Atherosclerosis Risk in Communities Study

---

# Cohort Exam Visit 9 NCS

# DERIVE91_NP Derived Variable Dictionary (v3.0)

## December 2024

Prepared by the Collaborative Studies Coordinating Center

ARIC DERIVE91_NP Derived Variable Dictionary

# Table of Contents

## NEW OR CHANGED FROM PREVIOUS DISTRIBUTION

This table describes the changes to the last published DERIVE91 dictionary. As the dataset undergoes modifications, this table will describe the updates made to the previously distributed dataset.

| Modification Date | Variable Name | Reason(s) for Change |
|---|---|---|
| **7/21/2023** | Administrative:<br>V9DATE91<br>LASTFUINTERVIEW_DATE91<br><br>Socio-Demographic:<br>BIRTHDAT<br>BIRTHDAT91<br><br>Laboratory Analytes:<br>fasting_time<br>FAST0891<br>FAST1291<br>EGFR91<br><br>Neurocognitive Study:<br>PRORATEDMMSE91<br>FAQ91<br>CESD91<br>LOWPROMMSE91<br>LOWMEMDOM91<br>LOWEFDOM91<br>LOWLANGDOM91<br>SIGCOGDECLINE92 | Variables with dates are removed due to changes in sIRB requirements.<br><br>Laboratory analytes variables are removed from DERIVE91 due to changes in sIRB consent requirements. The variables will be found in a future distribution of a derived laboratory dataset.<br><br>Neurocognitive study variables are removed from DERIVE91 and will be found in a future distribution of the V9NCSCOG91 dataset. |
| **7/21/2023** | Administrative:<br>V9DATE91_FollowUpDays<br>V9DATE91_year<br>LASTFUINTERVIEW_DAT_FollowUpDays<br>LASTFUINTERVIEW_DAT_year<br><br>Disease Prevalence:<br>DIABTS96<br>DIABTS97<br><br>Physical Function:<br>V7V9WTDELTAP91<br>TMW_TOTFT91<br>TMW_COMPSPEEDFTPERSEC91<br>TMW_COMPSPEEDMPERSEC91<br>TMW_NONCOMPSECWALK91<br>TMW_NONCOMPSPEEDFTPERSEC91<br>TMW_NONCOMPSPEEDMPERSEC91<br>LOWENERGYCOMP91<br>WTLOSSCOMPA91<br>WTLOSSCOMPB91<br>FRAILTY91a<br>FRAILTY91b<br>FRAILTY93<br>FRAILTY94 | Visit 9 exam date and date of last completed follow-up interview variables are replaced with variables indicating the year and number of follow up days after visit 1.<br><br>Disease prevalence variables are added.<br><br>Additional physical function variables are provided. |

| | | |
|---|---|---|
| **08/09/2024** | Disease Prevalence:<br>MDDXMI92<br>HXOFMI92<br>PRVCHD94<br>PRVSTR92<br>HYPERT94<br>HYPERT95<br>HYPERT96<br>HYPERT97<br>PRVCHD91<br>PRVCHD93<br>PRVSTR91<br>PREVDEFHF91<br>PREVDEFPOSSHF91<br><br>Neurocognitive Study:<br>FAQ91<br>CESD91<br><br>Physical Variables and Indicators:<br>AGENATMENOPAUSEF<br>AGESRGMENOPAUSEF | Disease Prevalence variables using surveillance data through incident year 2021 are included in DERIVE.<br><br>Variables FAQ91 and CESD91 are incorporated back into DERIVE datasets.<br><br>Physical Variables and Indicators are added to DERIVE datasets. |
| **12/18/24** | Administrative<br>V9PROXYATEXAM91<br>V9PROXYDATA91 | Two new administrative variables have been added to measure the role of proxies for participants at Visit 9.<br><br>The disease prevalence variables with surveillance data components may be considered complete for analytic purposes except for the caveat described in the overview below. |

## 1. OVERVIEW

The DERIVE91_241218_NP dataset has 2105 records, one for each participant who completed Stage 1 at Visit 9. The purpose of this dataset is to provide ARIC collaborators widely used, verified derived variables, many of which are consistent with variables derived at prior visits.

The dataset naming conventions are as follows: The dataset name retains the retrieval date (ex: DERIVE91_241218_NP) until the dataset is considered final, frozen. After a dataset is frozen, the retrieval date is dropped from the dataset name (ex: DERIVE91_NP). The first digit in the dataset name refers to the visit number. The second digit in the dataset name is incremented in number when the current dataset undergoes significant changes. Datasets with "NP" in the name have date variables removed; instead, date variables are replaced with a variable that calculates the number of follow up days after Visit 1 date and the year of the original date variable. The variable naming convention is similar: Across-visit variables have identical names except for the second to last digit in the variable name, which represents the visit number (ex: GENDER91 at Visit 9 vs. GENDER8T1 at Visit 8 Telephone, or 8T). The last digit in the variable name identifies the definition version of a variable.

Most of the variables are derived directly from the data collected at the visit. However, some variables use ARIC cohort surveillance and ARIC follow-up data in their definitions. DERIVE91 will be final, frozen after the surveillance datasets are complete for events in 2022 and following the updates to the medication variables.

Note: Due to lack of access to records at one large Jackson hospital in 2021-22, we excluded from the final datasets any hospitalizations for Jackson participants for 2021-22. This modification manifests itself in two different ways among the distributed cohort surveillance data.

1. The hospitalization records were not available for these participants in 2021-22 which will affect the disease prevalence derived variables in DERIVE91.
2. For files commonly used for survival analysis, the value for the administrative censoring variable (CENSDAT7) was set to be December 31, 2020 for Jackson participants, instead of the standard value of December 31, 2022 for participants from the other three field centers.

Laboratory variables and neurocognitive variables are separated from DERIVE, and will be found in future distributions, as described in the 'New or Changed from PREVIOUS Distribution' section.

Data on smoking status was not collected at Visit 9. For the most current data on smoking status, please refer to variables FORSMK72 and EVRSMK72 in the DERIVE71 dataset. These variables are current as of December 2019.

## 2. ADMINISTRATIVE

### 2.1 SUBJECTID (ARIC Subject ID (CIR))

Type: Character; length: $7.

### 2.2 ID (ARIC ID - same as SUBJECTID)

Description: The historical participant identifier from visits 1-4 is ID. The value of ID is the same value as SUBJECTID. Use ID when merging visit 9/NCS stage 1 data with datasets from previous visits necessary for longitudinal analyses.

Type: Character; length: $7.

Algorithm: ID=SUBJECTID

Source variable(s): SUBJECTID

### 2.3 CENTER (Field Center)

Description: Character variable with four possible values derived from the enrollment site:
F: Forsyth County, North Carolina
J: The city of Jackson, Mississippi
M: Selected northwestern suburbs of Minneapolis, Minnesota
W: Washington County, Maryland

Type: Character; length: $1.

Algorithm: CENTER = First letter of the subject ID

Source variable(s): SUBJECTID

### 2.4 V9CENTER (Visit 9 Field Center)

Description: Character variable with four possible values derived from the enrollment site:
F: Forsyth County, North Carolina
J: The city of Jackson, Mississippi
M: Selected northwestern suburbs of Minneapolis, Minnesota
W: Washington County, Maryland

Type: Character; length: $1.

Algorithm: The value of V9CENTER is the same as CENTER unless the ARIC study participant has relocated geographically and moved into another field center at visit 9. In that instance, the value of V9CENTER is the value of the field center where the participant was seen.

## 2.5 V9DATE91_FollowUpDays (Days of follow up from visit 1 to Visit 9 Date, NCS Stage 1)

Description: The number of days between visit 1 and the date of the participant's visit 9 exam, stage 1. The visit 9 exam date is taken from the first present form in V9 / NCS in the following order: SBP, BIO9, ANT.

Type: Numeric

Algorithm: If the eventname from the SBP form is "V9 / NCS" then V9DATE91_FollowUpDays=SBP0a - visit 1 date;

Else if the eventname from the BIO9 form is "V9 / NCS" then V9DATE91_FollowUpDays=BIO0a - visit 1 date;

Else if the eventname from the ANT form is "V9 / NCS" then V9DATE91_FollowUpDays=ANT0a - visit 1 date;

Source variable(s): STAGE_1_COMPLETE, ANT0a, BIO0a, SBP0a, visit 1 date

## 2.6 V9DATE91_year (Year of Visit 9 date)

Description: Year of the participant's visit 9 exam, stage 1. The visit 9 exam date is taken from the first present form in V9 / NCS in the following order: SBP, BIO9, ANT.

Type: Numeric

Algorithm: If the eventname from the SBP form is "V9 / NCS" then V9DATE91_year=year of SBP0a;

Else if the eventname from the BIO9 form is "V9 / NCS" then V9DATE91_year=year of BIO0a;

Else if the eventname from the ANT form is "V9 / NCS" then V9DATE91_year=year of ANT0a;

Source variable(s): STAGE_1_COMPLETE, ANT0a, BIO0a, SBP0a

### 2.7 RES_OTH (Restrictions on Other Procedures)

Description: The derived informed consent file, ICTDER05, includes information and dates of final consents for ARIC participants. Change in consent status is recorded in ARIC follow-up and visit 9 using the ICT form. RES_OTH indicates the types of restriction on other procedures. We request that the investigators exclude appropriate records with partial restrictions prior to data analysis.

Type: Character; length: $130.

Algorithm: if ICT3=1 and ICT4=1 and ICT7=1 then res_OTH='Full Consent';
else if ICT3=1 and ICT4=1 and (ICT7=0 or missing(ICT7)) then res_OTH='Not for Profit';
else if ICT3=1 and ICT4=0 then res_OTH='ARIC Only';
else if ICT3=0 and ICT4=0 then res_OTH='No Consent';

Source variable(s): ICT3, ICT4, ICT7

### 2.8 RES_DNA (Restrictions on DNA Usage)

Description: The derived informed consent file, ICTDER05, includes information and dates of final consents for ARIC participants. Change in consent status is recorded in ARIC follow-up and visit 9 using the ICT form. The variable RES_DNA indicates the type of restriction on DNA use. We request that the investigators exclude appropriate records with partial restrictions prior to data analysis.

Type: Character; length: $130.

Algorithm: if ICT5=1 and ICT6=1 and ICT7=1 then RES_DNA='Full Consent';
else if ICT5=1 and ICT6=1 and (ICT7=0 or missing(ICT7)) then RES_DNA='Not for Profit';
else if ICT5=1 and ICT6=0 then RES_DNA='ARIC Only';
else if ICT5=0 and ICT6=0 then RES_DNA='No use/storage DNA';

Source variable(s): ICT5, ICT6, ICT7

## 2.9 LASTFUINTERVIEW_DAT_FollowUpDays (Days of follow up from visit 1 to Date of last completed follow-up interview)

Description: The number of days between visit 1 and the date of the participant's last completed follow-up interview where an actual contact was made, prior to November 30, 2022.

Type: Numeric

Algorithm: Days between Visit 1 and the max value of AFUcomp1_A in the composite follow-up dataset among the records for a single ID where AFUcomp2_A indicates that the interview was accomplished (AFUcomp2_A in ('A','C','D')) and the date preceded November 30, 2022.

Source variable(s): AFUcomp1_A, AFUcomp2_A, visit 1

## 2.10 LASTFUINTERVIEW_DAT_year (Year of Date of last completed follow-up interview)

Description: Year of the participant's last completed follow-up interview where an actual contact was made, prior to November 30, 2022.

Type: Numeric

Algorithm: Year of the max value of AFUcomp1_A in the composite follow-up dataset among the records for a single ID where AFUcomp2_A indicates that the interview was accomplished (AFUcomp2_A in ('A','C','D')) and the date preceded November 30, 2022.

Source variable(s): AFUcomp1_A, AFUcomp2_A

## 2.11 STAGE_1_TYPE (Type of Stage 1 Exam)

Description: Categorical variable that describes the participant's type of Stage 1 exam.

Format: A=Full, B=Abbreviated, C=Home, D=Long Term Care Facility.

Type: Character; length=$1.

Algorithm: STAGE_1_TYPE=RTS17

Source variable(s): RTS17

## 2.12 V9PROXYATEXAM91 (Proxy present at V9)

Description: Proxy present at V9

Format: Y=Yes, N=No

Type: Character; length=$1.

