

icd10pcs — ICD-10-PCS procedure codes

[Description](#)
[Options](#)
[Also see](#)

[Quick start](#)
[Remarks and examples](#)

[Menu](#)
[Stored results](#)

[Syntax](#)
[Acknowledgments](#)

Description

`icd10pcs` is a suite of commands for working with ICD-10-PCS procedure codes from U.S. federal fiscal year 2016 to the present. To see the current version of the ICD-10-PCS procedure codes and any changes that have been applied, type `icd10pcs query`.

`icd10pcs check`, `icd10pcs clean`, and `icd10pcs generate` are data management commands. `icd10pcs check` verifies that a variable contains defined ICD-10-PCS procedure codes and provides a summary of any problems encountered. `icd10pcs clean` standardizes the format of the codes. `icd10pcs generate` can create a binary indicator variable for whether the code is in a specified set of codes, a variable containing a corresponding higher-level code, or a variable containing the description of the code.

`icd10pcs lookup` and `icd10pcs search` are interactive utilities. `icd10pcs lookup` displays descriptions of the codes specified on the command line. `icd10pcs search` looks for relevant ICD-10-PCS procedure codes from keywords given on the command line.

Quick start

Determine whether ICD-10-PCS procedure codes in `proc1` are invalid, and store reasons in `invalid`

```
icd10pcs check proc1, generate(invalid)
```

Standardize display of codes in `proc2` to add a period and left-align codes

```
icd10pcs clean proc2, replace
```

Check that the codes in `proc3` conform to ICD-10-PCS formatting rules, and if so, create `main` as the corresponding three-character category code

```
icd10pcs generate main = proc3, category check
```

Generate `descr4` as the current short description of procedure code in `proc4`

```
icd10pcs generate descr4 = proc4, description
```

Look up current descriptions for procedure codes 081.23J4 through 081.Y3Z3

```
icd10pcs lookup 081.23J4/081.Y3Z3
```

Look up codes where the description from FFY-2016 contains the word “foot”

```
icd10pcs search foot, version(2016)
```

Menu

Data > ICD codes > ICD-10-PCS

Syntax

Verify that variable contains defined codes

```
icd10pcs check varname [if] [in] [, checkopts]
```

Clean variable and verify format of codes

```
icd10pcs clean varname [if] [in], {generate(newvar)|replace} [cleanopts]
```

Generate new variable from existing variable

```
icd10pcs generate newvar = varname [if] [in], category [check]
```

```
icd10pcs generate newvar = varname [if] [in], description [genopts]
```

```
icd10pcs generate newvar = varname [if] [in], range(codelist) [check]
```

Display code descriptions

```
icd10pcs lookup codelist [, version(#)]
```

Search for codes from descriptions

```
icd10pcs search ["text"] [{"text"} ...] [, searchopts]
```

Display ICD-10-PCS version

```
icd10pcs query
```

codelist is one of the following:

<i>icd10code</i>	(the particular code)
<i>icd10code*</i>	(all codes starting with)
<i>icd10code/icd10code</i>	(the code range)

or any combination of the above, such as O41.E09P 2W3* BQ2L/BQ2LZZZ.

<i>checkopts</i>	Description
<u>fmtonly</u>	check only format of the codes
<u>summary</u>	frequency of each invalid or undefined code
<u>list</u>	list observations with invalid or undefined ICD-10-PCS codes
<u>generate</u> (<i>newvar</i>)	create new variable marking invalid codes
<u>version</u> (#)	fiscal year to check codes against; default is the current year

<i>cleanopts</i>	Description
* <u>generate</u> (<i>newvar</i>)	create new variable containing cleaned codes
* <u>replace</u>	replace existing codes with the cleaned codes
<u>check</u>	check that variable contains ICD-10-PCS codes before cleaning
<u>nodots</u>	format codes without a period

*Either `generate()` or `replace` is required.

