

How do I retrieve the version of the Java Compiler and the isCOBOL Compiler that produced a given class file?

Author: Veryant Support

Saved From: <http://support.veryant.com/support/phpkb/question.php?ID=271>

The compilation of a program with isCOBOL is done in two steps. In the first step the isCOBOL Compiler (iscc) parses the COBOL source code and produces an intermediate Java source. In the second step the Java Compiler (javac) compiles the Java source into a class file.

In order to know the version of the isCOBOL Compiler used to produce the class file, run the following command:

```
isccrun -info PROGRAM.class
```

The command output will contain this text:

```
COBOL: java OBJECTs
```

It means that the program was compiled with a newer version of isCOBOL and your old version is not able to process the class, or

```
COBOL: compiled with isCOBOL build #??? ....
```

Build number	isCOBOL version
261	2007
338	2007.1
413	2008
433	2008.1
510	2009
530	2009.1
546	2009.2
570	2010 R1
600	2010 R2
626	2010 R3
641	2011 R1
663	2011 R2
672	2011 R3
681	2012 R1
705	2012 R2
723	2013 R1
745	2013 R2
780	2014 R1
821	2015 R1
855	2016 R1
875	2016 R2

893	2017 R1
910	2017 R2
930	2018 R1
950	2018 R2
977	2019 R1
995	2019 R2
1011	2020 R1
1023	2020 R2
1041	2021 R1
1050	2021 R2
1060	2022 R1
1074	2022 R2
1090	2023 R1
1105	2023 R2

The ??? number tells you the build number. Here's a table of the build numbers and the isCOBOL version that matches it. In order to check the Java version number in a .class file you can use the javap utility that is distributed with the JDK.

For example, the following command will report the version of a program named MYPROG.

```
javap -verbose MYPROG | grep major
```

or run

```
javap -verbose MYPROG
```

and the version numbers will be reported in the first 5 lines.

Note that MYPROG.class must be in the class path.

You can specify a class path with "javap -cp".

The major (or "magic") version is mapped to the Java SE version [here](#).