Algorithm: V9PROXYATEXAM91=RTS18

Source variable(s): RTS18

## 2.13 V9PROXYDATA91 (Proxy contributed to data collection at V9)

Description: Proxy contributed to data collection at V9

Format: Y=Yes, N=No

Type: Character; length=$1.

Algorithm: If RTS18="Y" and RTS18a="Y" then V9PROXYDATA91="Y";
If RTS18="N" or RTS18a= "N" then V9PROXYDATA91="N";
Else V9PROXYDATA91= missing;

Source variable(s): RTS18, RTS18a

## 3. SOCIO-DEMOGRAPHIC

### 3.1 GENDER (Sex)

Description: Categorical variable that describes the participant's gender: M=Male, F=Female.

Type: Character; length=$1.

Algorithm: GENDER = GENDER from DERIVE13

Source variable(s): [DERIVE13] GENDER

### 3.2 GENDER91 (Corrected Gender (V1CORGE1))

Description: Categorical variable that describes the participant's gender: M=Male, F=Female. Incorrect values for the variable GENDER were identified following the initial data collection on the ARIC cohort. The ARIC Executive Committee has recommended continuing to use the uncorrected variable (GENDER) for Visit 1 and longitudinal analyses. The corrected version could be used for cross-sectional analyses other than Visit 1 and should be decided by the Investigator.

Type: Character; length=$1.

Algorithm: GENDER91 = V1CORGE1

Source variable(s): V1CORGE1

### 3.3 RACEGRP (Race)

Description: Categorical variable which describes the participant's race: A=Asian, B=Black, I=Native American, W=White.

Type: Character; length=$1.

Algorithm: RACEGRP = RACEGRP from DERIVE13

Source variable(s): [DERIVE13] RACEGRP

### 3.4 RACEGRP91 (Corrected Race (V1CORRA1))

Description: Categorical variable which describes the participant's race: A=Asian, B=Black, I=Native American, W=White. Incorrect values for the variable RACEGRP were identified following the initial data collection on the ARIC cohort. The ARIC Executive Committee has recommended continuing to use the uncorrected variable (RACEGRP) for Visit 1 and longitudinal analyses. The corrected version could be used for cross-sectional analyses other than Visit 1 and should be decided by the Investigator.

Type: Character; length=$1.

Algorithm: RACEGRP91 = V1CORRA1

Source variable(s): V1CORRA1

### 3.5 V9AGE91 (Visit 9 Age)

Description: Participant's age at the time of the visit 9 exam calculated from the date of birth originally reported for the participant.

Type: Numeric

Algorithm: If visit 9 exam occurred and date of birth>.z then V9AGE91 = floor((intck('month', date of birth, visit 9 date)-(day(visit 9 date) < day(date of birth)))/12);

Source variable(s): Date of birth reported at visit 1, visit 9 date

### 3.6 V9AGE92 (Corrected Visit 9 Age)

Description: Participant's age at the time of the visit 9 exam calculated from the corrected date of birth. The corrected date of birth corrects known errors in the date of birth values reported in the initial data collection on the ARIC cohort. The ARIC Executive Committee has recommended continuing to use the uncorrected variable (V9AGE91) for Visit 1 and longitudinal analyses. The corrected version could be used for cross-sectional analyses other than Visit 1 and should be decided by the Investigator.

Type: Numeric

Algorithm: If visit 9 exam occurred and corrected date of birth>.z then V9AGE92=floor((intck('month', corrected date of birth, visit 9 date)-(day(visit 9 date) < day(corrected date of birth)))/12);

Source variable(s): Corrected date of birth, visit 9 date

## 4. ANTHROPOMETRY AND BLOOD PRESSURE

### 4.1 BMI91 (V9 Body Mass Index in Kg/m$^2$)

Description: Body mass index [Weight (kg)] / [Height (cm) / 100]$^2$

Type: Numeric

Algorithm: If missing (V6IN129) or missing (ANT4) then BMI91=missing;
Else BMI91= ANT4/(V6IN129/100)$^2$

Source variable(s): V6IN129 (Last Measured Height in cm), ANT4

### 4.2 WSTHPR91 (V9 Waist-to-Hip Ratio)

Description: Ratio of waist girth to hip girth

Type: Numeric

Algorithm: If missing(ANT10a) or missing(ANT10b) or ANT10b=0 then
WSTHPR91=missing
Else WSTHPR91=(ANT10a/ANT10b)

Source variable(s): ANT10a, ANT10b

### 4.3 SYSTOLIC91 (V9 Mean Systolic BP of 2nd and 3rd Measurements)

Description: Mean of 2$^{nd}$ and 3$^{rd}$ systolic blood pressure measurements, consistent with V1 through V5

Type: Numeric

Algorithm: SYSTOLIC91=mean of SBP8 and SBP11
If SYSTOLIC91=missing then SYSTOLIC91=SBP14

Source variable(s): SBP8, SBP11, SBP14

### 4.4 DIASTOLIC91 (V9 Mean Diastolic BP of 2nd and 3rd Measurements)

Description: Mean of 2$^{nd}$ and 3$^{rd}$ diastolic blood pressure measurements, consistent with V1 through V5

Type: Numeric

Algorithm: DIASTOLIC91=mean of SBP9 and SBP12
If DIASTOLIC91=missing then DIASTOLIC91=SBP15

Source variable(s): SBP9, SBP12, SBP15

### 4.5 PULSE91 (V9 Mean Pulse of 2nd and 3rd Measurements)

Description: Mean of 2nd and 3rd pulse measurements, consistent with V1 through V5

Type: Numeric

Algorithm: PULSE91=mean of SBP10 and SBP13
If PULSE91=missing then PULSE91=SBP16

Source variable(s): SBP10, SBP13, SBP16

## 5. ALCOHOL USE

### 5.1 DRNKR91 (V9 Drinker Status)

Format: 1=Current Drinker
2=Former Drinker
3=Never Drinker
4=Unknown

Type: Numeric

Algorithm: Use the 1st not-permanently missing occurrence of ALC form if more than one.
If ALC2 in ('Y', missing) and ALC3= 'Y' then DRNKR91=1;
Else if (ALC2= 'Y' and ALC3= 'N') then DRNKR91=2;
Else if (ALC2= 'N' and ALC3 in ('N', missing)) then DRNKR91=3;
Else if (ALC2=missing and ALC3= 'N') or (ALC2= 'Y' and ALC3=missing) then DRNKR91=4;
Else DRNKR91=missing.

Source variable(s): ALC2, ALC3

### 5.2 ETHANL91 (V9 Usual Ethanol Intake (g/wk))

Format: continuous variable

Type: Numeric

Algorithm: Use the 1st not-permanently missing occurrence of ALC form if more than one.
If (DRNKR91 in (2, 3) or ALC3='N') then ETHANL91=0;
Else if (DRNKR91 in (4, missing) and missing(ALC5a) and missing(ALC6a) and missing(ALC7a) ) then ETHANL91=missing;
Else ETHANL91=(ALC5a x 10.8) + (ALC6a x 13.2)+(ALC7a x 15.1).

Source variable(s): ALC3, ALC5a, ALC6a, ALC7a, DRNKR91

### 5.3 CURDRK91 (V9 Current Drinker)

Format: 0=No,
1=Yes,
.T=missing (keeping .T for historical purposes).

Type: Numeric

Algorithm: Use the first not-permanently missing occurrence of ALC form if more than one.
If (ALC2 in ('Y', missing) and ALC3='Y') then CURDRK91=1;
Else if ALC3='N' or (ALC2='N' and ALC3=missing) then CURDRK91=0;
Else CURDRK91=.T

Source variable(s): ALC2, ALC3

### 5.4 FORDRK91 (V9 Former Drinker)

Format: 0=No,
1=Yes,
.T=missing.

Type: Numeric

Algorithm: Use the first not-permanently missing occurrence of ALC form if more than one.
If (ALC2= 'Y' and ALC3= 'N') then FORDRK91=1;
Else if (ALC2 in ('Y', missing) and ALC3= 'Y') or (ALC2= 'N' and ALC3 in ('N', missing)) then FORDRK91=0;
Else FORDRK91=.T

Source variable(s): ALC2, ALC3

### 5.5 EVRDRK91 (V9 Ever Drinker)

Format: 0=No,
1=Yes,
.T=missing.

Type: Numeric

Algorithm: Use the first not-permanently missing occurrence of ALC form if more than one.
If ALC2= 'Y' or (ALC2=missing and ALC3= 'Y') then EVRDRK91=1;
Else if (ALC2= 'N' and ALC3 in ('N', missing)) then EVRDRK91=0;
Else EVRDRK91=.T

Source variable(s): ALC2, ALC3

## 6. DISEASE PREVALENCE

There are a number of variables in the Disease Prevalence section that utilize cohort surveillance information, through event year 2022, to derive disease prevalence. All derived variables using surveillance data are considered final when the event year is closed. Versions 3 - 5 for diabetes prevalence are not created in DERIVE91.

### Diabetes Prevalence Variables

Temporary variables were created for use in the algorithms of the following diabetes-derived variables. They are not found in the DERIVE91 dataset.

**Temporary MDDX_DIAB91**

Algorithm: MDDX_DIAB91=1 if a participant reported being diagnosed during ARIC Follow-Up interviews with diabetes prior to the visit 9 date. The composite dataset variables considered are AFUCOMP1_A, AFUCOMP7D_G, and AFUCOMP15_M.

**Temporary INCSELFREPDM91**

Algorithm: If any of the records for a single ID have a 'Y' value for either AFUcomp7d_G or AFUcomp15_M and
.z<afucomp1_A<="30NOV2022"d
then INCSELFREPDM91 = 1;

Else if AFUcomp7d_G, AFUcomp15_M are (N,'') or ('',N) respectively in all records for a single ID, where
.z<afucomp1_A<="30NOV2022"d
then INCSELFREPDM91 = 0;

Else INCSELFREPDM1 = .T

### 6.1 DIABTS96 (V9 Diabetes – Hemogoblin A1C, cutpoint 6.5%)

Description: Diabetes variable defined as present if hemoglobin A1C value>=6.5 % or using medication for diabetes or self-report diagnosis of diabetes.

Format: 0=No,
1=Yes,
.T=missing.

Type: Numeric

Algorithm: If (CHEM1>=6.5) or (MSR2 ne 'T' and MSR33c = 'Y') or MDDX_DIAB91=1 then DIABTS96=1;

Else if (.z< CHEM1<6.5) and MDDX_DIAB91 ne 1 and MSR33c ne 'Y' then DIABTS96 =0;

Else DIABTS96=.T ;

Source variable(s): CHEM1, MDDX_DIAB91, MSR2, MSR33c

### 6.2 DIABTS97 (V9 Diabetes (DM medications or DM reported on AFU))

Description: Diabetes variable defined as present if participant is using medication for diabetes or self-report diagnosis of diabetes.

Format: 0=No,
1=Yes,
.T=missing.

Type: Numeric

Algorithm: If (MSR2 ^= 'T' and MSR33c='Y') or INCSELFREPDM91=1 or DIABMDCODE91=1 then DIABTS97=1

Else if INCSELFREPDM91 ^= 1 and MSR33c ^= 'Y' and DIABMDCODE91 ^= 1 then DIABTS97=0

Else DIABTS97=.T

Source variable(s): MSR2, MSR33c, INCSELFREPDM91, DIABMDCODE91

## Myocardial Infarction (MI) Prevalence Variables

### 6.3 MDDXMI92 (V9 MD Diagnosed Myocardial Infarction)

Description: This variable is derived from ARIC Follow-Up questions that ask if the participant was told by a doctor whether they had a heart attack between visit 7 and visit 9.

Format: 0=No,
1=Yes,
.T=missing.

Type: Numeric

Algorithm: If (AFUCOMP7a_G = 'Y' OR AFUCOMP11a_M = 'Y') AND V7DATE71 < AFUCOMP1_A (AFU DATE) < V9DATE91 then MDDXMI92=1

Else if all values of (AFUCOMP7a_G, AFUCOMP11a_M) for records between visit 7 date and visit 9 date are one of the following combinations ('',U), ('',''), (U,'') then MDDXMI92=.T

Else MDDXMI92=0

Source variable(s): AFUCOMP7a_G, AFUCOMP11a_M, V7DATE71, V9DATE91, AFUCOMP1_A

### 6.4 HXOFMI92 (V9 History of Myocardial Infarction)

Description: This variable is derived from ARIC Follow-Up questions that ask if the participant was told by a doctor whether they had a heart attack as well as questions asking the participant if they'd been hospitalized for heart attack. The follow-up records from the ARIC Follow-Up composite dataset considered for this variable were collected before the end of V9 data collection (30NOV2022).