<i>genopts</i>	Description
<u>addcode</u> (<i>begin</i> <i>end</i>)	add code to the beginning or end of the description
<u>nodots</u>	format codes without a period; must specify <code>addcode()</code>
<u>check</u>	check that variable contains ICD-10-PCS codes before generating new variable
<u>long</u>	use long description rather than short
<u>version</u> (#)	select description from fiscal year #; default is the current year

<i>searchopts</i>	Description
<u>or</u>	match any keyword
<u>matchcase</u>	match case of keywords
<u>version</u> (#)	search description from fiscal year #; default is all

`collect` is allowed with `icd10pcs check` and `icd10pcs clean`; see [U] [11.1.10 Prefix commands](#).

The `icd10pcs` suite of commands does not allow alias variables; see [D] [frunalias](#) for advice on how to get around this restriction.

Options

Options are presented under the following headings:

Options for icd10pcs check
Options for icd10pcs clean
Options for icd10pcs generate
Option for icd10pcs lookup
Options for icd10pcs search

Options for icd10pcs check

`fmtonly` tells `icd10pcs check` to verify that the codes fit the format of ICD-10-PCS procedure codes but not to check whether the codes are defined.

`summary` specifies that `icd10pcs check` should report the frequency of each invalid or undefined code that was found in the data. Codes are displayed in descending order by frequency. `summary` may not be combined with `list`.

`list` specifies that `icd10pcs check` list the observation number, the invalid or undefined ICD-10-PCS procedure code, and the reason the code is invalid or whether it is an undefined code. `list` may not be combined with `summary`.

`generate(newvar)` specifies that `icd10pcs check` create a new variable containing, for each observation, 0 if the observation contains a defined code. Otherwise, it contains a number from 1 to 11 if the code is invalid, 77 if the code is valid only for a previous version, 88 if the code is valid only for a later version, 99 if the code is undefined, or missing if the code is missing. The positive numbers indicate the kind of problem and correspond to the listing produced by `icd10pcs check`.

`version(#)` specifies the version of the codes that `icd10pcs check` should reference. `#` indicates the federal fiscal year for the codes. For example, use 2016 for federal fiscal year 2016 (FFY-2016), which is October 1, 2015 to September 30, 2016. `icd10pcs` supports all years after the United States officially adopted ICD-10-PCS. The appropriate value of `#` should be determined from the data source. The default is the current year.

Warning: The default value of `version()` will change over time so that the most recent codes are used. Using the default value rather than specifying a specific version may change results after a new version of the codes is introduced.

Options for `icd10pcs clean`

`generate(newvar)` and `replace` specify how the formatted values of `varname` are to be handled. You must specify either `generate()` or `replace`.

`generate()` specifies that the cleaned values be placed in the new variable specified in `newvar`.

`replace` specifies that the existing values of `varname` be replaced with the formatted values.

`check` specifies that `icd10pcs clean` should first check that `varname` contains codes that fit the format of ICD-10-PCS procedure codes. Specifying the `check` option will slow down `icd10pcs clean`.

`nodots` specifies that the period be removed in the final format.

Options for `icd10pcs generate`

`category`, `description`, and `range(codelist)` specify the contents of the new variable that `icd10pcs generate` is to create. You do not need to `icd10pcs clean varname` before using `icd10pcs generate`; it will accept any supported format or combination of formats.

`category` specifies to extract the three-character category code from the ICD-10-PCS procedure code. The resulting variable may be used with the other `icd10pcs` subcommands.

`description` creates `newvar` containing descriptions of the ICD-10-PCS procedure codes.

`range(codelist)` creates a new indicator variable equal to 1 when the ICD-10-PCS procedure code is in the range specified, equal to 0 when the ICD-10-PCS procedure code is not in the range, and equal to missing when `varname` is missing.

`addcode(begin|end)` specifies that the code should be included with the text describing the code. Specifying `addcode(begin)` will prepend the code to the text. Specifying `addcode(end)` will append the code to the text.

