SV90DATE	90-Month Contact Date
VITAL90M	Vital Status at time of 90-month contact
VTYPER90	Visit Type at 90-month contact
CV9AGE	Age at Year 9 Clinic Visit
CV9DATE	Year 9 Clinic Visit Date
VITAL96M	Vital Status at time of Year 9 (96-month) contact
VTYPER96	Visit Type at 96-month contact
SV102AGE	Age at 102-Month Contact
SV102DATE	102-Month Contact Date
VITAL102M	Vital Status at time of 102-month contact
VTYPER102	Visit Type at 102-month contact
CV10AGE	Age at YEAR 10 Clinic Visit
CV10DATE	YEAR 10 Clinic Visit Date
VITAL108M	Vital Status at time of YEAR 10 (108-month) contact
SV114AGE	Age at 114-month Contact
SV114DATE	114-month Contact Date
VITAL114M	Vital Status at time of 114-month contact
CV11AGE	Age at YEAR 11 Clinic Visit
CV11DATE	YEAR 11 Clinic Visit Date
VITAL120M	Vital Status at time of YEAR 11 (120-month) contact
SV126AGE	Age at 126-month Contact
SV126DATE	126-month Contact Date

VITAL126M	Vital Status at time of 126-month contact
CV12AGE	Age at YEAR 12 Clinic Visit
CV12DATE	YEAR 12 Clinic Visit Date
VITAL132M	Vital Status at time of YEAR 12 (132-month) contact
SV138AGE	Age at 138-month Contact
SV138DATE	138-month Contact Date
VITAL138M	Vital Status at time of 138-month contact
CV13AGE	Age at YEAR 13 Clinic Visit
CV13DATE	YEAR 13 Clinic Visit Date
VITAL144M	Vital Status at time of YEAR 13 (144-month) contact
SV150AGE	Age at 150-month Contact
SV150DATE	150-month Contact Date
VITAL150M	Vital Status at time of 150-month contact
VERSIONPH	Participant History File Release Date
VStatus	Vital Status (1=Alive, 2=Dead) as of PH file release date

The birthdate, race, and gender data come from the edited HCFA data. The Year 1 clinic visit date was taken from the corrected final participant list provided by the each clinic (with later corrections as necessary). All other dates were taken from the corresponding Clinic Visit workbook, Home Visit workbook, Annual Telephone interview, Proxy Interview, Proxy Contact Home Visit Workbook or Semi-Annual Telephone Contact form; participants who missed a visit have no corresponding date (set to .A). Age at each contact is a calculated variable based on birthdate and that contact date.

There are 3075 observations in the PH file. The demographic breakdown of participants in this dataset is as follows:

African-American Female	729
African-American Male	552
White Female	855
White Male	939
Memphis Participants	1548
Pittsburgh Participants	1527

2. Cross reference of dataset names with exact source

A complete list of variable names can be found under the “Proc Contents for All Datasets” link (search under PH).

3. Dataset structure and contents

The PH file contains a single observation per participant.

Key variables:

HABCID	HABC Enrollment ID without the 2-letter prefix
HCFAID	HCFA ID (as assigned by the Coordinating Center)

4. Condition of data

a. Known data errors: None at this time. The data have been edited. Editing will, however, be ongoing (iterative), so use of the most recent dataset is always advised.

b. Strength and weaknesses of dataset items: A new variable RACEGEN has been added to PH to ease analyses by race-gender group (1=White male, 2=White female, 3=Black male, 4=Black female)

If a death has been reported on a Missed Visit Form, an Event Form, or the Report of Death, the participant is listed as deceased in the vital status variable (VStatus) variable. Note: VStatus is vital status **as of the release date of the participant history file**, not as of any particular visit. This variable should only be used to determine the most up-to-date snapshot of vital status in Health ABC as of the file date. Vital status variables (VITALxxM) have been created for each Health ABC contact as follows: If a participant had a particular contact, or if they missed a contact but they were determined to still be alive at the time their contact was due (participant refused the contact, was too ill, etc), then VITALxxM is alive. If they missed a contact and were later discovered to have died before the end of their contact window, then VITALxxM is dead. If they missed a contact and were later discovered to have died after the end of their contact window, the VITALxxM is alive for that contact, but dead for the next. Finally, if they missed a contact without a determination of their vital status (participant could not be located, withdrew, no data for a visit for which they are past the visit window, etc) and no further contact with vital status determination has been made since then, then VITALxxM is missing.

The date of death (DOD) variable represents the best available information about the date of death for deceased participants as of the creation date of participant history file (PH).^{*} If a Report of Death form has been entered for the participant, the date of death from that adjudication form is used. If there is no Report of Death form yet, this information is taken from the Event Form dataset, and is therefore an un-confirmed, un-adjudicated date of death.

c. Missing Value Conventions: See Special Missing Value Codes below for special missing value codes applied

5. Dataset index formulation and key variable mapping

The PH file is sorted by HABCID, which is a unique identifier for each participant.

6. General strategies for manipulating and merging the data

Because the Health ABC datasets are sorted by Health ABC Enrollment ID, the HABCID variable is most useful for merging with other datasets.

^{*} Run proc contents in SAS to see creation date of the PH file.

YEAR 13 ANNUAL TELEPHONE CONTACT DATA (Y13Phone)

1. General description

The Y13Phone file contains information about the participants enrolled in the study gathered from the Year 13 Annual Telephone interview. If a participant did not have a Year 13 Annual Telephone Interview or a Year 13 Proxy Visit, they should have a Missed Follow-up Contact form that explains why. In addition, a variable VISITYPE has been appended to allow the analyst to account for all participants, whether or not they had a Year 13 Annual Telephone Contact. If VISITYPE= 13, the participant's Year 13 data can be found in Y13PHONE. If VISITYPE= 8, the participant's Year 13 data can be found in Y13PROXY. If VISITYPE=3, 4 or 5, the participant did not have a Year 13 contact and their Missed Follow-up Contact data can be found in MISSVIS.

There are 3075 observations in the Y13Phone file. The types of visits (VISITYPE) are as follows:

Annual Phone Visit	1,359
Proxy Phone Visit	232
Missed/Deceased	1,309
Missed/Withdrew	67
Missed/Other	108

2. Cross reference of dataset names with exact source

A complete list of variable names can be found under the “Proc Contents for All Datasets” link (search under Y13Phone). Variable names can also be found on the annotated forms. Please note that not all variables on the forms are contained in the dataset. All variables not found in the dataset are listed in Dropped Variables and Alternates (Appendix I). Alternate variables to use (if applicable) are also listed.

3. Dataset structure and contents

The Y13Phone file contains a single observation per participant.

Key variables:

HABCID	HABC Enrollment ID without the 2-letter prefix
SITE	HABC Clinic site: 1=Memphis; 2=Pittsburgh *
VISITYPE	(3=Missed (other), 4=Deceased, 5=Withdrew, 8=Proxy Phone; 13=Annual Telephone Interview)
MISSREAS	Reason Y13 Annual Telephone Contact Missed

* Must link to Health ABC participant history file (PH) to add this variable.

4. Condition of data

a. Known data errors:

Year 13 Annual Telephone Interview: None at this time.

b. Strength and weaknesses of dataset items:

Calculated (derived) variables have been created to complement those created for Years 1 through 13. To avoid confusion, these variables are listed in Y13Calc.

c. Missing Value Conventions: See Special Missing Value Codes on below for special missing value codes applied.

5. Dataset index formulation and key variable mapping

The Y13Phone file is sorted by HABCID, which is a unique identifier for each participant.

6. General strategies for manipulating and merging the data

Because the Health ABC datasets are sorted by Health ABC Enrollment ID, the HABCID variable is most useful for merging with other datasets.

YEAR 13 PROXY CONTACT DATA (Y13Proxy)

1. General description

As the Health ABC cohort has aged, some participants have begun having cognitive difficulties that prevent their being able to answer interview questions, and in some cases, being able to understand the instructions for some measurements. In response to this situation, a new form was created near the end of Year 3, the Proxy Interview. In year 13, there are 234 participants who have a proxy phone visit and can be found in Y13Proxy file.

Note that a “proxy phone” contact is empirically defined as having interview data only, but no physical measurements. This definition was used for the creation of the VISITYPE variable, but was not always followed by the clinics in assigning a value to YACONTAC. Thus VISITYPE is a more reliable variable to use to determine how many proxy contacts have only interview information, vs how many have physical measurements as well.

2. Cross reference of dataset names with exact source

A complete list of variable names can be found under the “Proc Contents for All Datasets” link (search under Y13Proxy). Variable names can also be found on the annotated forms. Please note that not all variables on the forms are contained in the dataset. All variables not found in the dataset are listed in Dropped Variables and Alternates (Appendix I). Alternate variables to use (if applicable) are also listed.

3. Dataset structure and contents

The Y13Proxy file contains a single observation per participant.

Key variables:

HABCID	HABC Enrollment ID without the 2-letter prefix
SITE	HABC Clinic site: 1=Memphis; 2=Pittsburgh [*]
VISITYPE	(8=Proxy Phone)

4. Condition of data

a. Known data errors: None at this time.

b. Strength and weaknesses of dataset items: No calculated variables have been created yet for this dataset. The only standard calculated variables that might apply to these data are the self-reported function variables; however, it has been reported that proxies over-report functional limitation (Elam, et al. Am J Public Health. 1991; 81:1127), and therefore more consideration needs to be applied to how these variables should be calculated.

c. Missing Value Conventions: See Special Missing Value Codes below for special missing value codes applied

5. Dataset index formulation and key variable mapping

The Y13Proxy file is sorted by HABCID, which is a unique identifier for each participant.

6. General strategies for manipulating and merging the data

Because the Health ABC datasets are sorted by Health ABC Enrollment ID, the HABCID variable is most useful for merging with other datasets.

150-MONTH SEMI-ANNUAL TELEPHONE CONTACT DATA (SA150Mo)

1. General description

The 150-month visit file contains information about the participants enrolled in the study gathered from the 150-Month Semi-Annual Telephone Interview. In some cases, a participant was unable to complete the contact by telephone because of deafness, cognitive impairment, or other reasons, and the information was gathered by proxy. Data for these participants can be found in SA150Prox. If a participant did not have a 150-month semi-annual telephone interview or a 150-month proxy, they should have a Missed Follow-up Contact form that explains why. A variable, VISITYPE, has been added to this dataset to allow the analyst to account for all participants, whether or not they had a 150-month follow-up telephone call. If VISITYPE=7, the participant's 150-month data can be found in SA150Mo. If VISITYPE= 8, the participant's 150-month data can be found in SA150PROX. If VISITYPE=3, 4 or 5, the participant did not have a 150-month contact and their Missed Follow-up Contact data can be found in MissVis.