Format: 0=No,
1=Yes,
.T=missing.

Type: Numeric

Algorithm: If MDDXMI92=1 or (AFUcomp7_V1= 'Y') or (AFUcomp30_deceased_A='Y') or (AFUcomp30non_deceased_A='Y') then HXOFMI92=1

Else if over all records for a single ID the following value combinations are found for (MDDXMI92, AFUcomp7_V1, AFUcomp30_deceased_A, AFUcomp30non_deceased_A): (.T,'','',''), (.T,'','U',''), (.T,'','','U') then HXOFMI92=.T

Else HXOFMI92=0

Source variable(s): MDDXMI92, AFUCOMP7_V1, AFUCOMP30_deceased_A, AFUcomp30non_deceased_A

## Coronary Heart Disease (CHD) Prevalence Variables

### 6.5 PRVCHD91 (V9 Prevalent CHD before Visit 9)

Description: This variable is derived from the baseline status of CHD (PRVCHD05) and the closed event years of ARIC Cohort Surveillance data through 2022, where the events occurred prior to the participant's Visit 9.

Format: 0=No,
1=Yes,
.=missing.

Type: Numeric

Algorithm: If PRVCHD05=1 or (C7_IN_21SP=1 and .<C7_DATEISP< V9DATE91) or (C7_IN_21SP=1 and V9DATE91=. and C7_DATEISP<="30NOV2022"d) then PRVCHD91=1

Else if PRVCHD05=0 and ((C7_IN_21SP=0 or C7_DATEISP>= V9DATE91>.) or (V9DATE91=. And C7_DATEISP>"30NOV2022"d)) then PRVCHD91=0

Else PRVCHD91=.T

Source variable(s): PRVCHD05, C7_IN_21SP, C7_DATEISP, V9DATE91

### 6.6 PRVCHD93 (V9 Prevalent CHD by end of Visit 9)

Description: This variable is derived from the baseline status of CHD (PRVCHD05) and the closed event years of ARIC Cohort Surveillance data through 2022, where the events occurred prior to the end of V9 data collection (30NOV2022).

Format: 0=No,
1=Yes,
.T=missing.

Type: Numeric

Algorithm: If PRVCHD05=1 or (C7_IN_21SP=1 and .<C7_DATEISP<="30NOV2022"d) then PRVCHD93=1

Else if PRVCHD05=0 and (C7_IN_21SP=0 or C7_DATEISP>"30NOV2022"d) then PRVCHD93=0

Else PRVCHD93=.T

Source variable(s): PRVCHD05, C7_IN_21SP, C7_DATEISP

### 6.7 PRVCHD94 (V9 Prevalent CHD - unverified)

Description: This variable is derived from self-reported ARIC Follow-Up data including questions on doctor told participant about heart attack, coronary bypass, and coronary angioplasty on records collected up through the end of V9 data collection (30NOV2022).

Format: 0=No,
1=Yes,
.T=missing.

Type: Numeric

Algorithm: use AFUcomp: only for those records where .z < AFUcomp1_A <= "30NOV2022"d

If MDDXMI92=1 or (AFUcomp13a_G='Y') or (AFUcomp15a_G='Y') then PRVCHD94=1

Else if over all records for a single ID the following value combinations are found for (MDDXMI92, AFUcomp13a_G, AFUcomp15a_G):((.M or .), ('' or 'U'), ('' or 'U')) then PRVCHD94=.

Else PRVCHD94=0

Source variable(s): AFUcomp1_A, MDDXMI92, AFUcomp13a_G, AFUcomp15a_G

## Stroke Prevalence Variables

### 6.8 PRVSTR91 (V9 Prevalent Stroke by the end of Visit 9)

Description: This variable is derived from the baseline status of stroke (HOM10D) and the closed event years of ARIC Surveillance data on the cohort through 2022, where the events occurred prior to the end of V9 data collection (30NOV2022).

Format: 0=No,

1=Yes,
.T=missing.

Type: Numeric

Algorithm: If HOM10D=1 or (C7_IN21DP=1 and .<C7_ED21DP<=V9DATE91) or (C7_IN21DP=1 and V9DATE91=. and C7_ED21DP<="30NOV2022") then PRVSTR91=1

Else if HOM10D=0 and ((C7_IN21DP=0 or C7_ED21DP>V9DATE91>.) or (V9DATE91=. and C7_ED21DP> "30NOV2022")) then PRVSTR91=0

Else PRVSTR91=.T

Source variable(s): HOM10D, C7_IN21DP, C7_ED21DP, V9DATE91

### 6.9 PRVSTR92 (V9 Prevalent Stroke-unverified)

Description: This variable is derived from self-reported ARIC Follow-Up data including questions that doctor told participant about stroke or TIA or participant hospitalized for stroke on records collected up through the end of V9 data collection (30NOV2022).

Format: 0=No,
1=Yes,
.T=missing.

Type: Numeric

Algorithm: If any record within a single ID has a "Y" value is found in either AFUcomp29_A or AFUcomp8b_K then PRVSTR92=1
Else if over all records within a single ID the following value combinations are found for (AFUcomp29_A, AFUcomp8b_K): ('','), ('',U), (U,'') then PRVSTR92=.T
Else PRVSTR92=0

Source variable(s): AFUcomp29_A, AFUcomp8b_K

## Hypertension Prevalence Variables

### 6.10 HYPERT94 (V9 HTN definition 4 (DIASTOLIC91 GE 90 or HTN med))

Description: Hypertension definition defined as diastolic blood pressure (mean of 2nd and 3rd measures) >=90 or medication is being taken for high blood pressure.

Format: 0=No,
1=Yes,
.T=missing.

Type: Numeric

Algorithm: If (DIASTOLIC91>= 90) or (MSR2 ne 'T' and MSR33d='Y') then HYPERT94=1;

Else if (0<DIASTOLIC91<90) and (MSR33d='N' or (MSR33d=missing and MSR2='T')) then HYPERT94=0;

Else HYPERT94=.T ;

Source variable(s): DIASTOLIC91, MSR2, MSR33d

### 6.11 HYPERT95 (V9 HTN definition 5 (SYSTOLIC91 GE 140 or DIASTOLIC91 GE 90 or HTN medication))

Description: Hypertension is defined as systolic blood pressure (mean of 2nd and 3rd measures) >=140 or diastolic blood pressure (mean of 2nd and 3rd measures) >=90 or medication is being taken for high blood pressure.

Format: 0=No,
1=Yes,
.T=missing.

Type: Numeric

Algorithm: If (DIASTOLIC91>= 90) or (SYSTOLIC91>= 140) or (MSR2 ne 'T' and MSR33d='Y') then HYPERT95=1

Else if (0<DIASTOLIC91<90) and (0<SYSTOLIC91<140) and (MSR33d='N' or (MSR33d=missing and MSR2='T')) then HYPERT95=0

Else HYPERT95=.T

Source variable(s): DIASTOLIC91, SYSTOLIC91, MSR2, MSR33d

### 6.12 HYPERT96 (V9 HTN definition 6 (SYSTOLIC91 GE 160 or DIASTOLIC91 GE 95 or HTN medication)

Description: Hypertension is defined as systolic blood pressure (mean of 2nd and 3rd measures) >=160 or diastolic blood pressure (mean of 2nd and 3rd measures) >=95 or medication is being taken for high blood pressure.

Format: 0=No,
1=Yes,
.T=missing.

Type: Numeric

Algorithm: If (DIASTOLIC91>= 95) or (SYSTOLIC91>=160) or (MSR2 ne 'T' and MSR33d='Y') then HYPERT96=1

Else if (0<DIASTOLIC91<95) and (0<SYSTOLIC91<160) and (MSR33d='N' or (MSR33d=missing and MSR2='T')) then HYPERT96=0

Else HYPERT96=.T

Source variable(s): DIASTOLIC91, SYSTOLIC91, MSR2, MSR33d

### 6.13 HYPERT97 (V9 HTN definition 7 (SYSTOLIC91 GE 150 or DIASTOLIC91 GE 90 or HTN medication)

Description: Hypertension is defined as systolic blood pressure (mean of 2nd and 3rd measures) >=150 or diastolic blood pressure (mean of 2nd and 3rd measures) >=90 or medication is being taken for high blood pressure.

Format: 0=No,
1=Yes,
.T=missing.

Type: Numeric

Algorithm: If (DIASTOLIC91>=90) or (SYSTOLIC91>=150) or (MSR2 ne 'T' and MSR33d='Y') then HYPERT97 = 1

Else if (0<DIASTOLIC91<90) and (0<SYSTOLIC91<150) and (MSR33d='N' or (MSR33d=missing and MSR2='T')) then HYPERT97=0

Else HYPERT97=.T

Source variable(s): DIASTOLIC91, SYSTOLIC91, MSR2, MSR33d

## Heart Failure (HF) Prevalence Variables

The heart failure (HF) variables included in this derived variable dictionary define HF prevalence at Visit 9. In general, the Definite HF variable will be most specific for prevalent HF, while the Definite or Possible HF variable will be the most sensitive. The appropriate variable for any given analysis will therefore depend on the balance between the sensitivity and specificity necessary for that analysis.

When conducting analyses of incident HF, it is important to note that the incident HF variable is available in the ARIC Cohort Surveillance dataset. The variable C7_INCHF21 uses V1 as the baseline and considers an event as a hospitalization with an ICD-9 discharge diagnosis code of 428.x in any position or corresponding ICD-10 code.

More detailed analysis recommendations from the ARIC Heart Failure Committee are available in the ARIC Analysis Manual (Manual 30).

---

The following definitions are temporary variables for use in the algorithms of the heart failure prevalence derived variables and are not included in DERIVE91.

**Temporary HospHF_2005onward**

Algorithm: For HFC21OCC1 records where
01JAN2005<=HFEVTDATE<V9DATE91
HospHF_2005onward=1, if CHFDIAG in ('A' 'B' 'C') for any record for SubjectID
HospHF_2005onward=0, otherwise

**Temporary OnsetDate**

Algorithm: Using data from the most recent PHF and PHFphi datasets and PHFA1104
If PHF2c is not missing, OnsetDate=PHF2c
Else if PHF7 is not missing, OnsetDate=PHF7
Else if PHFA2c is not missing, OnsetDate=PHFA2c

Else if PHFA7 is not missing, OnsetDate=PHFA7
Else if PHF0a is not missing, OnsetDate=PHF0a
Else if ENTRY_DA is not missing, OnsetDate=ENRTY_DA

**Temporary HF_byPHF**

Algorithm: Where missing<OnsetDate<V9DATE91
HF_byPHF=1, if phf1='Y' or phfa1='Y'
HF_byPHF=0, otherwise
Keep last record for each SubjectID only after sorted by SubjectID, HF_byPHF, and OnsetDate

**Temporary HF_first428_pre2005**

Algorithm: Using C21CELB1 records where missing<CELB04<01JAN2005
HF_first428_pre2005=1, if ICD code in first position (CELB10a) is 428.x for any record per SubjectID
HF_first428_pre2005=0, otherwise

**Temporary HF_non1st428_pre2005**

Algorithm: Using C21CELB1 records where missing<CELB04<01JAN2005
HF_non1st428_pre2005=1, if ICD code is 428.x anywhere but the first position (CELB10b-CELB10z) for any record for the SubjectID
HF_non1st428_pre2005=0, otherwise

**Temporary srHF/srHFdt/Form**

Algorithm: For [v3]phxa04, srHF=phxa8j, srHFdt=phxa63, Form=NULL
For [v4]phxb04, srHF=phxb6b, srHFdt=phxb21, Form=NULL
For afu&y1104new (y=g, h, i, j, k), srHF=afu&y7b, srHFdt=afu&y1, Form=NULL

For afu&y1104new (y=l, m),
srHF='Y' if afu&y7b, afu&y8, afu&y9, or afu&y10 is 'Y';
else srHF='N' if afu&y7, afu&y8, afu&y9, or afu&y10 is 'N'.
srHFdt=adu&y1 if not missing; else srHFdt=ENTRY_DA
Form=NULL

For afu_&mrt, &mrt=most recent data pull,
srHF='Y' if afu30 or afu31 is 'Y';
else srHF='N' if afu30 or afu31 is 'N'.
srHFdt=AFU0a if not missing;
else srHFdt=DATESTAMPINITIAL_AFU
Form=NULL