`nodots` specifies that the code that is added to the description should be formatted without a period. `nodots` may be specified only if `addcode()` is also specified.

`check` specifies that `icd10pcs generate` should first check that `varname` contains codes that fit the format of ICD-10-PCS procedure codes. Specifying the `check` option will slow down the `generate` subcommand.

`long` specifies that the long description of the code be used rather than the short (abbreviated) description.

`version(#)` specifies the version of the codes that `icd10pcs generate` should reference. `#` indicates the federal fiscal year for the codes. For example, use 2016 for federal fiscal year 2016 (FFY-2016), which is October 1, 2015 to September 30, 2016. `icd10pcs` supports all years after the United States officially adopted ICD-10-PCS. The appropriate value of `#` should be determined from the data source. The default is the current year.

Warning: The default value of `version()` will change over time so that the most recent codes are used. Using the default value rather than specifying a specific version may change results after a new version of the codes is introduced.

Option for `icd10pcs lookup`

`version(#)` specifies the version of the codes that `icd10pcs lookup` should reference. `#` indicates the federal fiscal year for the codes. For example, use 2016 for federal fiscal year 2016 (FFY-2016), which is October 1, 2015 to September 30, 2016. `icd10pcs` supports all years after the United States officially adopted ICD-10-PCS. The appropriate value of `#` should be determined from the data source. The default is the current year.

Warning: The default value of `version()` will change over time so that the most recent codes are used. Using the default value rather than specifying a specific version may change results after a new version of the codes is introduced.

Options for `icd10pcs search`

`or` specifies that ICD-10-PCS procedure codes be searched for descriptions that contain any word specified with `icd10pcs search`. The default is to list only descriptions that contain all the words specified.

`matchcase` specifies that `icd10pcs search` should match the case of the keywords given on the command line. The default is to perform a case-insensitive search.

`version(#)` specifies the version of the codes that `icd10pcs search` should reference. `#` indicates the federal fiscal year for the codes. For example, use 2016 for federal fiscal year 2016 (FFY-2016), which is October 1, 2015 to September 30, 2016. `icd10pcs` supports all years after the United States officially adopted ICD-10-PCS.

By default, descriptions for all versions are searched, meaning that codes that changed descriptions and that have descriptions in multiple versions that contain the search terms will be duplicated. To ensure a list of unique code values, specify the version number.

Remarks and examples

stata.com

Remarks are presented under the following headings:

[Introduction](#)
[Managing datasets with ICD-10-PCS codes](#)
[Interactive utilities](#)

If you have not yet read [Introduction to ICD coding](#) in [D] `icd`, please do so before using the `icd10pcs` commands.

Introduction

The general format of an ICD-10-PCS procedure code is a three-character category code followed by four alpha-numeric characters after an (implied) period. The full codes are always seven characters long and may be any combination of letters and numbers.

Some examples of ICD-10-PCS procedure codes are 081 (Eye, Bypass) and 0GT.D0ZZ (Resection of Aortic Body, Open Approach). Many datasets record (and some people write) codes without the period; for example, the code 090.KXZZ may appear as 090KXZZ. The `icd10pcs` commands understand both ways of recording codes. The commands are also insensitive to codes recorded with or without leading and trailing blanks and are case insensitive.

All the following are acceptable formats to record codes in Stata:

```
03R
      0jj
00f53zz
0TL.COZZ
091
```

Important note: What constitutes a valid code changes between versions. For the rest of this entry, a defined code is any code that is currently valid, was valid at some point since the ICD-10-CM/PCS coding system was introduced, or has a meaning as a grouping of codes. The list of valid codes and their associated descriptions is from the U.S. Centers for Medicare and Medicaid Services (CMS).