^{*} Must link to Health ABC participant history file (PH) to add this variable.

There are 3075 observations in the SA150Mo file as follows:

SATC Phone Visit	1,266
Proxy Phone Visit	213
Missed/Deceased	1,401
Missed/Withdrew	72
Missed/Other	123

2. Cross reference of dataset names with exact source

A complete list of variable names can be found under the “Proc Contents for All Datasets” link (search under SA150Mo). Variable names can also be found on the annotated forms. Please note that not all variables on the forms are contained in the dataset. All variables not found in the dataset are listed in Dropped Variables and Alternates (Appendix I). Alternate variables to use (if applicable) are also listed.

3. Dataset structure and contents

The SA150Mo file contains a single observation per participant.

Key variables:

HABCID	HABC Enrollment ID without the 2-letter prefix
SITE	HABC Clinic site: 1=Memphis; 2=Pittsburgh*
VISITYPE	(3=Missed (other), 4=Deceased, 5=Withdrew, 7=SATC Phone, 8=Proxy Phone)
MISSREAS	Reason 150MO Visit Missed

4. Condition of data

a. **Known data errors:**

None at this time.

b. **Strength and weaknesses of dataset items:**

Calculated (derived) variables have been added to the dataset (see Appendix II).

c. Missing Value Conventions: See Special Missing Value Codes for special missing value codes applied.

5. Dataset index formulation and key variable mapping

The SA150Mo file is sorted by HABCID, which is a unique identifier for each participant.

6. General strategies for manipulating and merging the data

Because the Health ABC datasets are sorted by Health ABC Enrollment ID, the HABCID variable is most useful for merging with other datasets.

* Must link to Health ABC participant history file (PH) to add this variable.

150-MONTH PROXY CONTACT DATA (SA150Prox)

1. General description

In some cases, a participant was unable to complete the 150-month contact by telephone because of deafness, cognitive impairment or other reasons. In response to this situation, the Proxy Interview was developed to allow another person to answer for them.

At the 150-month visit, there are 214 participants who have a Proxy Interview and can be found in SA150PROX.

A complete list of variable names can be found under the “Proc Contents for All Datasets” link (search under SA150PROX). Variable names can also be found on the annotated forms. Please note that not all variables on the forms are contained in the dataset. All variables not found in the dataset are listed in Dropped Variables and Alternates (Appendix I). Alternate variables to use (if applicable) are also listed.

3. Dataset structure and contents

The SA150Prox file contains a single observation per participant.

Key variables:

HABCID	HABC Enrollment ID without the 2-letter prefix
SITE	HABC Clinic site: 1=Memphis; 2=Pittsburgh [*]
VISITYPE	(8=Proxy Phone)

4. Condition of data

a. Known data errors: None at this time.

b. Strength and weaknesses of dataset items: No calculated variables have been created yet for this dataset. The only standard calculated variables that might apply to these data are the self-reported function variables; however, it has been reported that proxies over-report functional limitation (Elam, et al. Am J Public Health. 1991; 81:1127), and therefore more consideration needs to be applied to how these variables should be calculated.

Not all the proxy interview questions were asked at the semi-annual telephone contact. Only the “starred” questions were asked during this contact.

c. Missing Value Conventions: See Special Missing Value Codes for special missing value codes applied.

5. Dataset index formulation and key variable mapping

The SA150PROX file is sorted by HABCID, which is a unique identifier for each participant.

^{*} Must link to Health ABC participant history file (PH) to add this variable.

6. General strategies for manipulating and merging the data

Because the Health ABC datasets are sorted by Health ABC Enrollment ID, the HABCID variable is most useful for merging with other datasets.

MISSED FOLLOW-UP CONTACT DATA (MissVis)

1. General description

The MissVis file contains information about the participants who have missed a follow-up contact (died, refused, lost to follow-up, etc) at any time during the study, up through the 150-month contact. If a participant missed an annual or semi-annual contact, they should have a Missed Follow-up Contact form that explains why.

The number of Missed Follow-up Contact forms related to each of these visits is shown below:

6-month follow-up contact	23
Year 2 visit	77
18-month semi-annual contact	137
Year 3 visit	95
30-month semi-annual contact	199
Year 4 visit	160
42-month semi-annual contact	201
Year 5 visit	125
54-month semi-annual contact	151
Year 6 visit	144
66-month semi-annual contact	144
Year 7 visit	137
78-month semi-annual contact	178
Year 8 visit	204
90-month semi-annual contact	207
Year 9 visit	176
102-month semi-annual contact	175
Year 10 visit	211
Year 10.5 (114-month) semi-annual contact	167
Year 11 visit	235
Year 11.5 (126-month) semi-annual contact	221
Year 12 visit	210
Year 12.5 (138-month) semi-annual contact	181
Year 13 visit	193
Year 13.5 (150-month) semi-annual contact	221

2. Cross reference of dataset names with exact source

A complete list of variable names can be found under the “Proc Contents for All Datasets” link (search under MissVis). Variable names can also be found on the annotated forms.

3. Dataset structure and contents

The MissVis file contains multiple observations per participant.

Key variables:

HABCID	HABC Enrollment ID without the 2-letter prefix
SITE	HABC Clinic site: 1=Memphis; 2=Pittsburgh [*]
BJID2	Contact missed (9=6-month, 2=Year 2, 10=18-month, 3=Year 3, 11=30-month, 4=Year 4, 12=42-month, 5=Year 5, 13=54-month, 6=Year 6, 14=66-month, 7=Year 7, 15=78-month, 16=Year 8, 17=90-month, 18=96-month, 19=102-month, 20=Year 10, 21=114-month, 22=Year 11, 23=126-month, 24=132-month, 25=138-month, 26=144-month, 27=150-month)

4. Condition of data

a. Known data errors: None at this time..

b. Strength and weaknesses of dataset items: If a participant missed a visit due to death or withdrawal from the study, the Missed Follow-up Contact corresponding to the first contact missed for this reason is usually the last Missed Follow-up Contact for that participant. That is, field centers were instructed not to continue completing Missed Follow-up Contacts for each subsequent contact missed after the death of a participant or their withdrawal from the study. If a participant could not be located at one contact and therefore had a Missed Follow-up Contact completed for that contact, then subsequently was found to have died before that contact, the death was recorded on a Missed Follow-up Contact form for the subsequent contact. That is, the Missed Follow-up Contact information reflects the status of the participant as known to the field center at the time of the scheduled contact. Missed Follow-up Contact data should not be used to determine approximate date of death, nor even numbers of participants who had died as of a particular follow-up contact. The best information available at the time of the data analysis file regarding date of death can be found in the Participant History file (PH, DOD, see page 3). If a participant was found to have both a Missed Follow-up Contact form for a particular contact and the corresponding contact forms (e.g. Annual Telephone Interview, Proxy Interview, or Semi-Annual Follow-Up Contact form), the Missed Follow-up Contact form data for that participant were deleted from the analysis file.

c. Missing Value Conventions: See Special Missing Value Codes below for special missing value codes applied

5. Dataset index formulation and key variable mapping

The MissVis file is sorted by HABCID, which is a unique identifier for each participant. The combination of HABCID and BJID2 is a unique identifier for a participant/contact record in this dataset.

^{*} Must link to Health ABC participant history file (PH) to add this variable.

6. General strategies for manipulating and merging the data

Because the Health ABC datasets are sorted by Health ABC Enrollment ID, the HABCID variable is most useful for merging with other datasets. The MissVis file must first be subsetted by BJID2 to the contact desired before merging with a contact-specific, one-record-per-participant dataset.

SPECIAL MISSING VALUE CODES

SAS allows for stratification of missing values. The following missing values have been assigned:

. = 'Missing Form'
.A = 'A:Not Applicable'
.E = 'E:Recoded to Missing'
.F = 'F:Variable Missing from Form'
.L = 'L:Permanently Lost'
.M = 'M:Missing'
.N = 'N:Not Required'
.T = 'T:Missing Due to Technical Problems' (reading center data only)
.U = 'Unacceptable'

Description

. : Missing Form

Used when a value is missing because the entire form has not been entered.

A: Not Applicable

Used when a value is missing but the value is not required (due to simple skip pattern logic)

E: Recoded to Missing

Used to flag that a value was entered originally but should not have been (due to a skip pattern logic error) and that the value has been recoded to missing

F: Variable Missing from Form

Used to flag a variable that was not originally on the form (form was revised during the visit year) and therefore there is no value for this participant

L: Permanently Lost

Used to flag a tracking variable when a measurement involving a Reading Center was done, but the data either never made it to the Reading Center, or was lost at the Reading Center. A list of lost measurements, along with a brief description of what happened, can be found in Appendix I of the Reading Center Dataset documentation). Every effort was made to track down these data before they were declared "permanently lost" and the flag assigned.

M:Missing

Used to flag missing values when the value is required (i.e., true missing values).

N:Not Required

Used when a value is missing but the value is not required (not due to simple skip pattern logic). For example, for checkbox variables which are "Check all that apply" each one, individually, is not

required. In these cases, a summary calculated variable (not included on the dataset) was used to edit missing responses. Some variables whose skip pattern logic is non-standard (i.e., the skip pattern involves several variables and forms) also have .N flags when missing, whether or not a response was required due to the skip pattern.

T:Missing Due to Technical Problems

Used when a value is missing from the Reading Center dataset due to technical difficulties. An explanation of when this value has been assigned can be found under Strengths and Weaknesses of (Reading Center) dataset items for each Reading Center (Reading Center data documentation; Substudy documentation).

U:Unacceptable

Used with certain Reading Center data when the data exist but cannot be used., for example, DXA data when the whole scan has been reviewed as unacceptable

General Strategies for Using Special Missing Values

In SAS, when using special missing values in logical expressions, the missing value is no longer only equal to ‘.’ To express a value equal to missing, the code should be written: `<= .z` or alternately: `le .z`

To express a value not equal to missing, the code should be written `>.z` or alternately: `gt .z`. `.Z` is the greatest value of missing available in SAS.

DROPPED VARIABLES

A number of variables appear on the annotated forms but will not be found in the datasets. These variables are listed in Appendix I. There are several reasons why variables were dropped:

1. Participant confidentiality: identifying information such as participant name, acrostic, address, telephone number, etc. are omitted from the analysis file. All participants are instead identified by the HABC Enrollment ID# (HABCID).
2. Bookkeeping variables: a number of variables were put on the forms merely for bookkeeping purposes and are extremely unlikely to be useful for analysis. If an investigator notices that such a variable has been dropped and believes it should not have been, s/he should feel free to contact the Coordinating Center (HABCHelp@psg.ucsf.edu) and let us know that it should be included in future datasets.
3. Redundant variables: in many cases the same information was collected numerous times. We have made a special effort to clean up one version of each of these; and to avoid analysis using uncleaned data, the uncleaned version is omitted.