For saf_&mrt, &mrt=most recent data pull,
srHF='Y' if saf4, saf4a, saf5, or saf5a is 'Y';
else srHF='N' if saf4, saf4a, saf5, or saf5a is 'N'
srHFdt=SAF0a if not missing;

else srHFdt=DATESTAMPINITIAL_SAF
Form=NULL

For phfa1104, srHF=phfa1, Form=form
srHFdt is first completed date from the list phfa7, phfa9, ENTRY_DA

For phf_&mrt, &mrt=most recent data pull,
srHF=phf1, Form=form
srHFdt is first completed date from the list phf7, phf9, DATESTAMPINITIAL_PHF

Keep only records where srHF is 'Y' or 'N'

**Temporary selfreportHF**

Algorithm: From previous temporary dataset, sort by SubjectID, srHFDT, srHF
Keep records where missing<srHFdt<=v9date91 OR Form='PHF'

Start with selfreport=0 for first record per SubjectID then
Selfreport=1, if srHF='Y';
Selfreport=0, if srHF='Y' is followed by a record with srHF='N' and form='PHF'

Last record is kept for each SubjectID

**Temporary srHFmed/srHFmedDt**

Algorithm:
```
%macro selfreportHFmeds(dsn, medDt, med)
If &med='Y" then do;
srHFmed=1;
srHFmedDt=&medDt;
%mend;
%selfreportHFmeds(v1.msra, MSRA11, msra08d);
%selfreportHFmeds(v2.msrb, MSRB27, msrb24d);
%selfreportHFmeds(v3.msrc04, MSRC29, msrc24e);
%selfreportHFmeds(v4.msrd04, MSRD31, msrd24e);
%selfreportHFmeds(v5.msr, MSRF0A, msrf33h);
%selfreportHFmeds(v6.msr, MSRF0A, msrf33h);
%selfreportHFmeds(v7.msr, MSR0A, MSR33h);
%selfreportHFmeds(v9.msr, MSR0A, MSR33h);
%selfreportHFmeds(postv4.aful1104new, AFUL1, AFUL46D);
%selfreportHFmeds(postv4.afum1104new, AFUM1, AFUM46D);
%selfreportHFmeds(afu.afu_180425, AFU0a, AFU65D);
```

**Temporary selfreportHFmeds**

Algorithm: Keep records where missing<srHFmedDT<V9DATE91
selfreportHFmeds=1, if srHFmed=1 for any record;

selfreportHFmeds=0, otherwise

### 6.14 PREVDEFHF91 (V9 Prevalent Definite Heart Failure for Closed Event Years)

Description: A participant is defined to have prevalent definite heart failure if at least one of the following is true: (1) Prior hospitalization (01/01/2005 onward but before V9 visit) classified as Definite (A), Probable (B), or Chronic (C) HF; OR (2) Physician Heart Failure (PHF) Survey with HF onset date prior to V9 (from those with self-reported HF) in which the physician answers YES to "has this patient ever had HF or CM?"; OR (3) Hospitalization with an ICD code 428.x in first position (before 01/01/2005)

Format:
0=No,
1=Yes,
.=missing.

Algorithm: 1. Prior hospitalization (01/01/2005 onward but before V9 visit) classified as Definite (A), Probable (B), or Chronic (C) HF

2. Physician Heart Failure (PHF) Survey with HF onset date prior to V9 (from those with self-reported HF) in which the physician answers YES to "has this patient ever had HF or CM?"

3. Hospitalization with an ICD code 428.x in first position (before 01/01/2005)

Using Temporary Variables:
If V9DATE91 is not missing AND (HospHF_2005onward=1 OR HF_byPHF=1 OR HF_first428_pre2005=1) then PREVDEFHF91=1
Else PREVDEFHF91=0

Source variables: V9DATE91, HospHF_2005onward, HF_byPHF, HF_first428_pre2005

### 6.15 PREVDEFPOSSHF91 (V9 Prevalent Definite OR Possible Heart Failure for Closed Event Years)

Description: A participant is defined to have prevalent definite OR possible heart failure if at least one of the following is true: (1) Prior hospitalization (01/01/2005 onward but before V9 visit) classified as Definite (A), Probable (B), or Chronic (C) HF; OR (2) Physician Heart Failure Survey with HF onset date prior to V9 (from those with self-reported HF) in which the physician answers YES to "has this patient ever had HF or CM?"; OR (3) Hospitalization with an ICD code 428.x in

first position (before 01/01/2005); OR (4) Hospitalization with an ICD code 428.x in any position other than the first position (any time before 01/01/2005); OR (5) Self-report of HF at AFU prior to V9 or at visits 3-4*, not refuted by the physicians health survey (temporal association will need to be considered); OR (6) Self-report of treatment for HF from any study visit or AFU prior to V9. *Note that self-reported HF was only asked at V3 and V4.

Format:

0=No,
1=Yes,
.=missing.

Algorithm:

1. Prior hospitalization (01/01/2005 onward but before V9 visit) classified as Definite (A), Probable (B), or Chronic (C) HF

2. Physician Heart Failure Survey with HF onset date prior to V9 (from those with self-reported HF) in which the physician answers YES to "has this patient ever had HF or CM?"

3. Hospitalization with an ICD code 428.x in first position (before 01/01/2005)

4. Hospitalization with an ICD code 428.x in any position other than the first position (any time before 01/01/2005)

5. Self-report of HF at AFU prior to V9 or at visits 3-4*, not refuted by the physicians health survey (temporal association will need to be considered)

6. Self-report of treatment for HF from any study visit or AFU prior to V9

*Self-reported HF at V1, V2, V5, V6, V7, V9 not asked

Using Temporary Variables:
If V9DATE91 is not missing AND (PREVDEFHF91=1 OR HF_non1st428_pre2005=1 OR selfreportHF=1 OR selfreportHFmeds=1) then PREVDEFPOSSHF91=1
Else PREVDEFPOSSHF91=0

Source variables:

V9DATE91, PREVDEFHF91, HF_non1st428_pre2005, selfreportHF, selfreportHFmeds

## 7. MEDICATION USE

ARIC uses Generic Product Identifier (GPI) codes to identify use of selected medications based on data collected in the Medication Survey (MSR) form at the ARIC clinic visits. The derived medication variable definitions were reviewed following Visit 7 and resulted in new variable versions for most of the medications of interest. Analyses should only use the most recent versions of the derived medication variables.

Recall for Visits 5 - 7, the old variable versions were retained in the updated derived datasets for reproducibility purposes, however, those variables include "DO NOT USE" in the label.

In Visits 9 - 11, medication variables will include version number in the label. Medication variables with no adjustments will have “ver1” added to the label, and medication variables with new definitions will have “ver2” added to the label. The exception is for CHOLMDCODE: ver1 and ver2 are no longer created, use the updated ver3 and ver4.

For example, CHOLMDCODE51, CHOLMDCODE61, and CHOLMDCODE71 should no longer be used for analyses as “DO NOT USE” is present in the variable label. The updated versions, CHOLMDCODE53, CHOLMDCODE63, and CHOLMDCODE73 should be analyzed. The comparable variable in Visit 9 is CHOLMDCODE93, which now includes “ver3” in the variable label.

In another example, HYPTMD51, HYPTMD61, and HYPTMD71 may be used for analyses as the definitions have not changed. The comparable variable at Visit 9 is HYPTMD91, which includes “ver1” in the variable label as the definition is unchanged.

---

The following definitions are temporary variables, defined here as reference for the following medication-related, derived variables. They are not found on the DERIVE91 dataset.

**Temporary ALLMISS**

Algorithm: Do over MSR*a_gpi where * = 5 to 29:
If any MSR*a_gpi >'' then ALLMISS=0, else ALLMISS=1

**Temporary MSR*a2_sub10:**

Algorithm: Do over MSR*a_gpi where* = 5 to 29:
MSR*a2_sub10=substr(MSR*a_gpi,1,10)

**Temporary MSR*a2_sub8:**

Algorithm: Do over MSR*a_gpi where * = 5 to 29:
MSR*a2_sub8=substr(MSR*a_gpi,1,8)

**Temporary MSR*a2_sub6:**

Algorithm: Do over MSR*a_gpi where * = 5 to 29:
MSR*a2_sub6=substr(MSR*a_gpi,1,6)

**Temporary MSR*a2_sub4:**
Algorithm: Do over MSR*a_gpi where * = 5 to 29:
MSR*a2_sub4=substr(MSR*a_gpi,1,4)

**Temporary MSR*a2_sub2:**
Algorithm: Do over MSR*a_gpi where * = 5 to 29:
MSR*a2_sub2=substr(MSR*a_gpi,1,2)

**Temporary FOUNDCHOL93**
Algorithm: Do over MSR*a2_sub# where * = 5 to 29
If MSR*a2_sub6 in (279930, 390000-399999, 409925, 771030, 781044) or MSR*a2_sub8 in (40100025, 81250070, 96428049, 96782839) or MSR*a2_sub10 in (9646564700, 9652504244) then FOUNDCHOL93=1
Else FOUNDCHOL93=0.

**Temporary FOUNDCHOL94**
Algorithm: Do over MSR*a2_sub# where * = 5 to 29
If MSR*a2_sub4 in (2599) or (MSR*a2_sub6 in (121030, 121045, 121085, 121099, 363000, 369920, 376000, 379900, 379910, 590700 - 594000) and MSR*a2_sub10 NOT in (1210990250)) or MSR*a2_sub8 in (12105005, 24991002, 24993002, 24995002, 33100010, 33100025, 33100040, 33100045, 33100050, 33200020, 33200021, 33200022, 33200025, 33200030, 37200010) or MSR*a2_sub10 in (2400001500, 2400001600, 2400001700, 2400003000, 2400003500, 2400003504, 2400005500) then FOUNDCHOL94=1, else FOUNDCHOL94=0.

**Temporary FOUNDHYPT92**
Algorithm: Do over MSR*a2_sub# where * = 5 to 29
If (MSR*a2_sub6 in (330000-339999 or 340000-349999 or 360000-369999 or 370000-379999) and (MSR*a2_sub8 NOT in (37400010) and MSR*a2_sub6 NOT in (379920))) or MSR*a2_sub4 in (4099) or (MSR33d = 'Y') then FOUNDHYPT92=1,
Else FOUNDHYPT92=0.

**Temporary FOUNDSTAT92**
Algorithm: Do over MSR*a2_sub# where * = 5 to 29
If (MSR*a2_sub4 in (3940) and MSR*a2_sub8 NOT in (39409908)) or MSR*a2_sub6 in (279930, 399940, 409925) or MSR*a2_sub8 in (96428049, 96785839) then FOUNDSTAT92=1,
Else FOUNDSTAT92=0.

**Temporary FOUNDACOAG91**
Algorithm: Do over MSR*a2_sub2 where * = 5 to 29

If MSR*a2_sub2 equal 83 then FOUNDACOAG91=1
Else FOUNDACOAG91=0.

**Temporary FOUNDASP92**

Algorithm: Do over MSR*a2_sub# where * = 5 to 29
if MSR*a2_sub4 IN (6410) or MSR*a2_sub10 in (4399100232, 4399590415, 4399590419, 6030990225, 6499000220, 6499000221, 6499000225, 6499000320, 6499000321, 6499000340, 6499000450, 6499000460, 6499100222, 6499100330, 6599000222, 6599100430, 6599130310,7599000210, 7599000310, 7599000320, 8515001000, 8515990220) then FOUNDASP92=1,
Else FOUNDASP92=0.

**Temporary FOUNDANTIANX92**

Algorithm: Do over MSR*a2_sub2 where * = 5 to 29
If MSR*a2_sub2 in (57) then FOUNDANTIANX92=1, else FOUNDANTIANX92=0.

**Temporary FOUNDANTIPSYCH92**

Algorithm: Do over MSR*a2_sub# where * = 5 to 29
If MSR*a2_sub2 in (59) or MSR*a2_sub10 in (6200003000) then FOUNDANTIPSYCH92=1,
Else FOUNDANTIPSYCH92=0.