To view the current version of the ICD-10-PCS procedure codes in Stata, its source, and a log of changes that have been made to the list of ICD-10-PCS procedure codes since the `icd10pcs` commands were implemented, type

```
. icd10pcs query
ICD-10-PCS Procedure Code Version and Change Log
Note
  Stata obtains the ICD-10-PCS dataset from the Centers for Medicare and
  Medicaid Services website.
(output omitted)
```

Managing datasets with ICD-10-PCS codes

The `icd10pcs` suite of commands has three data management commands. `icd10pcs check` verifies that the ICD-10-PCS procedure codes in *varname* are valid. `icd10pcs clean` standardizes the format of ICD-10-PCS procedure codes in *varname*. And `icd10pcs generate` produces a new variable from an existing variable containing ICD-10-PCS procedure codes.

Examples in this section use `hosp2015.dta`, a fictional sample of inpatient hospital discharges in Washington state from July 2015 to December 2015. The data were simulated based on the Comprehensive Hospital Abstract Reporting System (CHARS); see <https://www.doh.wa.gov/DataandStatisticalReports/HealthcareinWashington/HospitalandPatientData/HospitalDischargeDataCHARS>. Examples analyzing the diagnosis codes for this dataset can be found in [D] [icd10cm](#).

```
. use https://www.stata-press.com/data/r18/hosp2015
(Fictional WA hospital discharges)
```

`icd10pcs check` is the primary subcommand for validating ICD-10-PCS procedure codes. However, if you just want to verify that the codes conform to the formatting rules for ICD-10-PCS procedure, you can use the `check` option with `icd10pcs clean` or `icd10pcs generate`.

▷ Example 1: Checking for valid code values

You use `icd10pcs check` just like you do `icd10cm check`. Because the data are from federal fiscal year 2016, we specify `version(2016)`.

In [example 1](#) of [\[D\] icd10cm](#), we found that we needed to account for the date of the admission when we used the `icd10cm` commands. The same is true of the `icd10pcs` commands because the two systems were implemented simultaneously. We preemptively exclude records before October 2015 here.

```
. drop if dmonth < tm(2015m10)
(1,955 observations deleted)
. icd10pcs check proc1, version(2016)
(proc1 contains defined codes; 594 missing values)
```

We find that there are no errors in the coding of the `proc1` variable and that 594 records in our dataset did not have any procedure at all.

◀

If we wanted to check codes in more than one procedure variable, we could use a `foreach` loop or `reshape` our data; see [Working with multiple codes](#) in [\[D\] icd](#). With large datasets, it is generally faster to use a loop.

It is a good idea to begin with `icd10pcs check` and fix any potential problems before proceeding to other `icd10pcs` commands. The `icd10pcs check` command with the `generate()` or `list` option is also useful for tracking down problems when any of the other `icd10pcs` commands tell you that the variable “contains invalid codes”.

`icd10pcs clean` formats the variable to ensure consistency and to make subsequent output from other commands such as `list` and `tabulate` look better. `icd10pcs clean` also can be used to verify that the codes in a variable conform to the ICD-10-CM format, without checking to see whether the codes are defined.

▷ Example 2: Cleaning an existing variable

We standardize all the ICD-10-PCS procedure codes in `proc1` to include a period after the third character. We specify the `replace` option rather than the `generate()` option so that the values in `proc1` are replaced with their formatted values.

```
. icd10pcs clean proc1, replace
variable proc1 was str7 now str8
(1,980 real changes made)
```

`icd10pcs clean` reports that 1,980 values were replaced. If we wanted to standardize to a format without the period, we could have specified the `nodots` option.

◀

Aside from validating values of codes, the `icd10pcs` command is primarily used to create inputs for other Stata commands. For example, in [example 5](#) of [\[D\] icd9](#), we show how to graph the frequency of category codes with descriptions, and in [example 3](#) of [\[D\] icd10cm](#), we show how to graph summary statistics by diagnosis.

▷ Example 3: Creating an indicator for common procedures

If we use `tabulate` on the primary procedure code (`proc1`) the same way we did for the primary diagnosis in [example 2](#), we find that the three most frequent primary procedure codes in our data are 10E0XZZ, 10D00Z1, and 0SRC0J9. Suppose we want to know the average billed amount (`billed`) for all admissions that had one of these procedure codes in the primary procedure field.