Appendix I lists the omitted variables in alphabetical order. There are two columns; the first, entitled “Variable omitted,” lists the name of the variable not included in the dataset. The second column, entitled “Variable to use,” lists the cleaned variable, for redundant variables, or the reason the variable was not included.

LISTINGS

A PDF listing of the SAS proc contents printout for all SAS datasets can be found under the “Proc Contents for All Datasets” link.

A text file, Formats.lst, showing all formats and value descriptions (e.g.: 1=White, 2=Black) contained in the SAS Format Library can also be downloaded. Click on FormatsList under the List of Current Datasets on the Health ABC website.

The following files are zipped together in a self-extracting document also available on the Health ABC website under the Current Datasets listing

Datadict.xls Microsoft Excel 4.0 spreadsheet containing information about all variables included in the SAS datasets

Datadict.dat Tab-delimited text file containing the same information as above.

DATA DICTIONARY (datadict.xls, datadict.txt)

General description

This is a searchable/sortable file that contains all the variables included in the SAS datasets. The following fields are included:

Variable	Variable Name
Label	Description of the variable
Form	Form or reading center origin of the variable
Page	Page number (not applicable for reading center or PH data)
Database	Database location of the variable
Variable Types	Type of variable (text, categorical (numeric), continuous (numeric), date, time, etc.)
Possible Values	Range of possible values associated with the variable (used for range edits)
SAS Format	SAS format assigned to the variable
Page Order	Variable order on the data collection forms (useful for sorting)

General Strategies for Use

The data dictionary is provided in two formats: Excel 2000 and tab-delimited text. They contain exactly the same information. The tab-delimited file was generated to provide easy access to those who do not have Excel version 2000 or higher.

The file is currently sorted by variable name. It can be used to search for details on a particular variable, to group by database or form, or to find a variable location on a form. It is a little more user friendly than the standard SAS proc contents, as it can be sorted, easily searched, and it provides additional details such as page number and form.

Appendix I

DROPPED VARIABLES AND ALTERNATES

Variable omitted	Variable to use
BJID	HABCID
BJACROS	N/A (confidentiality)
BJCONTAC	BJID2
BJSTFID	N/A (bookkeeping)
BJTYPE	BJID2
BJVISIT	BJID2
BLID	HABCID
BLACROS	N/A (confidentiality)
BLINK	N/A (bookkeeping)
BLMOVE	N/A (bookkeeping)
BLREF13A	N/A (bookkeeping)
BLREF13B	N/A (bookkeeping)
BLREF13C	N/A (bookkeeping)
BLREF14A	N/A (bookkeeping)
BLREF14B	N/A (bookkeeping)
BLREF14C	N/A (bookkeeping)
BLREF15A	N/A (bookkeeping)
BLREF15B	N/A (bookkeeping)
BLREF15C	N/A (bookkeeping)
BLREF16A	N/A (bookkeeping)
BLREF16B	N/A (bookkeeping)
BLREF16C	N/A (bookkeeping)
BLREF17A	N/A (bookkeeping)
BLREF17B	N/A (bookkeeping)
BLREF17C	N/A (bookkeeping)
BLREF18A	N/A (bookkeeping)
BLREF18B	N/A (bookkeeping)
BLREF18C	N/A (bookkeeping)
BLREF19A	N/A (bookkeeping)
BLREF19B	N/A (bookkeeping)
BLREF19C	N/A (bookkeeping)
BLREF19D	N/A (bookkeeping)
BLREF19E	N/A (bookkeeping)
BLREF19F	N/A (bookkeeping)
BLREF20A	N/A (bookkeeping)
R2ID	HABCID
R2ACROS	N/A (confidentiality)
R2CONTAC	VISITYPE

Variable omitted	Variable to use
R3ID	HABCID
R3ACROS	N/A (confidentiality)
R3CONTAC	VISITYPE
R4ID	HABCID
R4ACROS	N/A (confidentiality)
R4CONTAC	VISITYPE
R5ID	HABCID
R5ACROS	N/A (confidentiality)
R5CONTAC	VISITYPE
R6ID	HABCID
R6ACROS	N/A (confidentiality)
R6CONTAC	VISITYPE
R7ID	HABCID
R7ACROS	N/A (confidentiality)
R7CONTAC	VISITYPE
R8ID	HABCID
R8ACROS	N/A (confidentiality)
R8CONTAC	VISITYPE
R9ID	HABCID
R9ACROS	N/A (confidentiality)
R9CONTAC	VISITYPE
RAID	HABCID
RAACROS	N/A (confidentiality)
RACONTAC	VISITYPE
RBID	HABCID
RBACROS	N/A (confidentiality)
RBCONTAC	VISITYPE
RBHCHRS	RBHCTIM
RBHCMINS	RBHCTIM
RCEWHRS	RCEWTIM
RCEWMINS	RCEWTIM
RCID	HABCID
RCACROS	N/A (confidentiality)
RCCONTAC	VISITYPE
RDID	HABCID
RDACROS	N/A (confidentiality)
RDCONTAC	VISITYPE
RDHIA1HR	RDH1TIME
RDHIA1MN	RDH1TIME
REID	HABCID
REACROS	N/A (confidentiality)
RECONTAC	VISITYPE
REMA1HR	REM1TIME

Variable omitted	Variable to use
REMI1MN	REMI1TIME
RFID	HABCID
RFACROS	N/A (confidentiality)
RFCONTAC	VISITYPE
RGID	HABCID
RGACROS	N/A (confidentiality)
RGCONTAC	VISITYPE
RHID	HABCID
RHACROS	N/A (confidentiality)
RHCONTAC	VISITYPE
RIID	HABCID
RIACROS	N/A (confidentiality)
RICONTAC	VISITYPE
RJID	HABCID
RJACROS	N/A (confidentiality)
RJCONTAC	VISITYPE
RKID	HABCID
RKACROS	N/A (confidentiality)
RKCONTAC	VISITYPE
RKREF39A	N/A (bookkeeping)
RKREF39B	N/A (bookkeeping)
RKREF39C	N/A (bookkeeping)
RKREF40A	N/A (bookkeeping)
RKREF40B	N/A (bookkeeping)
RKREF40C	N/A (bookkeeping)
RLID	HABCID
RLACROS	N/A (confidentiality)
RLCONTAC	VISITYPE
RLREF41A	N/A (bookkeeping)
RLREF41B	N/A (bookkeeping)
RLREF41C	N/A (bookkeeping)
RLREF42A	N/A (bookkeeping)
RLREF42B	N/A (bookkeeping)
RLREF42C	N/A (bookkeeping)
RMID	HABCID
RMACROS	N/A (confidentiality)
RMCONTAC	VISITYPE
RMREF43A	N/A (bookkeeping)
RMREF43B	N/A (bookkeeping)
RMREF43C	N/A (bookkeeping)
RMREF44A	N/A (bookkeeping)
RMREF44B	N/A (bookkeeping)
RMREF44C	N/A (bookkeeping)

Variable omitted	Variable to use
RNID	HABCID
RNACROS	N/A (confidentiality)
RNCONTAC	VISITYPE
RNREF45A	N/A (bookkeeping)
RNREF45B	N/A (bookkeeping)
RNREF45C	N/A (bookkeeping)
RNREF45D	N/A (bookkeeping)
RNREF45E	N/A (bookkeeping)
RNREF45F	N/A (bookkeeping)
RNREF46A	N/A (bookkeeping)
ROID	HABCID
ROACROS	N/A (confidentiality)
ROCONTAC	VISITYPE
RPID	HABCID
RPACROS	N/A (confidentiality)
RPCONTAC	VISITYPE
RUID	HABCID
RUACROS	N/A (confidentiality)
RUCONTAC	VISITYPE
RVID	HABCID
RVACROS	N/A (confidentiality)
RVCONTAC	VISITYPE
RWID	HABCID
RWACROS	N/A (confidentiality)
RWCONTAC	VISITYPE
RXID	HABCID
RXACROS	N/A (confidentiality)
RXCONTAC	VISITYPE
RXADDYN	N/A (bookkeeping)
RXSSESPY	N/A (bookkeeping)
RYID	HABCID
RYACROS	N/A (confidentiality)
RYCONTAC	VISITYPE
RYCIYN	N/A (bookkeeping)
RYKNOK	N/A (bookkeeping)
RYKYN	N/A (bookkeeping)
RZID	HABCID
RZACROS	N/A (confidentiality)
RZCONTAC	VISITYPE
RZPPOA	N/A (bookkeeping)
RZPAYN	N/A (bookkeeping)
RZP2YN	N/A (bookkeeping)
SIID	HABCID

Variable omitted	Variable to use
S1ACROS	N/A (confidentiality)
S1CONTAC	VISITYPE
S1C1YN	N/A (bookkeeping)
T4ID	HABCID
T4ACROS	N/A (confidentiality)
T4CONTAC	VISITYPE
T4FNM	N/A (confidentiality)
T4LNM	N/A (confidentiality)
T4REASON	N/A (bookkeeping)
YAID	HABCID
YAACROS	N/A (confidentiality)
YACONTAC	VISITYPE
YAVISIT	N/A (bookkeeping)
YALINK	N/A (bookkeeping)
YAMOVE	N/A (confidentiality)
YAREF11A	N/A (bookkeeping)
YAREF11B	N/A (bookkeeping)
YAREF11C	N/A (bookkeeping)
YAREF12A	N/A (bookkeeping)
YAREF12B	N/A (bookkeeping)
YAREF12C	N/A (bookkeeping)
YAREF13A	N/A (bookkeeping)
YAREF13B	N/A (bookkeeping)
YAREF13C	N/A (bookkeeping)
YAREF14A	N/A (bookkeeping)
YAREF14B	N/A (bookkeeping)
YAREF14C	N/A (bookkeeping)
YAREF15A	N/A (bookkeeping)
YAREF15B	N/A (bookkeeping)
YAREF15C	N/A (bookkeeping)
YAREF16A	N/A (bookkeeping)
YAREF16B	N/A (bookkeeping)
YAREF16C	N/A (bookkeeping)
YAREF17A	N/A (bookkeeping)
YAREF17B	N/A (bookkeeping)
YAREF17C	N/A (bookkeeping)
YAREF17D	N/A (bookkeeping)
YAREF17E	N/A (bookkeeping)
YAREF17F	N/A (bookkeeping)
YAREF18A	N/A (bookkeeping)

Appendix II

150-MONTH CALCULATED VARIABLES

Variable	Variable Description	Grouping
CSAINDEX	Climbing stairs ability index	Self-Reported Function Calculated Vars.
EASE1F	Ease climbing 1 flight	Self-Reported Function Calculated Vars.
EASE1M	Ease walking 1 mile	Self-Reported Function Calculated Vars.
EASE2F	Ease climbing 2 flights	Self-Reported Function Calculated Vars.
EASEQM	Ease walking 1/4 mile	Self-Reported Function Calculated Vars.
WKAINDEX	Walking ability index	Self-Reported Function Calculated Vars.