**Temporary FOUNDHYPNOT92**

Algorithm: Do over MSR*a2_sub# where * = 5 to 29
If MSR*a2_sub2 in (59) or MSR*a2_sub6 in (439935, 439940, 439959, 439966, 439968, 439975, 439988, 600000-609999) or MSR*a2_sub8 in (41100010-41100030, 41200030, 41200040, 41400020, 41500020, 41992002, 43992002, 50200030) or MSR*a2_sub10 in (4399300220, 4399300221, 4399300224, 4399300230, 4399300234, 4399300248, 4399300250, 4399300254, 4399300255, 4399300258, 4399300270, 4399300292, 4399300293, 4399300295, 4399300296, 4399520231-4399520234, 4399520236, 4399530310, 4399530311, 4399530313, 4399530314, 4399530317, 4399530319, 4399530320, 4399530327, 4399530354, 4399530357, 4399530390, 4399570210, 4399570220, 4399570230, 4399570243, 4399580306, 4399580308, 4399580312, 4399580315, 4399580330, 4399580332, 4399580346, 4399580348, 4399580349, 4399580350, 4399580354, 4399580361, 4399580362, 4399580363, 4399580364, 4399580367, 4399580368, 4399580376, 4399580377, 4399800426, 4399800439, 4399800470, 4399890315, 4399890325, 4399890332, 4399890335,

5030990210, 6499000280, 6599300220) then
FOUNDHYPNOT92=1,
Else FOUNDHYPNOT92=0.

**Temporary FOUNDANTICONV92**

Algorithm: Do over MSR*a2_sub# where * = 5 to 29
If MSR*a2_sub2 in (72) or MSR*a2_sub8 in (49109904, 59400015, 60100060) or MSR*a2_sub10 in (9672561675) then FOUNDANTICONV92=1, else FOUNDANTICONV92=0.

**Temporary FOUNDANTIDEM92**

Algorithm: Do over MSR*a2_sub# where * = 5 to 29
If MSR*a2_sub4 in (6205) or MSR*a2_sub8 in (62000001) then FOUNDANTIDEM92=1,
Else FOUNDANTIDEMT92=0.

**Temporary FOUNDCNSALT92**

Algorithm: Do over MSR*a2_sub# where * = 5 to 29
If FOUNDHYPNOT92=1 or FOUNDANTICONV92=1 then FOUNDCNSALT92=1;
Else if MSR*a2_sub2 in (57 - 60) or MSR*a2_sub4 in (6140, 6299) or MSR*a2_sub6 in (613540, 932000) or MSR*a2_sub8 in (62206040, 96426631) or MSR*a2_sub10 in (9652646380) then FOUNDCNSALT92=1,
Else FOUNDCNSALT92=0.

**Temporary FOUNDDIAB91**

Algorithm: Do over MSR*a2_sub2 where * = 5 to 29
If MSR*a2_sub2=27 then FOUNDDIAB91=1, else FOUNDDIAB91=0.

**Temporary FOUNDBETA92**

Algorithm: Do over MSR*a2_sub# where * = 5 to 29
If (MSR*a2_sub2in (33)) or MSR*a2_sub6 in (369920,369927, 369988) then FOUNDBETA92=1,
Else FOUNDBETA92=0.

**Temporary FOUNDANGINH92**

Algorithm: Do over MSR*a2_sub# where * = 5 to 29
If MSR*a2_sub6 in (361000, 369918, 369985) or MSR*a2_sub8 in (96645857) then FOUNDANGINH92=1;
Else FOUNDANGINH92=0.

**Temporary FOUNDANGIANT92**

Algorithm: Do over MSR*a2_sub6 where * = 5 to 29
If MSR*a2_sub6 in (361500, 369930-369945, 369965) then FOUNDANGIANT92=1;
Else FOUNDANGIANT92=0.

**Temporary FOUNDALDANT92**

Algorithm: Do over MSR*a2_sub# where * = 5 to 29
If MSR*a2_sub6 IN (362500) or MSR*a2_sub10 IN (3750002000, 3799000220) then FOUNDALDANT92=1;
Else FOUNDALDANT92=0.

**Temporary FOUNDLOOPDIU92**

Algorithm: Do over MSR*a2_sub# where * = 5 to 29
If MSR*a2_sub6 in (372000) or MSR*a2_sub8 in (96508007) then FOUNDLOOPDIU92=1;
Else FOUNDLOOPDIU92=0.

**Temporary FOUNDDIG92**

Algorithm: Do over MSR*a2_sub8 where * = 5 to 29
If MSR*a2_sub8 in (31200010, 96485821) then FOUNDDIG92=1;
Else FOUNDDIG92=0.

## 7.1 CHOLMDCODE93 (V9 Cholesterol Lowering Medication in past 4 weeks – Using Medi-Span GPI Code ver3)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDCHOL93=1 then CHOLMDCODE93=1;

Else if FOUNDCHOL93=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then CHOLMDCODE93=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then CHOLMDCODE93=.T;

Else CHOLMDCODE93=.;

Source variable(s): FOUNDCHOL92, ALLMISS, MSR2

## 7.2 CHOLMDCODE94 (V9 Medications Which Secondarily Affect Cholesterol in past 4 Weeks – Using Medi-Span GPI Code ver4)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDCHOL94=1 then CHOLMDCODE94=1;

Else if FOUNDCHOL94=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then CHOLMDCODE94=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then CHOLMDCODE94=.T;

Else CHOLMDCODE94=.;

Source variable(s): FOUNDCHOL94, ALLMISS, MSR2

## 7.3 HYPTMD91 (V9 Hypertension Medications in past 4 Weeks: Self-reported ver1)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If (MSR2 NE 'T') and (MSR33D='Y') then HYPTMD91=1;

Else if (MSR2='T' and MSR33D='') OR MSR33D='N' then HYPTMD91 = 0;

Else If ((MSR2 NE 'T') and (MSR33D='U' or MSR33D = '')) or ((MSR2='T') and (MSR33D = 'Y' or MSR33D='U')) then HYPTMD91=.T;

Source variable(s): MSR2, MSR33D

## 7.4 HYPTMDCODE92 (V9 Hypertension Lowering Medication in past 4 Weeks – Using Medi-Span GPI Code ver2)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDHYPT92=1 OR (MSR2 NE ‘T’) and (MSR33D='Y')) then HYPTMDCODE92=1;

Else if FOUNDHYPT92=0 AND ((ALLMISS=1 AND MSR2=’T’) OR (ALLMISS=0) OR (MSR2='T' and MSR33D='') OR (MSR33D='N’)) then HYPTMDCODE92=0;

Else if ALLMISS=1 AND (MSR2=’F’ OR MSR2=’ ‘) then HYPTMDCODE92=.T;

Else HYPTMDCODE92=.;

Source variable(s): FOUNDHYPT92, MSR2, MSR33D, ALLMISS

### 7.5 STATINCODE92 (V9 Statin Use in past 4 Weeks – Using Medi-Span GPI Code ver2)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDSTAT92=1 then STATINCODE92=1;

Else if FOUNDSTAT92=0 AND ((ALLMISS=1 AND MSR2=’T’) OR (ALLMISS=0)) then STATINCODE92=0;

Else if ALLMISS=1 AND (MSR2=’F’ OR MSR2=’’) then STATINCODE92=.T;

Else STATINCODE92=.;

Source variable(s): FOUNDSTAT92, ALLMISS, MSR2

### 7.6 ANTICOAGCODE91 (V9 Anticoagulant Use in past 4 Weeks – Using Medi-Span GPI Code ver1)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDACOAG91=1 then ANTICOAGCODE91=1;

Else if FOUNDACOAG91=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then ANTICOAGCODE91=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then ANTICOAGCODE91=.T;

Else ANTICOAGCODE91=.;

Source variable(s): FOUNDACOAG91, ALLMISS, MSR2

### 7.7 ASPIRINCODE92 (V9 Aspirin Use in past 4 Weeks – Using Medi-Span GPI Code ver2)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDASP92=1 then ASPIRINCODE92=1;

Else if FOUNDASP92=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then ASPIRINCODE92=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then ASPIRINCODE92=.T;

Else ASPIRINCODE92=.;

Source variable(s): FOUNDASP92, ALLMISS, MSR2

### 7.8 ANTIANXMDCODE92 (V9 Antianxiety Medication in past 4 Weeks – Using Medi-Span GPI Code ver2)

Format: 0=No
1=Yes

.T=Missing

Type: Numeric

Algorithm: If FOUNDANTIANX92=1 then ANTIANXMDCODE92=1;

Else if FOUNDANTIANX92=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then ANTIANXMDCODE92=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then ANTIANXMDCODE92=.T;

Else ANTIANXMDCODE92=.;

Source variable(s): FOUNDANTIANX92, ALLMISS, MSR2

### 7.9 ANTIPSYCHMDCODE92 (V9 Antipsychotic Medication in past 4 Weeks – Using Medi-Span GPI Code ver2)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDANTIPSYCH92=1 then ANTIPSYCHMDCODE92=1;

Else if FOUNDANTIPSYCH92=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then ANTIPSYCHMDCODE92=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then ANTIPSYCHMDCODE92=.T;

Else ANTIPSYCHMDCODE92=.;

Source variable(s): FOUNDANTIPSYCH92, ALLMISS, MSR2

### 7.10 HYPNOTMDCODE92 (V9 Hypnotic/Sedative Medication in past 4 Weeks – Using Medi-Span GPI Code ver2)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDHYPNOT92=1 then HYPNOTMDCODE92=1;

Else if FOUNDHYPNOT92=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then HYPNOTMDCODE92=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then HYPNOTMDCODE92=.T;

Else HYPNOTMDCODE92=.;

Source variable(s): FOUNDHYPNOT92, ALLMISS, MSR2

### 7.11 ANTICONVMDCODE92 (V9 Anticonvulsant Medication in past 4 weeks – Using Medi-Span GPI Code ver2)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDANTICONV92=1 then ANTICONVMDCODE92=1;

Else if FOUNDANTICONV92=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then ANTICONVMDCODE92=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then ANTICONVMDCODE92=.T;

Else ANTICONVMDCODE92=.;

Source variable(s): FOUNDANTICONV92, ALLMISS, MSR2

### 7.12 ANTIDEMMDCODE92 (V9 Antidementia/Nootropic Medication in past 4 weeks – Using Medi-Span GPI Code ver2)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDANTIDEM92=1 then ANTIDEMMDCODE92=1;

Else if FOUNDANTIDEM92=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then ANTIDEMMDCODE92=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then ANTIDEMMDCODE92=.T;

Else ANTIDEMMDCODE92=.;

Source variable(s): FOUNDANTIDEM92, ALLMISS, MSR2

### 7.13 CNSALTMDCODE92 (V9 CNS Altering Medication in past 4 weeks – Using Medi-Span GPI Code ver2)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDCNSALT92=1 then CNSALTMDCODE92=1;

Else if FOUNDCNSALT92=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then CNSALTMDCODE92=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then CNSALTMDCODE92=.T;

Else CNSALTMDCODE92=.;

Source variable(s): FOUNDCNSALT92, ALLMISS, MSR2

### 7.14 DIABMDCODE91 (V9 Diabetic Medications in past 4 weeks – Using Medi-Span GPI Code ver1)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDDIAB91=1 then DIABMDCODE91=1;

Else if FOUNDDIAB91=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then DIABMDCODE91=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then DIABMDCODE91=.T;

Else DIABMDCODE91=.;

Source variable(s): FOUNDDIAB91, ALLMISS, MSR2

### 7.15 BETAMDCODE92 (V9 Beta-Blocker in past 4 weeks – Using Medi-Span GPI Code ver2)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDBETA92=1 then BETAMDCODE92=1;

Else if FOUNDBETA92=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then BETAMDCODE92=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then BETAMDCODE92=.T;

Else BETAMDCODE92=.;

Source variable(s): FOUNDBETA92, ALLMISS, MSR2

### 7.16 ANGINHMDCODE92 (V9 Angiotensin converting enzyme inhibitor in past 4 weeks – Using Medi-Span GPI Code ver2)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDANGINH92=1 then ANGINHMDCODE92=1;

Else if FOUNDANGINH92=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then ANGINHMDCODE92=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then ANGINHMDCODE92=.T;

Else ANGINHMDCODE92=.;

Source variable(s): FOUNDANGINH92, ALLMISS, MSR2

### 7.17 ANGIANTMDCODE92 (V9 Angiotensin II receptor antagonists in past 4 weeks – Using Medi-Span GPI Code ver2))

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDANGIANT92=1 then ANGIANTMDCODE92=1;

Else if FOUNDANGIANT92=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then ANGIANTMDCODE92=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then ANGIANTMDCODE92=.T;

Else ANGIANTMDCODE92=.;

Source variable(s): FOUNDANGIANT92, ALLMISS, MSR2

### 7.18 ALDANTMDCODE92 (V9 Aldosterone Antagonist in past 4 weeks – Using Medi-Span GPI Code ver2)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDALDANT92=1 then ALDANTMDCODE92=1;

Else if FOUNDALDANT92=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then ALDANTMDCODE92=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then ALDANTMDCODE92=.T;

Else ALDANTMDCODE92=.;

Source variable(s): FOUNDALDANT92, ALLMISS, MSR2

### 7.19 LOOPDIUMDCODE92 (V9 Loop Diuretic in past 4 weeks – Using Medi-Span GPI Code ver2)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDLOOPDIU92=1 then LOOPDIUMDCODE92=1;

Else if FOUNDLOOPDIU92=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then LOOPDIUMDCODE92=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then LOOPDIUMDCODE92=.T;

Else LOOPDIUMDCODE92=.;

Source variable(s): FOUNDLOOPDIU92, ALLMISS, MSR2

### 7.20 DIGMDCODE92 (V9 Digoxin in past 4 weeks – Using Medi-Span GPI Code ver2)

Format: 0=No
1=Yes
.T=Missing

Type: Numeric

Algorithm: If FOUNDDIG92=1 then DIGMDCODE92=1;

Else if FOUNDDIG92=0 AND ((ALLMISS=1 AND MSR2='T') OR (ALLMISS=0)) then DIGMDCODE92=0;

Else if ALLMISS=1 AND (MSR2='F' OR MSR2='') then DIGMDCODE92=.T;

Else DIGMDCODE92=.;

Source variable(s): FOUNDDIG92, ALLMISS, MSR2

## 8. PHYSICAL ACTIVITY

### 8.1 SPRT_I91 (V9 Sport during Leisure Time)

Description: This index is a composite score of sport during leisure time.