Our first step is to create an indicator for whether one of these codes is present in `proc1`. Then, we summarize `billed` over the three top values of `proc1` by using `tabulate`; see [\[R\] `tabulate, summarize\(\)`](#).

```
. icd10pcs generate top3 = proc1, range(10E0XZZ 10D00Z1 0SRC0J9)
. tabulate proc1 if top3==1, summarize(billed) freq means
```

Procedure 1	Summary of Amount billed (\$1,000s)	
	Mean	Freq.
0SR.C0J9	60.62	40
10D.00Z1	27.55	92
10E.OXZZ	14.05	180
Total	24.00	312

We find that the highest average billed amount for the top three codes is for ICD-10-PCS procedure code 0SR.C0J9. There are 40 discharges in our dataset with this code as their principal procedure, and their average billed amount is about \$60,620.

◀

Interactive utilities

`icd10pcs lookup` and `icd10pcs search` are interactive tools. You can use them without having any ICD-10-PCS procedure data in memory.

`icd10pcs lookup` lists the descriptions of codes given on the command line, and `icd10pcs search` looks for relevant ICD-10-PCS procedure codes from the specified keywords. The two commands complement each other.

▷ Example 4: Finding procedure code descriptions

Suppose we wanted to find the short descriptions of the most frequent codes in our dataset. We can supply `icd10pcs lookup` with the same list of codes we used in [example 3](#).

```
. icd10pcs lookup 10E0XZZ 10D00Z1 0SRC0J9, version(2016)
    OSR.C0J9 Replace of R Knee Jt with Synth Sub, Cement, Open Approach
    10D.00Z1 Extraction of POC, Low Cervical, Open Approach
    10E.OXZZ Delivery of Products of Conception, External Approach
```

We see, for example, that ICD-10-PCS procedure code 0SR.C0J9 is for a type of knee replacement surgery.

◀

Using `icd10pcs search` is similar to using `icd10cm search`. See [example 4](#) in [\[D\] `icd10cm`](#).

Stored results

`icd10pcs check` stores the following in `r()`:

Scalars

<code>r(e#)</code>	number of errors of type #
<code>r(esum)</code>	total number of errors
<code>r(miss)</code>	number of missing values
<code>r(N)</code>	number of nonmissing values

`icd10pcs clean` stores the following in `r()`:

Scalars

<code>r(N)</code>	number of changes
-------------------	-------------------

`icd10pcs lookup` and `icd10pcs search` store the following in `r()`:

Scalars

<code>r(N_codes)</code>	number of codes found
-------------------------	-----------------------

Acknowledgments

We thank the Washington State Department of Health's Center for Health Statistics for providing us with access to its 2015 Comprehensive Hospital Abstract Reporting System (CHARS) inpatient dataset. The `hosp2015` dataset used here was partially simulated based on information from the 2015 limited use CHARS. We also thank Jeanne M. Sears of the University of Washington for bringing the CHARS to our attention.

We thank Joe Canner of Yale University School of Medicine, who wrote `mycd10` and `mycd10p`, which provide many utilities for ICD-10 diagnosis and procedure codes. The commands rely on a user-supplied ICD-10 lookup dataset for diagnosis codes and ICD-10-PCS codes from the U.S. Centers for Medicare and Medicaid Services for procedure codes.

Also see

[D] [icd](#) — Introduction to ICD commands

[D] [icd9p](#) — ICD-9-CM procedure codes

[D] [icd10cm](#) — ICD-10-CM diagnosis codes

[D] [frunalias](#) — Change storage type of alias variables

Stata, Stata Press, and Mata are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow and NetCourseNow are trademarks of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2023 StataCorp LLC, College Station, TX, USA. All rights reserved.



For suggested citations, see the FAQ on [citing Stata documentation](#).