Documentation for Year 13 Calculated Variable Dataset

YEAR 13 CALCULATED VARIABLE DATASET (Y13Calc).....24

- Appendix I Calculated Variable List
- Appendix II Self-Reported Function
- Appendix III Smoking Habits
- Appendix IV Physical Activity and Exercise

YEAR 13 CALCULATED VARIABLE DATASET (Y13Calc)

1. General description

The Y13Calc contains 26 variables derived (calculated) as described below. The documentation is grouped by the type of data from which the variables are calculated (self-reported function, smoking habits, etc.), and the variables are positioned in the dataset in the same groupings.

There are 3075 observations in the Y13Calc file.

2. Cross reference of dataset names with exact source

A complete list of variable names can be found under the “Proc Contents for All Datasets” link (search under Y13Calc) and in Appendix I. NOTE: SOME CALCULATED VARIABLES USE THE SAME VARIABLE NAMES EVERY YEAR. THE ANALYST IS REMINDED TO RENAME VARIABLES BEFORE MERGING WITH OTHER YEARS AS NECESSARY.

3. Dataset structure and contents

The Y13Calc file contains a single observation per participant.

Key variables:

HABCID	Health ABC Enrollment ID# without the 2-letter prefix
GENDER	Gender (1=Male; 2=Female)*
RACE	Race (1=White; 2=Black)*
RACEGEN	Race-gender group (1=White male, 2=White female, 3=Black male, 4=Black female)*
SITE	Clinic Site (1=Memphis, 2=Pittsburgh)*

4. Condition of data

a. Strengths and weaknesses of dataset items: When a calculated variable includes imputations for missing variables, the method of imputation is **bolded** in the documentation. There were no major logic changes used to derive calculated variables for year 13 as compared with previous years.

5. Dataset index formulation and key variable mapping

The Y13Calc file is sorted by HABCID, which is a unique identifier for each participant.

* Must link to Health ABC participant history file (PH) to add this variable.

6. General strategies for manipulating and merging the data

Because the Health ABC datasets are sorted by Health ABC Enrollment ID, the HABCID variable is most useful for merging with other datasets.

Appendix I
CALCULATED VARIABLE LIST

Variable	Variable Description	Grouping
EASE1F	Ease climbing 1 flight	Self-reported function
EASE2F	Ease climbing 2 flights	Self-reported function
CSAINDEX	Climbing stairs ability index	Self-reported function
EASE10P	Ease lift/carry 10 lbs	Self-reported function
EASE20P	Ease lift/carry 20 lbs	Self-reported function
LCAINDEX	Lift/carry ability index	Self-reported function
EASEUP	Ease standing from chair without using arms	Self-reported function
SMK13	Smoking status at Year 13 visit	Smoking Habits
Y13MCKKWK	Kcal/kg/week doing major chores	Physical Activity
Y13FSKKWK	Kcal/kg/week climbing stairs	Physical Activity
Y13TWKKWK	Kcal/kg/week total walking	Physical Activity
WALKTIME	Minutes walking/week	Physical Activity
WALKCAT	Minutes walking/week category	Physical Activity
BKTWTIME	Minutes walking briskly/week	Physical Activity
BRISK90	Walks briskly \geq 90 min/week	Physical Activity
BRISK180	Walks briskly \geq 180 min/week	Physical Activity
WSKKWK	Kcal/kg/week - walking + stairs	Physical Activity

Appendix II
Self-Reported Function Calculated Variables

Investigator Name: Eleanor Simonsick

Variable	General Description	Detailed Description	How variable is calculated	How to handle missing or special values	Value labels
EASEQM	Ease walking 1/4 mile	Recode of Y13DWQMEZ where 3 is easiest; imputations applied for missing Y13DWQMEZ	If Y13DWQMYN in (0,7,8) then EASEQM=7-Y13DWQMEZ If Y13DWQMYN=1 then EASEQM=4-Y13DWQMDF	IF Y13DWQMYN=9 AND (Y13MNRS>0 OR Y13MNRS4 IN(1,2)) THEN Y13DWQMYN=1; IF Y13DWQMEZ=8 and Y13DWQMYN ne 8 THEN Y13DWQMEZ=2; IF Y13DWQMDF=8 and Y13DWQMYN ne 8 THEN Y13DWQMDF=2; IF (Y13DWQMYN=0 AND Y13DWQMEZ<0) THEN EASEQM=7-Y13DW1MEZ; IF (Y13DWQMYN=0 AND Y13DWQMEZ<0) AND Y13DW1MEZ<0 AND (Y13DW1MYN=1 OR Y13DW1MYN=8 OR Y13DW1MYN<=.z) THEN EASEQM=4; IF (Y13DWQMYN=0 AND Y13DWQMEZ<0) AND (Y13DW1MYN=0 AND Y13DW1MEZ<=.z) THEN EASEQM=5; IF Y13DWQMYN=1 AND Y13DWQMDF<0 THEN EASEQM=7-Y13DWQMEZ; IF Y13DWQMYN=1 AND Y13DWQMDF<0 AND Y13DWQMEZ<0 THEN EASEQM=2; IF Y13DWQMYN<0 AND Y13DWQMDF>0 THEN EASEQM=4-Y13DWQMDF; IF Y13DWQMYN<0 AND Y13DWQMEZ>0 THEN EASEQM=7-Y13DWQMEZ; IF Y13DWQMYN=8 and Y13DWQMEZ=8 then EASEQM=4; IF Y13DWQMYN=9 AND Y13MNRS<0 AND (Y13MNRS4<0 OR Y13MNRS4=8) THEN EASEQM=.	6=very easy 5=somewhat easy 4=not that easy 3=a little difficult 2=somewhat difficult 1=very difficult 0=unable to do

Self-Reported Function Calculated Variables

Variable	General Description	Detailed Description	How variable is calculated	How to handle missing or special values	Value labels
EASE1M	Ease walking 1 mile	Recode of Y13DW1MEZ where 3 is easiest; imputations applied for missing Y13DW1MEZ	If Y13DW1MYN=1 then EASE1M=0; if Y13DW1MEZ=3 then EASE1M=1; if Y13DW1MEZ=2 then EASE1M=2; if Y13DW1MEZ=1 then EASE1M=3	IF Y13DW1MYN=8 AND (Y13DW1MEZ<0 OR Y13DW1MEZ=8) THEN EASE1M=1; IF Y13DW1MYN<=.z AND Y13DW1MEZ<=.z AND EASEQM=4 THEN EASE1M=0; IF Y13DW1MYN<=.z AND Y13DW1MEZ<=.z AND EASEQM=5 THEN EASE1M=1; IF Y13DW1MYN<=.z AND Y13DW1MEZ<=.z AND EASEQM=6 THEN EASE1M=2; IF Y13DW1MYN=0 AND Y13DW1MEZ<=.z AND EASEQM=4 THEN EASE1M=0; IF Y13DW1MYN=0 AND Y13DW1MEZ<=.z AND EASEQM=5 THEN EASE1M=1; IF Y13DW1MYN=0 AND Y13DW1MEZ<=.z AND EASEQM=6 THEN EASE1M=2; IF Y13DWQMYN=1 THEN EASE1M=0; IF EASEQM GT 0 AND ((EASEQM-3) LT EASE1M) THEN DO; IF EASEQM LE 3 THEN EASE1M=0; ELSE EASE1M=EASEQM-3; END; if Y13DW1MYN=8 and Y13DW1MEZ in (1,2,3) then EASE1M=4-Y13DW1MEZ; if Y13DW1MYN=0 and Y13DW1MEZ=8 then EASE1M=EASEQM-4;	3=very easy 2=somewhat easy 1=not that easy 0=difficult
WKAINDEX	Walking ability index	Summary measure of self-reported walking ability.	WKAINDEX=EASEQM + EASE1M	Imputed version of component variables used	This scale ranges from 0 (unable) to 9 (very easy)

Self-Reported Function Calculated Variables

Variable	General Description	Detailed Description	How variable is calculated	How to handle missing or special values	Value labels
EASE1F	Ease Climbing 1 flight	Recode of Y13DW10EZ where 3 is easiest; imputations applied for missing Y13DW20EZ	EASE1F=7-Y13DW10EZ; IF Y13DW10YN=1 THEN EASE1F=4-Y13DIF;	IF Y13DW10YN=9 then EASE1F=.; IF Y13DW10EZ=8 THEN Y13DW10EZ=2; IF Y13DIF=8 THEN Y13DIF=2; IF Y13DW10EZ<0 THEN EASE1F=7-Y13DW20EZ; IF Y13DW10YN=1 AND (Y13DIF=8 OR Y13DIF<0) THEN EASE1F=2; IF Y13DW10YN=0 AND Y13DW10EZ<0 AND Y13DW10EZ<0 THEN EASE1F=4; IF EASE1F=. AND Y13DW10YN NE 9 AND Y13DW10YN>0 AND (Y13DW20YN=1 OR Y13DW20YN=8 OR Y13DW20YN<=.z) THEN EASE1F=4; IF EASE1F=. AND Y13DW20YN=0 AND Y13DW20EZ<=.z THEN EASE1F=5;	6=very easy 5=somewhat easy 4=not that easy 3=a little difficult 2=somewhat difficult 1=very difficult 0=unable to do
EASE2F	Ease Climbing 2 flights	Recode of Y13DW20EZ where 3 is easiest; imputations applied for missing Y13DW20EZ	IF Y13DW20YN=1 THEN EASE2F=0; ELSE IF Y13DW20EZ=3 THEN EASE2F=1; ELSE IF Y13DW20EZ=2 THEN EASE2F=2; ELSE IF Y13DW20EZ=1 THEN EASE2F=3;	IF Y13DW10YN=9 then EASE1F=.; IF (Y13DW20YN=8 OR Y13DW20YN<0)AND (Y13DW20EZ<0 OR Y13DW20EZ=8) AND EASE1F>0 THEN EASE2F=1; IF Y13DW20YN<=.z AND Y13DW20EZ<=.z AND EASE1F=4 THEN EASE2F=0; IF Y13DW20YN<=.z AND Y13DW20EZ<=.z AND EASE1F=5 THEN EASE2F=1; IF Y13DW20YN<=.z AND Y13DW20EZ<=.z AND EASE1F=6 THEN EASE2F=2; IF Y13DW20YN=0 AND (Y13DW20EZ<=.z OR Y13DW20EZ=8) AND EASE1F>0 THEN EASE2F=EASE1F-4; IF Y13DW10YN=1 THEN EASE2F=0; IF .z<EASE1F<=3 THEN EASE2F=0; ELSE IF (EASE1F-3) LT EASE2F THEN EASE2F=EASE1F-3;	3=very easy 2=somewhat easy 1=not that easy 0=difficult
CSAINDEX	Climbing stairs ability index	Summary measure of self-reported walking ability.	CSAINDEX=EASE1F + EASE2F	Imputed version of component variables used	This scale ranges from 0 (unable) to 9 (very easy)

```

*****
*****
*   Saved as:
*   \\Fu-hsing-c\HABC\HABC_SAS\Calculated
Variables\Programs\Year 13\srfcn_Y13.sas
*
*   Todd Glasser 01/06/2011 - Modified year 11
program for use in Year 13
*****
*****;

*****
*****;
**   HABC Year 13 SELF-REPORT PHYSICAL FUNCTION CALC
VAR CODE          **;
**
**;
**   Creates the following variables:
**;
**       EASEQM:    ease of walking 1/4 mile
**;
**       EASE1M:    ease of walking 1 mile
**;
**       EASE1F:    ease of climbing 1 flight
**;
**       EASE2F:    ease of climbing 2 flights
**;
**       EASE10P:   ease of lifting/carrying 10 lbs
**;
**       EASE20P:   ease of lifting/carrying 20 lbs
**;
**       EASEUP:    ease of standing up from chair w/o
using arms          **;
**       CSAINDEX: climbing stairs ability index
**;
**       WKAINDEX: walking ability index
**;
**       LCAINDEX: lift/carry ability index
**;