Type: Continuous variable between 1 and 5.

Algorithm: Numeric

- Create temporary variables I2_1, I2_2, I2_3, I2_4 (sport score corresponding to each of the 4 recorded sports or exercises reported by the participant)

| | | PAC variables being used in temporary variable | | | |
|---|---|---|---|---|---|
| | Sport Count | Activity | Time | Proportion | Remark |
| I2_1 | 1 | PAC2 | PAC3 | PAC4 | |
| I2_2 | 2 | PAC6 | PAC7 | PAC8 | 0 if PAC5='N' |
| I2_3 | 3 | PAC10 | PAC11 | PAC12 | 0 if PAC9='N' |
| I2_4 | 4 | PAC14 | PAC15 | PAC16 | 0 if PAC13='N' |

- An intensity level is assigned to each of the activity codes

| Intensity | Activity Codes (PAC2, PAC6, PAC10, PAC14) |
|---|---|
| Light | 1,31,43,55,60,61,73,76,79,97,121,125,136,142,154,169,178,208, 229,244,286,304,322,328,403,412,415,418,421,448,499 |
| Moderate | 2,3,4,7,10,16,25,28,37,40,49,52,70,82,94,100,118,124,130,139, 145,146,148,160,163,175,181,187,190,193,199,205,211,217,232, 235,238,247,249,350,333,498,259,262,265,271,289,292,295,301, 310,313,319,325,331,340,352,355,358,364,376,385,388,391,397, 400,404,406,409,427,430,436,451 |
| Heavy | 13,19,22,46,58,67,85,88,91,109,112,115,127,133,151,157,166,172, 184,196,202,214,220,223,226,241,250,253,268,274,277,280,283, 298,316,334,337,343,346,349,361,367,370,373,379,382,394,424, 433,437,439,442 |

- Calculate sport score for each activity using the following formula, intensity(**X**)*hours_weeks(**Y**)*months_year(**Z**) where X, Y, and Z are assigned values according to the following:

| Intensity | **X** |
|---|---|
| Light | 0.76 |
| Moderate | 1.26 |
| Heavy | 1.76 |

| Hours (per week) | Variable value | **Y** |
|---|---|---|
| Less than one hour | A | .5 |
| At least 1 but not quite 2 | B | 1.5 |
| At least 2 but not quite 3 | C | 2.5 |
| At least 3 but not quite 4 | D | 3.5 |
| 4 or more | E | 4.5 |

| Months (per year) | Variable value | **Z** |
|---|---|---|
| Less than 1 | A | 0.04 |
| At least 1 but not quite 4 | B | 0.17 |
| At least 4 but not quite 7 | C | 0.42 |
| At least 7 but not quite 10 | D | 0.67 |
| 10 or more | E | 0.92 |

**I2 Summary sports score**

- Calculate I_2=I2_1+ I2_2 +I2_3+I2_4: sum of four simple sports scores.
  Note:
  If PAC5='N' then I2_2=0
  If PAC9='N' then I2_3=0
  If PAC13='N' then I2_4=0

  This sum score is then recoded to a score of 1 to 5 based on the following criteria:

  If PAC1='N' then I2=1 else
  If 0<=I_2<0.01 then I2=1
  If 0.01<=I_2<4 then I2=2
  If 4<=I_2<8 then I2=3
  If 8<=I_2<12 then I2=4
  If 12<=I_2 then I2=5

**I3 Leisure sport exercise activity versus peers, recoded**

- PAC19 was recoded as follows:

| Format | PAC19 value | Recoded value for algorithm |
|---|---|---|
| Much less | A | 1 |
| Less | B | 2 |
| The same | C | 3 |
| More | D | 4 |
| Much more | E | 5 |

**I4 Sweat during leisure time, recoded**

- PAC20 was recoded as follows:

| Format | PAC20 value | Recoded value for algorithm |
|---|---|---|
| Never | A | 1 |
| Seldom | B | 2 |
| Sometimes | C | 3 |
| Often | D | 4 |
| Very often | E | 5 |

**I5 Sports/exercise during leisure activity, recoded**

- PAC18 was recoded as follows:

| Format | PAC18 value | Recoded value for algorithm |
|---|---|---|
| Never | A | 1 |
| Seldom | B | 2 |
| Sometimes | C | 3 |
| Often | D | 4 |
| Very often | E | 5 |

SPRT_I91= (I2+I3+I4+I5)/4
If I2 or I3 or I4 or I5 is missing, then SPRT_I91=.T

## 8.2 LISR_I92 (V9 Physical Activity during Leisure Time Excluding Sport)

Description: This index is a composite score of leisure time activity that includes frequency of TV viewing, frequency of walking, and frequency of bicycling. The question about the number of minutes walked or bicycled per day to and from work or shopping was not included in the battery so the variable is not exactly defined as was in previous visits.

Type: Continuous variable between 1 and 5.

Algorithm: Numeric

**I6 Leisure time television watching, recoded**

- PAC21 was recoded as follows:

| Format | PAC21 value | Recoded value for algorithm |
|---|---|---|
| Never | A | 5 |
| Seldom | B | 4 |
| Sometimes | C | 3 |
| Often | D | 2 |
| Very often | E | 1 |

**I7 Leisure time walking frequency, recoded**

- PAC22 was recoded as follows:

| Format | PAC22 | Recoded value for algorithm |
|---|---|---|
| Never | A | 1 |
| Seldom | B | 2 |
| Sometimes | C | 3 |
| Often | D | 4 |
| Very often | E | 5 |

**I8 Leisure time cycling frequency, recoded**

- PAC23 was recoded as follows:

| Format | PAC23 value | Recoded value for algorithm |
|---|---|---|
| Never | A | 1 |
| Seldom | B | 2 |
| Sometimes | C | 3 |
| Often | D | 4 |
| Very often | E | 5 |

LISR_I92= (I6+I7+I8+ ((I7+I8)/2))/4
If I6 or I7 or I8 is missing, then LISR_I92=.T

## 9. PHYSICAL FUNCTION

The physical function variables mainly use the data collected on the ARIC PFX form. The short physical performance battery (SPPB) is a group of measures that combines the results of the gait speed, chair stand and balance tests (Guralnik et al., 2000). The scores range from 0 (worst performance) to 12 (best performance).

### 9.1 SPPBCS91 (V9 Physical Function Chair Stand)

Format: integer variable (0-4 possible points)

Type: Numeric

Algorithm: If (PFX1 in (2,3,4)) OR (PFX2 in (2,3)) OR (PFX2b_DER>=60) then SPPBCS91=0;
Else if (16.70<= PFX2b_DER <60) then SPPBCS91=1;
Else if (13.70<= PFX2b_DER <16.70) then SPPBCS91=2;
Else if (11.20<= PFX2b_DER <13.70) then SPPBCS91=3;
Else if (. <PFX2b_DER <11.20) then SPPBCS91=4;
Else SPPBCS91=.;

Source variable(s): PFX1, PFX2, PFX2b_DER

### 9.2 SPPBST91 (V9 Physical Function Semi Tandem Stand)

Format: integer variable

Type: Numeric

Algorithm: If (PFX3 in (2,3,4)) then SPPBST91=0;
Else if (PFX3=5) then SPPBST91=1;
Else SPPBST91=.;

Source variable(s): PFX3

### 9.3 SPPBSBS91 (V9 Physical Function Side-by-Side Stand)

Format: integer variable

Type: Numeric

Algorithm: If (PFX4 in (2,3,4)) then SPPBSBS91=0;
Else if (PFX3=5 OR PFX4=5) then SPPBSBS91=1;
Else SPPBSBS91=.;

Source variable(s): PFX3, PFX4

### 9.4 SPPBTS91 (V9 Physical Function Tandem Stand)

Format: integer variable

Type: Numeric

Algorithm: If max(PFX5, PFX6) in (2,3,4) then SPPBTS91=0;
else if 3 <= max(PFX5a_DER, PFX6a_DER) <10 then SPPBTS91=1;
else if max(PFX5, PFX6) in (5) then SPPBTS91=2;
else SPPBTS91=.;

Source variable(s): PFX5, PFX5a_DER, PFX6, PFX6a_DER

### 9.5 SPPBBAL91 (V9 Physical Function Summary Balance Score)

Format: integer variable

Type: Numeric

Algorithm: SPPBBAL91=sum(of SPPBST91,SPPBSBS91,SPPBTS91);

Source variable(s): SPPBSBS91, SPPBST91, SPPBTS91

### 9.6 SPPB4M91 (V9 Physical Function 4 Meter Walk Score)

Format: integer variable (0-4)

Type: Numeric

Algorithm: If max(PFX7,PFX8)=2 then SPPB4M91=0;
Else if (WALK4M91> 8.70) SPPB4M91=1;
Else if (6.21 <= WALK4M91<= 8.70) then SPPB4M91=2;
Else if (4.82 <= WALK4M91< 6.21) then SPPB4M91=3;
Else if (.z < WALK4M91< 4.82) then SPPB4M91=4;
Else SPPB4M91=.T;

Source variable(s): PFX7, PFX8, WALK4M91

### 9.7 SPPB91 (V9 Short Physical Performance Summary Battery Score)

Format: integer variable

Type: Numeric

Algorithm: The SPPB total score is only calculated when all three of its subcomponent tasks are not missing. If any of SPPBCS91, SPPBBAL91, or SPPB4M91 are missing, then SPPB91 is set to missing.
If SPPBCS91>NULL and SPPBBAL91>NULL and SPPB4M91>NULL then SPPB91=sum(of SPPBCS91,SPPBBAL91,SPPB4M91);
Else SPPB91 = NULL;

Source variable(s): SPPB4M91, SPPBBAL91, SPPBCS91

### 9.8 WALK4M91 (V9 Physical Function 4 Meter Walk, Fastest Time of 2 Trials)

Format: continuous variable (f4.2)

Type: Numeric

Algorithm: The better of (1) PFX7a and PFX8a if walking aid is used OR the better of (2) PFX7b and PFX8b if no walking aid is used. There are some instances where a walking aid is used in one trial and not in the other trial. The variable will be the fastest time walked regardless of using the aid or not. All participants at the clinic visits should have a timed walk. Any who did not do the test due to "not attempted/unable" will have a missing value for this variable. If only one trial completed, use the results from that trial.

If .<max(pfx7,pfx8)<=2 then WALK4M91=.T;
Else WALK4M91 = min(PFX7a_der, PFX7b_der, PFX8a_der, PFX8b_der);

Source variable(s): PFX7, PFX7a_der, PFX7b_der, PFX8, PFX8a_der, PFX8b_der

### 9.9 WALKAID91 (V9 Physical Function 4 Meter Walk: Used Walking Aid)

Format: 0=No,
1=Yes,
.T=missing.

Type: Numeric

Algorithm: This indicator variable will be set to 1 when a patient used a walking aid during the 4 meter walk. The walk is performed twice and in the instance where a patient uses an aid in one trial, but not in the other, this variable will be set according to the presence or absence of a walking aid that goes with the fastest time of the 2 trials.

If WALK4M91>.T and (WALK4M91=PFX7a_der or WALK4M91=PFX8a_der) then WALKAID91 = 1;
Else if WALK4M91>.T then WALKAID91 = 0;
Else WALKAID91=.T;

Source variable(s): PFX7a_der, PFX8a_der, WALK4M91

### 9.10 GRIPBEST91 (V9 Physical Function Grip, Best of 2 Trials)

Format: continuous variable

Type: Numeric

Algorithm: Max of PFX11b and PFX11c. Only participants who respond "Both" to PFX10a should be excluded (i.e., surgery on both hands). Less than 2% are missing grip strength as of May 2012, likely to have little impact on inferences. Standard approaches to missing data such as sensitivity analyses with multiple imputations can be employed if inappropriate to ignore missingness.

If PFX10 NE missing and PFX10a NE 'B' then GRIPBEST91=max(PFX11b, PFX11c);

Source variable(s): PFX10, PFX10a, PFX11b, PFX11c

### 9.11 GRIPMEAN91 (V9 Physical Function Grip, Mean of 2 Trials or Result for 1 Trial)

Format: continuous variable

Type: Numeric

Algorithm: Mean of PFX11b and PFX11c. Only participants who respond "Both" to PFX10a should be excluded (i.e., surgery on both hands). The variable will be missing if less than 2 trials are completed. Less

than 2% are missing grip strength as of May 2012, likely to have little impact on inferences. Standard approaches to missing data such as sensitivity analyses with multiple imputations can be employed if inappropriate to ignore missingness.

If PFX10 NE missing and PFX10a NE 'B' then GRIPMEAN91=mean(PFX11b,PFX11c);

Source variable(s): PFX10, PFX10a, PFX11b, PFX11c

### 9.12 V7V9WTDELTAP91 (% of V7 weight change from V7 to V9 (neg val means loss) – used in frailty weight loss component)

Description: The change in weight from visit 7 to visit 9 is calculated and presented as the % of visit 7 weight, rounded to 0.1.

Format: continuous variable (0-99, %)

Type: Numeric

Algorithm: ((v9ant4-v7ant4)/v7ant4)*100, where v7ant4 = ant4 from visit 7 and v9ant4=ant4 from visit 9

Source variable(s): ant4

### 9.13 WALKSPEED15FT91 (V9 Time in seconds used to walk 15ft – used in frailty slowness component)

Description: The physical function form collects data on times to walk 4 meters. The WALKSPEED15FT91 variable uses the 4m walk rate to calculate the time in seconds needed to walk 15ft.

Format: numeric continuous variable

Type: Numeric

Algorithm: WALKSPEED15FT91=MIN((15/3.28084)* (pfx7a_der/4), (15/3.28084)*(pfx7b_der/4), (15/3.28084)*(pfx8a_der/4), (15/3.28084)*(pfx8b_der/4))

Source variable(s): PFX7A_DER, PFX7B_DER, PFX8A_DER, PFX8B_DER

### 9.14 TR1WALK4MSP91 (V9 4M Walking Speed for Trial 1 (m/sec))

Format: continuous variable

Type: Numeric

Algorithm: TR1WALK4MSP91=4/(max(PFX7a_der, PFX7b_der)

Source variable(s): PFX7a, PFX7b

### 9.15 TR2WALK4MSP91 (V9 4M Walking Speed for Trial 2 (m/sec))

Format: continuous variable

Type: Numeric

Algorithm: TR2WALK4MSP91=4/(max(PFX8a_der, PFX8b_der)

Source variable(s): PFX8a, PFX8b

### 9.16 AVGWALK4MSP91 (V9 Average 4M Walking Speed (m/sec))

Format: continuous variable

Type: Numeric

Algorithm: AVGWALK4MSP91=mean(TR1WALK4MSP91, TR2WALK4MSP91)

Source variable(s): TR1WALK4MSP91, TR2WALK4MSP91

### 9.17 MINWALK4MSP91 (V9 Fastest 4M Walking Speed for Both Trials (m/sec))

Format: continuous variable

Type: Numeric

Algorithm: MINWALK4MSP91=min(TR1WALK4MSP91, TR2WALK4MSP91) ;

Source variable(s): TR1WALK4MSP91, TR2WALK4MSP91

### 9.18 UNABLETOWALK91 (V9 Indicator Variable Noting PPT No Attempt to Walk 4M, Not Able)

Format: 0=No,
1=Yes,
.=Missing.

Type: Numeric

Algorithm: If PFX7 and PFX8 are both NULL then UNABLETOWALK91=NULL
Else if PFX7=2 and PFX8 = missing or 2 then
UNABLETOWALK91=1
Else UNABLETOWALK91=0

Source variable(s): PFX7, PFX8

### 9.19 UNINTEND_WTLOSS91 (V9 Unintentional weight loss)

Format: 0=No,
1=Yes,
.=missing.

Type: Numeric

Algorithm: If TMW1 is not missing then do;
If TMW3=0 then UNINTEND_WTLOSS91=1;
Else UNINTEND_WTLOSS91=0;

Source variable(s): TMW1, TMW3

### 9.20 TMW_TOTFT91 (V9 Total Feet Walked in 2-Minute Walk)

Format: continuous variable

Type: Numeric

Algorithm: Calculate if TMW_PRES =1
TMW_TOTFT91=sum((tmw8*50),tmw9);

Source variable(s): TMW_PRES, TMW8, TMW9

### 9.21 TMW_COMPSPEEDFTPERSEC91 (V9 TMW Speed for Completers (ft/sec))

Format: continuous variable

Type: Numeric

Algorithm: Calculate if TMW_PRES =1 and tmw11=5
TMW_COMPSPEEDFTPERSEC91=tmw_totft91/120;

Source variable(s): TMW11, TMW_TOTFT91

### 9.22 TMW_COMPSPEEDMPERSEC91 (V9 TMW Speed for Completers (m/sec))

Format: continuous variable

Type: Numeric

Algorithm: Calculate if TMW_PRES =1 and tmw11=5
TMW_COMPSPEEDMPERSEC91=tmw_totft91*0.3048/120

Source variable(s): TMW11, TMW_TOTFT91

### 9.23 TMW_NONCOMPSECWALK91 (V9 TMW Seconds Walking for Noncompleters)

Format: continuous variable

Type: Numeric

Algorithm: Calculate if TMW_PRES =1 and tmw11=4
TMW_NONCOMPSECWALK91=sum((tmw12a*60),tmw12b)

Source variable(s): TMW11, TMW12a, TMW12b

### 9.24 TMW_ NONCOMPSPEEDFTPERSEC91 (V9 TMW Speed for Noncompleters (ft/sec))

Format: continuous variable

Type: Numeric

Algorithm: Calculate if tmw_pres=1 and tmw11=4 and
tmw_noncompsecwalk91>0

TMW_NONCOMPSPEEDFTPERSEC91=tmw_totft91/tmw_noncompsecwalk91

Source variable(s): tmw11, tmw_noncompsecwalk91, tmw_totft91

### 9.25 TMW_ NONCOMPSPEEDMPERSEC91 (V9 TMW Speed for Noncompleters (m/sec))

Format: continuous variable

Type: Numeric

Algorithm: Calculate if tmw_pres=1 and tmw11=4 and tmw_noncompsecwalk91>0
TMW_NONCOMPSPEEDMPERSEC91=tmw_totft91*0.3048/tmw_noncompsecwalk91

Source variable(s): tmw11, tmw_noncompsecwalk91, tmw_totft91

### 9.26 EXHAUST91 (V9 Responded 2 or 3 on CES3 or CES11 (CESD) – frailty exhaustion component)

Description: Second component in considering frailty of PPT based on exhaustion from depression.

Format: 0=No
1=Yes
.=Missing

Type: Numeric

Algorithm: If (CES3=2 OR CES3=3) OR (CES11=2 OR CES11=3) then EXHAUST91=1;

Else if CES3=. AND CES11=. then EXHAUST91=.;

Else EXHAUST91=0;

Source variable(s): CES2, CES3, CES11

### 9.27 LOWENERGYCOMP91 (V9 Lowest quintile of SPRT_i91 - frailty low physical activity component)

Description: The third component in considering frailty of PPT based on low physical activity during leisure time.

Format: 0=No
1=Yes
.=Missing

Type: Numeric

Algorithm: LOWENERGYCOMP91=1, if (.<SPRT_I91<1.70 AND GENDER='F') OR (SPRT_I91<2.00 AND GENDER='M')

LOWENERGYCOMP91=0, if (.<SPRT_I91>=1.70 AND GENDER='F') OR (SPRT_I91>=2.00 AND GENDER='M')

LOWENERGYCOMP91=NULL, if gender=NULL or SPRT_I91=NULL

Source variable(s): SPRT_I91, GENDER

### 9.28 WTLOSSCOMPA91 (V9 Lost >5% weight or Low BMI (<18.5) – frailty weight loss component for 91a definition)

Description: First component in considering frailty of PPT based on weight loss greater than 5% or low BMI.

Format: 0=No
1=Yes
.=Missing

Type: Numeric

Algorithm: WTLOSSCOMP91=1, if (.<V7V9WTDELTAP91 <=-5) OR (.<BMI91<18.5)

WTLOSSCOMP91=0, if (V7V9WTDELTAP91 >-5) AND (BMI91>=18.5)

WTLOSSCOMP91=NULL, if V7V9WTDELTAP91 and BMI91 are NULL

Source variable(s): V7V9WTDELTAP91 , BMI91

### 9.29 WTLOSSCOMPB91 (V9 Lost >10% weight or Low BMI (<18.5) – frailty weight loss component for 91b definition)

Description: First component in considering frailty of PPT based on more restrictive weight loss greater than 10% or low BMI.

Format: 0=No
1=Yes
.=Missing

Type: Numeric

Algorithm: WTLOSSCOMP91=1, if (.<V7V9WTDELTAP91 <=-10) OR (.<BMI91 <18.5)

WTLOSSCOMP91=0, if (V7V9WTDELTAP91 >-10) AND (BMI91 >=18.5)

WTLOSSCOMP91=NULL, if V7V9WTDELTAP91 and BMI91 are NULL

Source variable(s): V7V9WTDELTAP91 and, BMI91

### 9.30 WALKSPEEDCOMP91 (V9 Slowest 20% time to walk 15ft – frailty slowness component)

Description: Fourth component in considering frailty of PPT based on slowness by walking.

Format: 0=No
1=Yes
.=Missing

Type: Numeric

Algorithm: WALKSPEEDCOMP91=1:

Males:
IF (gender = 'M' AND V6IN129 <=173 AND Walkspeed15ft91>= 7)
OR
IF (gender = 'M' AND V6IN129 > 173 AND Walkspeed15ft91>= 6)
OR

Females:

IF (gender = 'F' AND V6IN129 <=159 AND Walkspeed15ft91>= 7)
OR
IF (gender = 'F' AND V6IN129 > 159 AND Walkspeed15ft91>= 6)

WALKSPEEDCOMP91=NULL:
IF gender=NULL or V6IN129=NULL or Walkspeed15ft91= NULL
ELSE WALKSPEEDCOMP91=0

Source variable(s): gender, V6IN129 (participant height last measured at V5), Walkspeed15ft91

### 9.31 GRIPSTRENGTHCOMP91 (V9 Lowest 20% grip strength – frailty weakness component)

Description: Fifth component in considering frailty of PPT based on grip strength and BMI.