```

```

**
** ;
** Adapted from FY13n Harris Y2 code: srfcn_y2.sas
** ;
*****
***** ;

```

```
*DM LOG 'CLEAR' ; *DM OUTPUT 'CLEAR' ;
```

```

%include '\\Fu-hsing-
c\HABC\HABC_SAS\Programs\initV8_ase.sas' ;
libname calc '\\Fu-hsing-c\habc\habc_sas\calculated
variables\datasets\Year 13' ;

```

```

*options ls=90 ps=56 formchar='|----|+|---+=| -
/\<>*' nodate nofmterr ;

```

```

data calc.Y13srfcn (keep=HABCID EASE1F EASE2F
EASEQM EASE1M EASE10P EASE20P EASEUP CSAINDEX
WKAINDEX
                LCAINDEX) ;
    ***Y13 data*** ;
/*
    merge habcrts.r2 (keep=habcid r2contac
where=(r2contac=20))
                habcrts.Y13 (keep=habcid
Y13contac Y13dwqmyn Y13dwqmdfY134mnrs
wherY13(r4contac=20))
                habcrts.Y13 (keep=habcid
Y13contac Y13dwqmez Y13dw1mynY135dw1mez
wherY13(r5contac=20))
                habcrts.Y13 (keep=habcid
Y13contac Y13dw10yn Y13dif where=Y136contac=20))
                habcrts.Y13 (keep=habcid
Y13contac Y13dw10ez Y13dw20ynY137dw20ez
wherY13(r7contac=20))

```

```

                                habcrts.Y13 (keep=habcid
Y13contac Y13difsta Y13ezstaY138dstamt
wherY13(r8contac=20))
                                habcrts.Y13 (keep=habcid
Y13contac Y13dif10 Y13d10amtY13aez10lb rad20lY13
raez20lb whY13e=(racontac=20)) ;
*/
        set daf.Y13visit (keep=habcid Y13dwqmyn
Y13dwqmdf Y13mnrs
                                Y13dwqmez Y13dwlmyn
Y13dwlmez
                                Y13dw10yn Y13dif
                                Y13dw10ez Y13dw20yn
Y13dw20ez
                                Y13difsta Y13ezsta
Y13dstamt
                                Y13dif10 Y13d10amt
Y13ez10lb Y13d20lbs Y13ez20lb) ;
        by habcid ;
        LABEL    EASEQM    ='EASE WALKING 1/4 MILE,
6=VERY EASY'
                                EASE1M    ='EASE WALKING 1
MILE, 3=VERY EASY'
                                WKAINDEX ='WALKING ABILITY
INDEX, 9=BEST'
                                EASE1F    ='EASE CLIMBING 1
FLIGHT, 6=VERY EASY'
                                EASE2F    ='EASE CLIMBING 2
FLIGHTS, 3=VERY EASY'
                                CSAINDEX  ='CLIMBING STAIRS
ABILITY INDEX, 9=BEST'
                                EASE10P   ='EASE LIFT/CARRY
10 LBS, 6=VERY EASY'
                                EASE20P   ='EASE LIFT/CARRY
20 LBS, 3=VERY EASY'
                                LCAINDEX  ='LIFT/CARRY
ABILITY INDEX, 9=BEST'
                                EASEUP    ='EASE STANDING
FROM CHAIR WITHOUT USING ARMS, 6=VERY EASY' ;

```

*EASEQM: CODE FOR EASE OF WALKING 1/4 MILE
IF EASE LEVEL MISSING, EASE LEVEL WALKING 1
MILE WAS ASSIGNED. IF NO EASE LEVEL FOR WALKING
1M WAS CODED AND Y13dw1myn WAS YES, DONT
KNOW, OR MISSING THEN NOT THAT EASY WAS ASSIGNED.
IF NO

EASE LEVEL FOR WALKING 1M WAS CODED AND
Y13dw1myn WAS NO AND Y13dw1mez WAS MISSING THEN
SOMEWHAT

EASY WAS ASSIGNED. THE 4 CASES WHO
REPORTED DIFFICULTY WERE RECODED TO NOT THAT EASY
FOR QM

AND DIFFICULTY FOR 1M;

IF Y13dwqmyn=9 AND (Y13mnrs>0 and Y13mnrs
ne 22) THEN Y13dwqmyn=1;

IF Y13dwqmez=8 and Y13dwqmyn ne 8 THEN
Y13dwqmez=2;

IF Y13dwqmdf=8 and Y13dwqmyn ne 8 THEN
Y13dwqmdf=2;

IF Y13dwqmyn IN (0,7,8) AND Y13dwqmez<0
THEN EASEQM=4;

ELSE IF Y13dwqmyn IN (0,7,8) THEN EASEQM=7-
Y13dwqmez;

IF (Y13dwqmyn=0 AND Y13dwqmez<0) THEN
EASEQM=7-Y13dw1mez;

IF (Y13dwqmyn=0 AND Y13dwqmez<0) &
Y13dw1mez<0 & (Y13dw1myn in(1,8) OR
Y13dw1myn<=.z) THEN EASEQM=4;

IF (Y13dwqmyn=0 & Y13dwqmez<0) &
(Y13dw1myn=0 & Y13dw1mez<=.z) THEN EASEQM=5;

IF Y13dwqmyn=1 THEN EASEQM=4-Y13dwqmdf;

IF Y13dwqmyn=1 AND Y13dwqmdf<0 THEN
EASEQM=7-Y13dwqmez;

IF Y13dwqmyn=1 AND Y13dwqmdf<0 AND
Y13dwqmez<0 THEN EASEQM=2;

IF Y13dwqmyn<0 AND Y13dwqmdf>0 THEN
EASEQM=4-Y13dwqmdf;

```
IF Y13dwqmyn<0 AND Y13dwqmez>0 THEN  
EASEQM=7-Y13dwqmez;
```

```
if Y13dwqmyn=8 and Y13dwqmez=8 then  
EASEqm=4;
```

```
IF Y13dwqmyn=9 AND (Y13mnrs<0 or  
Y13mnrs=22) THEN EASEQM=.;
```

*EASE1M: CODE FOR EASE OF WALKING 1 MILE.
AS SELF-REPORTED WALKING ABILITY IS A KEY DEPENDENT
MEASURE IN HEALTH ABC, MISSING VALUES WERE
RECODED TO NON-MISSING WHENEVER A REASON
GUESS COULD BE MADE BASED ON OTHER
AVAILABLE INFORMATION. SEE BELOW FOR CODE AND
RECODING RULES FOR MISSING VALUES;

```
IF Y13dwlmyrn=1 THEN EASE1M=0;  
ELSE IF Y13dwlmez=3 THEN EASE1M=1;  
ELSE IF Y13dwlmez=2 THEN EASE1M=2;  
ELSE IF Y13dwlmez=1 THEN EASE1M=3;
```

*MISSING VALUE RECODES;

```
IF Y13dwlmyrn=8 AND (Y13dwlmez<0 OR  
Y13dwlmez=8) THEN EASE1M=1;  
IF Y13dwlmyrn<=.z AND Y13dwlmez<=.z AND  
EASEQM=4 THEN EASE1M=0;  
IF Y13dwlmyrn<=.z AND Y13dwlmez<=.z AND  
EASEQM=5 THEN EASE1M=1;  
IF Y13dwlmyrn<=.z AND Y13dwlmez<=.z AND  
EASEQM=6 THEN EASE1M=2;  
IF Y13dwlmyrn=0 AND Y13dwlmez<=.z AND  
EASEQM=4 THEN EASE1M=0;  
IF Y13dwlmyrn=0 AND Y13dwlmez<=.z AND  
EASEQM=5 THEN EASE1M=1;  
IF Y13dwlmyrn=0 AND Y13dwlmez<=.z AND  
EASEQM=6 THEN EASE1M=2;  
IF Y13dwqmyn=1 THEN EASE1M=0;
```

*IN SOME CASES THE AMOUNT OF EASE REPORTED FOR WALKING 1 MILE WAS GREATER THAN THE AMOUNT OF EASE REPORTED FOR 1/4 MILE. THE CODE BELOW DOWNGY13DES THE EASE LEVEL REPORTED FOR 1M TO THAT REPORTED FOR 1/4M;

```
IF EASEQM GT 0 AND ((EASEQM-3) LT EASE1M)
THEN DO;
    IF EASEQM LE 3 THEN EASE1M=0; ELSE
    EASE1M=EASEQM-3;
END;

if Y13dw1myn=8 and Y13dw1mez in (1,2,3)
then EASE1M=4-Y13dw1mez;
if Y13dw1myn=0 and Y13dw1mez=8 then
EASE1M=EASEQM-4;
```

*CODE TO CREATE A SUMMARY MEASURE OF SELF-REPORTED WALKING ABILITY. THIS SCALE WILL ULTIMATELY Y13NGE FROM 0 TO 9, BUT AT BASELINE IT Y13NGES FROM 4-9, AS NO ONE HAS DIFFICULTY. THIS SCALE CAN BE CREATED IN TWO WAYS: 1) KEYING OFF THE RESPONSE TO EASE1M AND ONLY LOOKING AT EASEQM WHEN DIFFICULTY IS REPORTED FOR 1M OR 2) SUMMING THE RESPONSES TO THE TWO QUESTIONS. I DID IT BOTH WAYS AND FOUND THE LATTER APPROACH SUPERIOR;

```
WKAINDEX=EASEQM + EASE1M;
```

*CODE FOR EASE OF CLIMBING STAIRS;

*EASE1F;

```

        IF Y13dw10yn=9 THEN DO; EASE1F=.; EASE2F=.;
END;
        IF Y13dw10ez=8 THEN Y13dw10ez=2;
        IF Y13dif=8 THEN Y13dif=2;
        EASE1F=7-Y13dw10ez;
        IF Y13dw10ez<0 THEN EASE1F=7-Y13dw20ez;
        IF Y13dw10yn=1 THEN EASE1F=4-Y13dif;
        IF Y13dw10yn=1 AND (Y13dif=8 OR Y13dif<0)
THEN EASE1F=2;
        IF Y13dw10yn=0 AND Y13dw10ez<0 AND
Y13dw10ez<0 THEN EASE1F=4;

        IF EASE1F=. & Y13dw10yn NE 9 & Y13dw10yn>0
& (Y13dw20yn=1 OR
        Y13dw20yn=8 OR Y13dw20yn<=.z ) THEN
EASE1F=4;
        IF EASE1F=. AND Y13dw20yn=0 AND
Y13dw20ez<=.z THEN EASE1F=5;

        *EASE2F;

        IF Y13dw20yn=1 THEN EASE2F=0;
        ELSE IF Y13dw20ez=3 THEN EASE2F=1;
        ELSE IF Y13dw20ez=2 THEN EASE2F=2;
        ELSE IF Y13dw20ez=1 THEN EASE2F=3;
        IF (Y13dw20yn=8 OR Y13dw20yn<0)AND
(Y13dw20ez<0 OR Y13dw20ez=8) AND EASE1F>0 THEN
EASE2F=1;
        IF Y13dw20yn<=.z AND Y13dw20ez<=.z AND
EASE1F=4 THEN EASE2F=0;
        IF Y13dw20yn<=.z AND Y13dw20ez<=.z AND
EASE1F=5 THEN EASE2F=1;
        IF Y13dw20yn<=.z AND Y13dw20ez<=.z AND
EASE1F=6 THEN EASE2F=2;
        IF Y13dw20yn=0 AND (Y13dw20ez<=.z OR
Y13dw20ez=8) AND EASE1F>0 THEN EASE2F=EASE1F-4;
        IF Y13dw10yn=1 THEN EASE2F=0;

        IF .z<EASE1F<=3 THEN EASE2F=0; ELSE

```

```
IF (EASE1F-3) LT EASE2F THEN EASE2F=EASE1F-3;
```

```
*CODE TO CREATE A SUMMARY MEASURE OF SELF-REPORTED STAIR CLIMBING ABILITY. THIS SCALE WILL Y13NGE FROM 0 TO 9, BUT AT BASELINE IT Y13NGES FROM 4-9;
```

```
CSAINDEX=EASE1F + EASE2F;
```

```
*CODE FOR LIFTING/CARRYING;
```

```
IF Y13dif10=0 AND (Y13ez10lb=8 OR Y13ez10lb<0) THEN EASE10P=5;  
ELSE IF Y13dif10 IN (0,7,8) and Y13ez10lb ne 8 THEN EASE10P=7-Y13ez10lb;  
else if Y13dif10 in (0,7,8) and Y13ez10lb=8 then EASE10P=4;  
IF Y13dif10=1 AND (Y13d10amt=8 OR Y13d10amt<0) THEN EASE10P=.;  
ELSE IF Y13dif10=1 THEN EASE10P=4 - Y13d10amt;  
IF Y13dif10<=.z & Y13d10amt>0 & Y13ez10lb<0 THEN EASE10P=4 - Y13d10amt;  
IF Y13dif10<=.z & Y13d10amt<0 & Y13ez10lb>0 THEN EASE10P=7 - Y13ez10lb;  
IF EASE10P=. & Y13d20lbs NE 1 & Y13d20lbs NE 8 THEN EASE10P=7-Y13ez20lb;  
IF EASE10P=. & Y13d20lbs=0 & (Y13ez20lb LT 0 OR Y13d20lbs=8) THEN EASE10P=6;  
IF EASE10P=. AND Y13dif10=0 AND Y13d20lbs=1 THEN EASE10P=4;  
IF EASE10P=. AND Y13dif10=8 AND Y13d20lbs=1 THEN EASE10P=4;  
IF EASE10P=. & Y13dif10 IN (0,8) & (Y13d20lbs<0 OR Y13d20lbs=8) & (Y13ez20lb<0 OR Y13ez20lb=8) THEN EASE10P=4;
```

```

        IF EASE10P=. AND Y13dif10=0 AND
Y13ez10lb<=.z AND Y13d10amt>0 THEN
            EASE10P=4 - Y13d10amt;

        *IF Y13ez20lb=8 and Y13d20lbs ne 8 THEN
Y13ez20lb=2;
        EASE20P=4-Y13ez20lb;
        if Y13d20lbs=8 and (Y13ez20lb=8 or
Y13ez20lb<0) then EASE20p=1;
        if Y13d20lbs=0 and Y13ez20lb=8 then
EASE20p=EASE10p-4;

        IF Y13d20lbs=1 THEN EASE20P=0;
        IF (0 LE EASE10P LE 3) THEN EASE20P=0;

        IF (Y13d20lbs<=.z OR Y13d20lbs=0) and
(Y13ez20lb<=.z OR Y13ez20lb=8)
            AND EASE10P=4 THEN EASE20P=0;
        IF (Y13d20lbs=0 OR Y13d20lbs<=.z) AND
(Y13ez20lb<=.z OR Y13ez20lb=8)
            AND EASE10P=6 THEN EASE20P=2;
        IF (Y13d20lbs=0 OR Y13d20lbs<=.z) AND
(Y13ez20lb<=.z OR Y13ez20lb=8)
            AND EASE10P=5 THEN EASE20P=1;

        IF (EASE10P-3) LT EASE20P THEN DO;
            EASE20P=EASE10P-3;
            IF EASE20P<0 THEN EASE20P=0;
        END;

        *CODE FOR SUMMARY INDEX;

        LCAINDEX=EASE10P + EASE20P;

        *OTHER FUNCTION MEASURES;

```

```

        IF Y13ezsta=8 and Y13difsta ne 8 THEN
Y13ezsta=2;
        IF Y13dstamt=8 and Y13difsta ne 8 THEN
Y13dstamt=2;
        IF Y13difsta IN (0,7,8) THEN EASEUP=7 -
Y13ezsta;
        IF Y13difsta=1 THEN EASEUP=4 - Y13dstamt;
        if Y13difsta=8 and Y13ezsta=8 then
EASEup=4;
        IF Y13difsta<=.z AND Y13dstamt>0 AND
Y13ezsta<0 THEN EASEUP=4 - Y13dstamt;
        IF Y13difsta=8 AND Y13ezsta<0 AND
Y13dstamt<0 THEN EASEUP=4;
        IF Y13difsta=0 AND Y13ezsta<=.z THEN
EASEUP=5;
        IF Y13difsta=0 & Y13ezsta<=.z & Y13dstamt
>0 THEN EASEUP=4 - Y13dstamt;

        *Formats;
        FORMAT EASE1F EASE2F EASEQM EASE1M EASE10P
EASE20P EASEUP CSAINDEX WKAINDEX
                LCAINDEX SPMISS. /*TIRED1F TIREDQM
YNDK.*;/;

run ;

%macro skip ;

proc freq data=calc.Y13srfcn ;
        tables EASE1F EASE2F EASEQM EASE1M EASE10P
EASE20P EASEUP CSAINDEX WKAINDEX
                LCAINDEX ;
*         tables easelm * Y13dw1myn * Y13dwqmyn *
Y13dw1mez / list missing nocum ;
*         tables Y13dwqmyn Y13dwqmdf Y13mnrs ;
*         tables Y13dwqmyn * (Y13dwqmdf Y13mnrs) / list
missing ;
*         tables (Y13dwqmez Y13dwqmdf) * Y13dwqmyn / list
missing ;

```

```

* tables Y13dwqmyn * Y13dwqmez * Y13dwlmez / list
missing ;
* tables Y13dwqmyn * Y13dwqmdf * Y13dwqmez / list
missing nocum ;
* tables easeqm ;
* tables
EASEQM*Y13dwqmyn*Y13dwqmez*Y13dwqmdf*Y13dwlmyn*Y13d
wlmez*Y13mnrs / list
missing nocum ;
* tables Y13dwlmyn*Y13dwlmez*EASEQM*Y13dwqmyn
/ list missing nocum ;
* tables
EASE1M*Y13dwlmyn*Y13dwlmez*EASEQM*Y13dwqmyn / list
missing nocum ;
* tables WKAINDEX*EASEQM*EASE1M / list
missing ;
* tables
Y13dw10yn*Y13dw10ez*Y13dw20yn*Y13dw20ez*Y13dif /
list missing ;
* tables
EASE1F*Y13dw10yn*Y13dw10ez*Y13dw20yn*Y13dw20ez*Y13d
if / list missing ;
* tables EASE2F*Y13dw20yn*Y13dw20ez*EASE1F /
list missing ;
* tables CSAINDEX*EASE1F*EASE2F / list
missing ;
* tables TIRED1F*FADW10WX ;
* tables
EASE10P*Y13dif10*Y13ez10lb*Y13d10amt*Y13d20lbs*Y13e
z20lb / list missing ;
* tables EASE20P*Y13ez20lb*Y13d20lbs*EASE10P
/ list missing ;
* tables LCAINDEX*EASE10P*EASE20P / list
missing ;
* tables EASEUP*Y13difsta*Y13ezsta*Y13dstamt
/ list missing ;
* tables EASEHHW*Y13difHW*FAEZHW*FADHWAMT /
list missing ;
* tables Y13dwqmez Y13dwlmyn Y13dwlmez ;
* tables Y13dw10yn Y13dif ;