Format: 0=No
1=Yes
.=Missing

Type: Numeric

Algorithm: GRIPSTRENGTHCOMP91=1:

Males:
IF gender = 'M' AND BMI91 <= 24 AND .<MAX(pfx11b, pfx11c) <= 29 OR
IF gender = 'M' AND 24 < BMI91 <= 26 AND .<MAX(pfx11b, pfx11c) <= 30 OR
IF gender = 'M' AND 26 < BMI91 <= 28 AND .<MAX(pfx11b, pfx11c) <= 30 OR
IF gender = 'M' AND BMI91 > 28 AND .<MAX(pfx11b, pfx11c) <= 32

Females:
IF gender = 'F' AND BMI91 <= 23 AND .<MAX(pfx11b, pfx11c) <= 17 OR
IF gender = 'F' AND 23 < BMI91 <= 26 AND .<MAX(pfx11b, pfx11c) <= 17.3 OR
IF gender = 'F' AND 26 < BMI91 <= 29 AND .<MAX(pfx11b, pfx11c) <= 18 OR
IF gender = 'F' AND BMI91 > 29 AND .<MAX(pfx11b, pfx11c) <= 21

GRIPSTRENGTHCOMP91=NULL:

IF gender=NULL or BMI91 =NULL or (pfx11b and pfx11c)=NULL

GRIPSTRENGTHCOMP91=0:
Otherwise

Source variable(s): gender, BMI91, pfx11b, pfx11c

### 9.32 FRAILTY91a (V9 ARIC Physical Function WG Frailty Definition (a))

Description: There are 5 components that are considered in deriving the frailty variable. Indicators for these components are defined below. If 3 or more of the components are present, then FRAILTY91a=1. If 1 or 2 of the components are present, then FRAILTY91a=2. If none of the components are present and there is data for each component, then FRAILTY91a=3.

Format: 1=Frail
2=Prefrail
3=Robust
.=Missing

Type: Numeric

Algorithm:

1. Weight loss:
   If (.<V7V9WTDELTAP91 <=-5) OR (.<BMI91<18.5)

2. Exhaustion:
   If EXHAUST91=1

3. Low Energy:
   If (.<SPRT_I91<1.70 AND GENDER='F') OR (.<SPRT_I91<2.00 AND GENDER='M')

4. Walking Speed:
   IF (gender = 'M' AND V6IN129 <=173 AND Walkspeed15ft91>= 7)
   OR
   IF (gender = 'M' AND V6IN129 > 173 AND Walkspeed15ft91>= 6)
   OR
   IF (gender = 'F' AND V6IN129 <=159 AND Walkspeed15ft91>= 7)
   OR
   IF (gender = 'F' AND V6IN129 > 159 AND Walkspeed15ft91>= 6)

5. Grip Strength

IF gender = 'M' AND BMI91 <= 24 AND .<MAX(pfx11b, pfx11c) <= 29 OR
IF gender = 'M' AND 24 < BMI91 <= 26 AND .<MAX(pfx11b, pfx11c) <= 30 OR
IF gender = 'M' AND 26 < BMI91 <= 28 AND .<MAX(pfx11b, pfx11c) <= 30 OR
IF gender = 'M' AND BMI91 > 28 AND .<MAX(pfx11b, pfx11c) <= 32 OR
IF gender = 'F' AND BMI91 <= 23 AND .<MAX(pfx11b, pfx11c) <= 17 OR
IF gender = 'F' AND 23 < BMI91 <= 26 AND .<MAX(pfx11b, pfx11c) <= 17.3 OR
IF gender = 'F' AND 26 < BMI91 <= 29 AND .<MAX(pfx11b, pfx11c) <= 18 OR
IF gender = 'F' AND BMI91 > 29 AND .<MAX(pfx11b, pfx11c) <= 21

Source variable(s): V7V9WTDELTAP91, BMI91, EXHAUST91, SPRT_I91, GENDER, V6IN129 (participant height last measured at V5), Walkspeed15ft91, PFX11B, PFX11C

### 9.33 FRAILTY91b (V9 ARIC Physical Function WG Frailty Definition (b))

Description: There are 5 components that are considered in deriving the frailty variable. Indicators for these components are defined below. If 3 or more of the components are present, then FRAILTY91b=1. If 1 or 2 of the components are present, then FRAILTY91b=2. If none of the components are present and there is data for each component, then FRAILTY91b=3.

Format:
1=Frail
2=Prefrail
3=Robust
.=Missing

Type: Numeric

Algorithm:

1. Weight loss:
   If (.<V7V9WTDELTAP91 <=-10) OR (.<BMI91<18.5)

2. Exhaustion:
   If EXHAUST91=1

3. Low Energy:

If (.<SPRT_I91<1.70 AND GENDER='F') OR (.<SPRT_I91<2.00 AND GENDER='M')

4. Walking Speed:
   IF (gender = 'M' AND V6IN129 <=173 AND Walkspeed15ft91>= 7) OR
   IF (gender = 'M' AND V6IN129 > 173 AND Walkspeed15ft91>= 6) OR
   IF (gender = 'F' AND V6IN129 <=159 AND Walkspeed15ft91>= 7) OR
   IF (gender = 'F' AND V6IN129 > 159 AND Walkspeed15ft91>= 6)

5. Grip Strength
   IF gender = 'M' AND BMI91 <= 24 AND .<MAX(pfx11b, pfx11c) <= 29 OR
   IF gender = 'M' AND 24 < BMI91 <= 26 AND .<MAX(pfx11b, pfx11c) <= 30 OR
   IF gender = 'M' AND 26 < BMI91 <= 28 AND .<MAX(pfx11b, pfx11c) <= 30 OR
   IF gender = 'M' AND BMI91 > 28 AND .<MAX(pfx11b, pfx11c) <= 32 OR
   IF gender = 'F' AND BMI91 <= 23 AND .<MAX(pfx11b, pfx11c) <= 17 OR
   IF gender = 'F' AND 23 < BMI91 <= 26 AND .<MAX(pfx11b, pfx11c) <= 17.3 OR
   IF gender = 'F' AND 26 < BMI91 <= 29 AND .<MAX(pfx11b, pfx11c) <= 18 OR
   IF gender = 'F' AND BMI91 > 29 AND .<MAX(pfx11b, pfx11c) <= 21

Source variable(s): V7V9WTDELTAP91 , BMI91, EXHAUST91, SPRT_I91, GENDER, V6IN129 (participant height last measured at V5), Walkspeed15ft91, PFX11B, PFX11C

### 9.34 FRAILTY93 (V9 ARIC Physical Function WG Frailty Definition – weight loss dropped)

Description: There are 4 components considered in deriving the frailty variable. Indicators for these components are defined below. If 3 or more of the components are present, then frailty93=1. If 1 or 2 of the components are present, then frailty93=2. If none of the components are present (=0) AND there is data for each component (meaning no component has a missing value), then frailty93=3.

Format: 1=Frail

2=Prefrail
3=Robust
.=Missing

Type: Numeric

Algorithm:

1. Exhaustion:
   If EXHAUST91=1

2. Low Energy:
   If (.<SPRT_I91<1.70 AND GENDER='F') OR (.<SPRT_I91<2.00 AND GENDER='M')

3. Walking Speed:
   IF (gender = 'M' AND V6IN129 <=173 AND Walkspeed15ft91>= 7)
   OR
   IF (gender = 'M' AND V6IN129 > 173 AND Walkspeed15ft91>= 6)
   OR
   IF (gender = 'F' AND V6IN129 <=159 AND Walkspeed15ft91>= 7)
   OR
   IF (gender = 'F' AND V6IN129 > 159 AND Walkspeed15ft91>= 6)

4. Grip Strength
   IF gender = 'M' AND BMI91 <= 24 AND .<MAX(pfx11b, pfx11c) <= 29 OR
   IF gender = 'M' AND 24 < BMI91 <= 26 AND .<MAX(pfx11b, pfx11c) <= 30 OR
   IF gender = 'M' AND 26 < BMI91 <= 28 AND .<MAX(pfx11b, pfx11c) <= 30 OR
   IF gender = 'M' AND BMI91 > 28 AND .<MAX(pfx11b, pfx11c) <= 32 OR
   IF gender = 'F' AND BMI91 <= 23 AND .<MAX(pfx11b, pfx11c) <= 17 OR
   IF gender = 'F' AND 23 < BMI91 <= 26 AND .<MAX(pfx11b, pfx11c) <= 17.3 OR
   IF gender = 'F' AND 26 < BMI91 <= 29 AND .<MAX(pfx11b, pfx11c) <= 18 OR
   IF gender = 'F' AND BMI91 > 29 AND .<MAX(pfx11b, pfx11c) <= 21

Source variable(s): BMI91, EXHAUST91, SPRT_I91, GENDER, V6IN129 (participant height last measured at V5), Walkspeed15ft91, PFX11B, PFX11C

## 9.35 FRAILTY94 (V9 ARIC Physical Function WG Frailty Definition (unintentional weight loss))

Description: There are 5 components considered in deriving the frailty variable. Indicators for these components are defined below. If 3 or more of the components are present, then frailty94=1. If 1 or 2 of the components are present, then frailty94=2. If none of the components are present AND there is data for each component, then frailty94=3.

Format:
1=Frail
2=Prefrail
3=Robust
.=Missing

Type: Numeric

Algorithm:

1. Unintentional weight loss
   If UNINTEND_WTLOSS91=1

2. Exhaustion:
   If EXHAUST91=1

3. Low Energy:
   If (.<SPRT_I91<1.70 AND GENDER='F') OR (.<SPRT_I91<2.00 AND GENDER='M')

4. Walking Speed:
   IF (gender = 'M' AND V6IN129 <=173 AND Walkspeed15ft91>= 7) OR
   IF (gender = 'M' AND V6IN129 > 173 AND Walkspeed15ft91>= 6) OR
   IF (gender = 'F' AND V6IN129 <=159 AND Walkspeed15ft91>= 7) OR
   IF (gender = 'F' AND V6IN129 > 159 AND Walkspeed15ft91>= 6)

5. Grip Strength
   IF gender = 'M' AND BMI91 <= 24 AND .<MAX(pfx11b, pfx11c) <= 29 OR
   IF gender = 'M' AND 24 < BMI91 <= 26 AND .<MAX(pfx11b, pfx11c) <= 30 OR
   IF gender = 'M' AND 26 < BMI91 <= 28 AND .<MAX(pfx11b, pfx11c) <= 30 OR
   IF gender = 'M' AND BMI91 > 28 AND .<MAX(pfx11b, pfx11c) <= 32 OR

IF gender = 'F' AND BMI91 <= 23 AND .<MAX(pfx11b, pfx11c) <= 17 OR
IF gender = 'F' AND 23 < BMI91 <= 26 AND .<MAX(pfx11b, pfx11c) <= 17.3 OR
IF gender = 'F' AND 26 < BMI91 <= 29 AND .<MAX(pfx11b, pfx11c) <= 18 OR
IF gender = 'F' AND BMI91 > 29 AND .<MAX(pfx11b, pfx11c) <= 21

Source variable(s): UNINTEND_WTLOSS91, BMI91, EXHAUST91, SPRT_I91, GENDER, V6IN129 (participant height last measured at V5), Walkspeed15ft91, PFX11B, PFX11C

### 9.36 GAITSPEED91 (V9 Gait Speed (m/sec))

Type: Numeric

Algorithm: If WALK4M91>0 then GAITSPEED91=4/WALK4M91

Source variable(s): WALK4M91

# 10. NEUROCOGNITIVE STUDY

## 10.1 CESD91 (V9 CES-Depression Scale)

Description: Numeric variable for score on the CES-Depression scale.

Type: Numeric

Algorithm: If there are 10 non-missing items among CES1 through CES11, then CESD91=SUM of (CES1-CES11)
Otherwise, missing.

Source variable(s): CES1-CES11

## 10.2 FAQ91 (V9 Functional Activities Questionnaire)

Description: Numeric variable for score on the Functional Activities Questionnaire.

Type: Numeric

Algorithm: FAQ91=CDI25 + CDI26 + CDI31 + 2(CDI35) + CDI36 + CDI37 + CDI18 + CDI17 + CDI22

Source variable(s): CDI25, CDI26, CDI31, CDI35, CDI36, CDI37, CDI18, CDI17, CDI22

## 11. PHYSICAL VARIABLES AND INDICATORS

### 11.1 AGENATMENOPAUSEF (Age (years) at natural menopause)

Description: Numeric variable indicating age in years at natural menopause.

Type: Numeric

Algorithm: AGENATMENOPAUSEF=AGENATMENOPAUSEF [STATUS51]

Source variable(s): AGENATMENOPAUSEF (from STATUS51)

### 11.2 AGESRGMENOPAUSEF (Age (years) at surgical menopause)

Description: Numeric variable indicating age in years at surgical menopause.

Type: Numeric

Algorithm: AGESRGMENOPAUSEF=AGESRGMENOPAUSEF [STATUS51]

Source variable(s): AGESRGMENOPAUSEF (from STATUS51)