```

```

*         tables Y13dw10ez Y13dw20yn Y13dw20ez ;
*         tables Y13difsta ;
*         tables Y13dif10 Y13d10amt Y13ez10lb
Y13d20lbs Y13ez20lb ;
*         format EASEQM Y13dwqmyn Y13dwqmez Y13dwqmdf
Y13dw1myn Y13dw1mez Y13mnrns
                EASE1M WKAINDEX /*FADWQMT2*/
                EASE1F Y13dw10yn Y13dw10ez Y13dw20yn
Y13dw20ez Y13dif
                EASE2F CSAINDEX /*TIRED1F FADW10WX */
                EASE10P Y13dif10 Y13ez10lb Y13d10amt
Y13d20lbs Y13ez20lb
                EASE20P LCAINDEX
                /*EASEHHW Y13difHW FAEZHW FADHWAMT*/;
                FORMAT EASE1F EASE2F EASEQM EASE1M EASE10P
EASE20P EASEUP CSAINDEX WKAINDEX
                LCAINDEX SPMISS. /*TIRED1F TIREDQM
YNDK.*;/;
run ;

proc contents data=calc.Y13srfcn ;
    title 'Year 13 - Self Reported Physical
Functioning' ;
run ;
%mend skip ;
%skip;

***check for duplicates***;
data dupes;
    set calc.Y13srfcn;
    by habcid;
    if not(first.habcid and last.habcid);
run;

proc print;
title4 'Duplicates in SRFCN Y13';
run;

/*

```

```
libname calc_y10 '\\Fu-hsing-
c\habc\habc_sas\calculated variables\datasets\Year
10';
proc freq data=calc_y10.Y10srfcn ;
    tables EASE1F EASE2F EASEQM EASE1M EASE10P
EASE20P EASEUP CSAINDEX WKAINDEX
    LCAINDEX ;
    title 'Year 10 - Self Reported Physical
Functioning' ;
    title2 "For comparison";
run ;
proc contents data=calc_y10.Y10srfcn ;
run ;
*/
```

Appendix III
Smoking Calculated Variables

Using Clinic Visit, Home Visit and Annual Telephone Interview Variables

Investigator Name: Stephen Kritchevsky

Variable	Descriptive Title	Detailed Description	How variable is calculated	How to handle missing or special values	Value labels
SMK13	Smoking Status at Year 13	0 is never smoker; 1 is former smoker; 2 is current smoker	If Y13smoke=1 then SMK12=1; Else if Y13smoke=0 and smk1=0 and (smk3=0 or smk3 le .z) and (smk5=0 or smk5 le .z) and (smk8=0 or smk8 le .z) and (smk9=0 or smk9 le .z) and (smk10=0 or smk10 le .z) and (smk11=0 or smk11 le .z) and (smk12=0 or smk12 le .z) then SMK13=0; Else if Y13smoke=0 and (smk1 in (1,2) or smk3 in (1,2) or smk5 in(1,2) or smk8 in (1,2) or smk9 in (1,2) or smk10 in (1,2) or smk11 in (1,2) or smk12 in (1,2) then SMK13=2; Else if Y13smoke le .z or Y13smoke in (7,8) then SMK13=.A; Else SMK13=.M;	If Y13SMOKE ≤ .z or Y13SMOKE in (7,8) then SMK13=.A	0=Never 1=Current 2=Former

```

*****
*****
* Create calculated variables requested by
SKritchevsky for:
* SMK13      Smoking status at Year 13
*
*
* Todd Glasser 01/06/2011 - Modified year 11
program for use in Year 13
*****
***** ;

*DM LOG 'CLEAR' ; *DM OUTPUT 'CLEAR' ;

%include '\\Fu-hsing-
c\habc\habc_sas\programs\initV8_ase.sas';
libname calc '\\Fu-hsing-c\habc\habc_sas\calculated
variables\datasets\Year 13';

*options ls=141 ps=46 formchar='|----|+|----+=|'-
/\<>*' nodate nofmterr pageno=1 nocenter;

***** Smoking status at year 13;
data Y13smoke ;
    set daf.y13visit (keep=habcid y13smoke);
run;

data bl; set current.y1calc (keep=habcid
smk1);run;
data y3; set current.y3calc (keep=habcid
smk3);run;
data cm; set current.y5calc (keep=habcid
smk5);run;
data y8; set current.y8calc (keep=habcid
smk8);run;
data y9; set current.y9calc (keep=habcid
smk9);run;
data Y10; set current.Y10calc (keep=habcid
smk10);run;

```

```

data Y11; set current.Y11calc (keep=habcid
smk11);run;
data Y12; set current.Y12calc (keep=habcid
smk12);run;

data smoke;
merge bl y3 cm y8 y9 y10 y11 y12 y13smoke;
by habcid;
label SMK13 = 'Smoking status at Year 13 visit';
format SMK13 ncffmt. ;
if Y13smoke le .z or Y13smoke in (7,8) then
SMK13=.A;
else if Y13smoke=1 then SMK13=1;
*** Use smoking status at prior years to
determine never/former status ***;
else if Y13smoke=0 and smk1=0 and (smk3=0 or smk3
le .z) and (smk5=0 or smk5 le .z)
and (smk8=0 or smk8 le .z) and (smk9=0 or smk9
le .z) and (smk10=0 or smk10 le .z) and (smk11=0 or
smk11 le .z)
and (smk12=0 or smk12 le .z) then SMK13=0;
else if Y13smoke=0 and (smk1 in (1,2) or smk3 in
(1,2) or smk5 in(1,2) or smk8 in (1,2) or
smk9 in (1,2) or smk10 in (1,2) or smk11 in
(1,2) or smk12 in (1,2)) then SMK13=2;
else SMK13=.M;
** 0=never, 1=current, 2=former **;
run;

data calc.Y13smoke;
set smoke (keep=habcid SMK13);
run;

%macro skip ;

proc contents data=calc.Y13smoke;
run;
proc freq data=calc.Y13smoke;
tables SMK13;
run;

```

```
proc freq data=smoke;
  table smk13*Y13smoke / list missing;
  tables
smk13*y13smoke*SMK10*smk9*smk8*smk5*smk3*smk1 /
list missing nopercnt;
  format _all_;
run;

***check for duplicates***;
data dupes;
  set calc.Y13smoke;
  by habcid;
  if not(first.habcid and last.habcid);
run;
proc print data=dupes;
title4 'Duplicates in Y13Smoke';
run ;

%mend skip ;
```

Appendix IV
Physical Activity Calculated Variables
Using Clinic Visit, Home Visit and Annual Telephone Interview Variables

Investigator Name: Eleanor Simonsick

Step 1: assign met estimates to each activity

Met estimates derive from Ainsworth BE, Haskell WL, Leon AS, Jacobs DR, Montoye HJ, Sallis JF, Paffenbarger RS. Compendium of physical activities: classification of energy costs of human physical activities. Med Sci Sports Exerc 25(1):71-80.

Variable	General description	Detailed Description	How variable is calculated	How to handle missing or special values	Value labels
Y13FSKKWK	kcal/kg/week climbing stairs	Assigns 4.0 kcal/kg/hour of stair climbing plus an additional 1.0 kcal/kg/hour carrying a load (Y10 Qaire, page 12, Q19). (Assumes 1 flight up/down takes 30 seconds.)	If Y13FS7DAY=1 then Y13FSKKWK = 4.0 x Y13FSNUM/120 +1.0 x Y13FSLOAD/120; If Y13FS7DAY=0 then Y13FSKKWK=0;	Correction for outliers: If Y13FSNUM>210 then Y13FSNUM=210; if Y13FSLOAD>210 then Y13FSLOAD=210; If Y13FS12MO≤.z and Y13FS7DAY≤.z then Y13FSKKWK=.; if (Y13FS12MO in (0,7,8)) and (Y13FS7DAY≤.z or Y13FS7DAY=8) then Y13FSKKWK=0; if Y13FS12MO=1 and (Y13FS7DAY≤.z or Y13FS7DAY=8) then Y13FSKKWK=0; if Y13FSKKWK<0 then do: if Y13FS7DAY=1 and Y13FSNUM > 0 and (Y13FSLOAD≤.z or Y13FSLODK=-1) then Y13FSLOAD=0; if Y13FS7DAY=1 and (Y13FSNUM≤.z or Y13FSNUMD=-1) and Y13FSLOAD > 0 then Y13FSNUM=Y13FSLOAD	kkcal/kg/week

Variable	General description	Detailed Description	How variable is calculated	How to handle missing or special values	Value labels
Y13MCKKWK	kcal/kg/week doing major chores	Assigns 3.5 kcal/kg/hour doing major chores (Y10 Qaire, page 11, Q17). Note: this question combines activities listed separately at baseline. Y13MCKKWK is therefore equivalent to the sum of baseline variables FPPAKKWK and FPHCKKWK.	3.5 x Y13HCTIM/60 if Y13HC7DAY=0 then Y13MCKKWK=0;	If Y13HC12MO≤.z and Y13HC7FAY≤.z then Y13MCKKWK=.; if (Y13HC12MO=0 or Y13HC12MO in (7,8) and (Y13HC7DAY≤.z or Y13HC7DAY=0) then Y13MCKKWK=0; if Y13HC12MO=1 and Y13HC7DAY≤.z then Y13MCKKWK=0; if Y13MCKKWK<0 then do: if Y13HC12MO=1 and Y13HC7DAY=1 and (Y13HCTIM≤.z or Y13HCDK=-1) then Y13HCTIM=120 (median value for non-missings at baseline; used here for consistency)	kkcal/kg/week
Y13TWKKWK	kcal/kg/week total walking	Assigns 4.0 kcal/kg/hour walking briskly, 3.0 to walking at moderate pace, and 2.0 to strolling (Y10 Qaire, page 12, Q18 - all walking)	If Y13EW7DAY=1 then Y13TWKKWK= 4.0 x Y13EWTIME* Y13EWTIM/60 if Y13EWPACE=1, 3.0 x Y13EWTIME* Y13EWTIM/60 if Y13EWPACE=2, 2.0 x Y13EWTIME* Y13EWTIM/60 if Y13EWPACE=3 if Y13EW7DAY=0 then Y13TWKKWK=0;	If Y13EW12MO≤.z and Y13EW7RCY≤.z then Y13TWKKWK=.; if (Y13EW12MO in (0,7,8) and Y13EW7RCY≤.z then Y13TWKKWK=0; if Y13EW12MO=1 and Y13EW7RCY≤.z then Y13TWKKWK=0; if Y13TWKKWK<0 then do: if Y13EWTIME > 0 and Y13EWTIM > 0 and (Y13EWPACE≤.z or Y13EWPACE=8) then Y13TWKKWK=3.0 x Y13EWTIME*Y13EWTIM/60 (median value at baseline; used here for consistency); if Y13EWTIME > 0 and (Y13EWTIM≤.z or Y13EWTDK=-1) then Y13EWTIM=35 (median value at baseline; used here for consistency); if (Y13EWTIME≤.z or Y13EWTMDK=-1) and Y13EWTIM > 0 then Y13EWTIME=4 (median value at baseline; used here for consistency)	kkcal/kg/week

Step 2: create composite variables

Variable	General description	Detailed Description	How variable is calculated	How to handle missing or special values	Value labels
WSKKWK	kcal/kg/week - walking + stairs	Sum of exercise walking, other walking, and stair climbing variables	Y13TWKKWK + Y13FSKKWK	Imputed version of component variables used; SAS SUM function used to sum all non-missing components	kkcal/kg/week
WALKTIME	minutes walking/week	Adds minutes exercise walking and other walking in past week	WALKTIME=(Y13EWTIME x Y13EWTIM)	If Y13TWKKWK=0 then WALKTIME=0;	min
BKTWTIME	minutes walking briskly/week	Sum of minutes brisk exercise walking plus minutes brisk other walking	If Y13EWPACE=1 then BKTWTIME= Y13EWTIME x Y13EWTIM; If Y13EWPACE>1 then BKTWTIME=0; If Y13EWKKWK=0 then BKTWTIME=0;		min

Step 3: create categorical variables

Variable	General description	Detailed Description	How variable is calculated	How to handle missing or special values	Value labels
WALKCAT	minutes walking/week category	Categories of time spent on any walking per week	If WALKTIME=0 then WALKCAT=0; if 0<WALKTIME<150 then WALKCAT=1; if WALKTIME ≥ 150 then WALKCAT=2	Imputed version of component variables used	0= 0 min/wk 1= between 0 and 150 min/wk 2= more than 150 min/wk
BRISK90	walks briskly ≥ 90 min/week	Dichotomous variable for walking briskly ≥90 min/week vs <90 min/week	If BKTWTIME=0 then BRISK90=0; if 0<BKTWTIME<90 then BRISK90=0; if BKTWTIME ≥ 90 then BRISK90=1	Imputed version of component variables used	0=No 1=Yes
BRISK180	walks briskly ≥ 180 min/week	Dichotomous variable for walking briskly ≥180 min/week vs <180 min/week	If BKTWTIME=0 then BRISK180=0; if 0<BKTWTIME<180 then BRISK180=0; if BKTWTIME ≥ 180 then BRISK180=1	Imputed version of component variables used	0=No 1=Yes

```

*****
*****
*   Saved as   \\Fu-hsing-c\HABC\HABC_SAS\Calculated
Variables\Programs\Year 13\Phact_Y13.sas
*
*   Todd Glasser 01-06-2011 - Modified year 11
program for use in Year 13
*****
***** ;

%include '\\Fu-hsing-
c\habc\habc_sas\programs\initV8.sas';
libname daf 'M:\VSS_HABC\ToddFiles\Year13.13.5';
libname calc '\\Fu-hsing-c\habc\habc_sas\calculated
variables\datasets\Year 13';

data calc.Y13phact (KEEP=HABCID Y13MCKKWK Y13FSKKWK
Y13TWKKWK WALKTIME WALKCAT BKTWTIME

BRISK180 BRISK90 WSKKWK );
    set daf.Y13visit;
    by habcid;

    Label Y13MCKKWK='KCAL/KG/WEEK DOING MAJOR
CHORES'
          Y13FSKKWK='KCAL/KG/WEEK CLIMBING STAIRS'
          Y13TWKKWK='KCAL/KG/WEEK TOTAL WALKING'
          WALKTIME='MINUTES WALKING/WEEK'
          WALKCAT='MINUTES WALKING/WEEK CATEGORY'
          BKTWTIME='MINUTES WALKING BRISKLY/WEEK'
          BRISK90='WALKS BRISKLY >= 90 MIN/WEEK'
          BRISK180='WALKS BRISKLY >= 180 MIN/WEEK'
          WSKKWK='KCAL/KG/WEEK - WALKING + STAIRS';

    *****----- MAJOR CHORES (17);
    IF Y13HC12MO<=.Z AND Y13HC7DAY<=.Z THEN
Y13MCKKWK=. ;
    IF (Y13HC12MO=0 OR Y13HC12MO IN (7,8)) AND
(Y13HC7DAY<=.Z OR Y13HC7DAY=0) THEN Y13MCKKWK=0;
    IF Y13HC7DAY=0 THEN Y13MCKKWK=0;

```

```

    IF Y13HC12MO=1 AND Y13HC7DAY<=.z THEN
Y13MCKKWK=0;
    IF Y13HC7DAY=1 THEN Y13MCKKWK=3.5*Y13hctim/60;
    *IMPUTED MISSING CODE;
    IF Y13MCKKWK LT 0 THEN DO;
    IF Y13HC12MO=1 AND Y13HC7DAY=1 AND (Y13hctim<=.z
OR Y13HCDK=-1) THEN Y13hctim=120;
    IF Y13HC7DAY=1 THEN Y13MCKKWK=3.5*Y13hctim/60;
    END;

    *****----- WALKING FOR EXERCISE (18);
    IF Y13EWPACE=1 THEN Y13EWMET=4.0;
    IF Y13EWPACE=2 THEN Y13EWMET=3.0;
    IF Y13EWPACE=3 THEN Y13EWMET=2.0;
    IF Y13EW12MO<=.z AND Y13EW7DAY<=.z THEN
Y13EWKKWK=. ;
    else do ;
        IF (Y13EW12MO IN (0,7,8)) AND
(Y13EW7DAY<=.z) THEN Y13EWKKWK=0;
        IF Y13EW7DAY=0 THEN Y13EWKKWK=0;
        IF Y13EW12MO=1 AND Y13EW7DAY<=.z THEN
Y13EWKKWK=0;
        IF Y13EW7DAY=1 THEN
Y13EWKKWK=Y13EWMET*Y13EWTIME*Y13EWTIM/60;
        end ;
        *IMPUTED MISSING CODE;
        IF Y13EWKKWK LT 0 THEN DO;
            IF Y13EWTIME > 0 AND Y13EWTIM > 0 AND
(Y13EWPACE<=.z OR Y13EWPACE=8) THEN Y13EWMET=3.0;
            IF Y13EWTIME > 0 AND (Y13EWTIM<=.z OR
Y13EWTDK=-1) THEN Y13EWTIM=35;
            IF (Y13EWTIME<=.z OR Y13EWTMDK=-1) AND
Y13EWTIM > 0 THEN Y13EWTIME=4;
            IF Y13EW7DAY=1 THEN
Y13EWKKWK=Y13EWMET*Y13EWTIME*Y13EWTIM/60;
        END;

    *****----- Fix calculation of BKEWTIME EK
9/25/03;
    IF Y13EWPACE=1 THEN BKEWTIME=Y13EWTIME*Y13EWTIM;

```

```

ELSE IF Y13EWPACE>1 THEN BKEWTIME=0;
ELSE IF Y13TWKKWK=0 THEN BKEWTIME=0;
BKTWTIME=BKEWTIME;
IF BKTWTIME=0 THEN BRISK90=0;
ELSE IF 0<BKTWTIME<90 THEN BRISK90=0;
ELSE IF BKTWTIME GE 90 THEN BRISK90=1;
IF BKTWTIME=0 THEN BRISK180=0;
ELSE IF 0<BKTWTIME<180 THEN BRISK180=0;
ELSE IF BKTWTIME GE 180 THEN BRISK180=1;

*****----- New variable for total walking;
Y13TWKKWK=Y13EWKKWK;
IF Y13TWKKWK=0 THEN WALKTIME=0;
ELSE IF Y13TWKKWK>0 THEN
WALKTIME=Y13EWTIME*Y13EWTIM;
IF WALKTIME=0 THEN WALKCAT=0;
ELSE IF 0<WALKTIME<150 THEN WALKCAT=1;
ELSE IF WALKTIME GE 150 THEN WALKCAT=2;

*****----- CLIMBING STAIRS (19);
* ASSUME UP/DOWN 1 FLIGHT TAKES 30 SECONDS AND
ONE ADDITIONAL MET FOR CARRYING A LOAD;
* Cutoff for flights of stairs - only for
purpose of calculated variables ***;
*** See email from MLE dated 4/6/01
***;
IF Y13FSNUM>210 then Y13FSNUM=210;
IF Y13FSLOAD>210 THEN Y13FSLOAD=210;
IF Y13FS12MO<=.z AND Y13FS7DAY<=.z THEN
Y13FSKKWK=. ;
IF (Y13FS12MO=0 OR Y13FS12MO=8 OR Y13FS12MO=7)
AND (Y13fs7day=8 OR Y13fs7day<=.z) THEN
Y13FSKKWK=0;
IF Y13fs7day=0 THEN Y13FSKKWK=0;
IF Y13FS12MO=1 AND (Y13fs7day<=.z OR
Y13fs7day=8) THEN Y13FSKKWK=0;
IF Y13fs7day=1 THEN Y13FSKKWK=(4.0*Y13FSNUM/120)
+ (1.0*Y13FSLOAD/120);
*IMPUTED MISSING CODE;
IF Y13FSKKWK LT 0 THEN DO;

```

```

    IF Y13fs7day=1 AND Y13FSNUM GT 0 AND
(Y13FSLOAD<=.z OR Y13FSLODK=-1) THEN Y13FSLOAD=0;
    IF Y13fs7day=1 AND (Y13FSNUM<=.z OR Y13FSNUMD=-
1) AND Y13FSLOAD GT 0 THEN Y13FSNUM=Y13FSLOAD;
    IF Y13fs7day=1 THEN Y13FSKKWK=(4.0*Y13FSNUM/120)
+ (1.0*Y13FSLOAD/120); END;

```

```

*****----- WALKING AND STAIRS;
WSKKWK=SUM(OF Y13FSKKWK Y13TWKKWK);

```

```

format brisk180 brisk90 yndk. walkcat walk9x.;
run;

```

```

%MACRO SKIP ;
proc contents data=CALC.Y13phact varnum;
  title4 'Final Dataset';
run;
proc freq data=CALC.Y13phact;
tables brisk90 brisk180 walkcat/missing;
run;
proc univariate data=CALC.Y13phact;
var bktwtime Y13mckkwk Y13fskkwk Y13twkkwk walktime
wskkwk;
run;
%MEND SKIP ;

```

```

***check for duplicates***;
data dupes;
  set CALC.Y13phact;
  by habcid;
  if not(first.habcid and last.habcid);
run;

```

```

proc print;
title4 'Duplicates in phact Y13';
run;

```