OASIS HEALTH CHECK



TABLE OF CONTENTS

[1.1 Summary 2](#_Toc524506800)

[1.2 Intended Audience 2](#_Toc524506801)

[1.3 Command Line Help 2](#_Toc524506802)

[1.4 Installation 3](#_Toc524506803)

[1.4.1 Requirement 3](#_Toc524506804)

[1.4.2 Steps 3](#_Toc524506805)

[1.5 Uninstallation 5](#_Toc524506806)

[1.6 Run OASIS Health Check – command line 5](#_Toc524506807)

[1.7 Run OASIS Health Check – file 7](#_Toc524506808)

[1.8 Display Health Checks 7](#_Toc524506809)

[1.9 Validations 8](#_Toc524506810)

[1.10 Health Check Interpretation 9](#_Toc524506811)

[1.11 Health check results intrepretation 11](#_Toc524506812)

[1.11.1 Parameter configuration missing 11](#_Toc524506813)

[1.11.2 Invalid parameter value 11](#_Toc524506814)

[1.11.3 Parameter validation failure 11](#_Toc524506815)

[1.11.4 Incorrect parameter value 12](#_Toc524506816)

[1.12 Report 12](#_Toc524506817)

[1.12.1 Last Run Report 12](#_Toc524506818)

[1.12.2 Report of a Particular Timestamp 13](#_Toc524506819)

[1.12.3 All Reports 14](#_Toc524506820)

[1.13 Examples of checks performed, and remedies made in environments 14](#_Toc524506821)

[1.13.1 ODEV20181SE 14](#_Toc524506822)

## Summary

OASIS Health Check is a health check tool for OASIS environment. Every run of this tool goes through routine checks on configuration to check for validity of the configuration. It creates a report of every run in a table, a report of which can be downloaded as a comma-separated file.

OASIS Health Check is available through a web download at this link: <link>

## Intended Audience

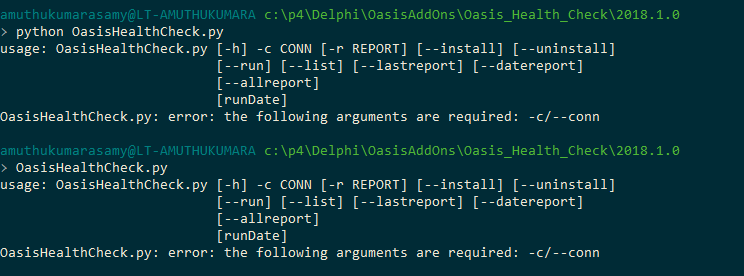
This document is intended to help orient DBAs and system administrators to several database and system maintenance-related tasks and the tools at their disposal.

## Command Line Help

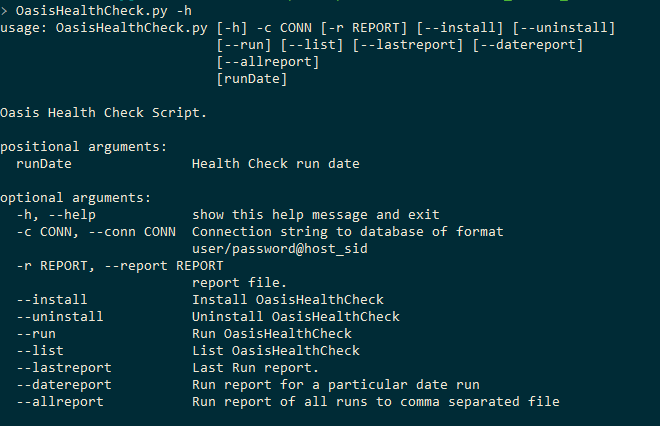
To get command line options to run the tool, enter the following command:

**OASISHealthCheck.py (if Python is installed and available in the Windows path)**

**python OASISHealthCheck.py (if Python is installed but not available in the Windows path)**



**Python OASISHealthCheck.py -h (detailed help messages)**



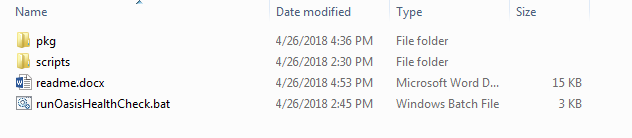
## Installation

### Requirement

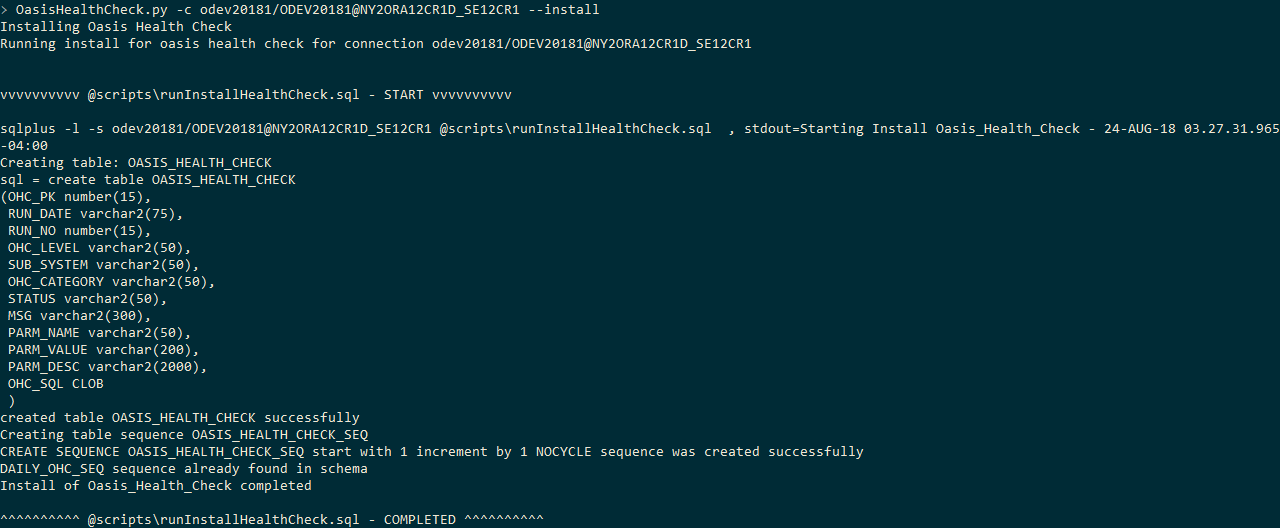
Python is required to run the oasis health check script. Oasis health check tool was tested and certified with Python 3.6.4

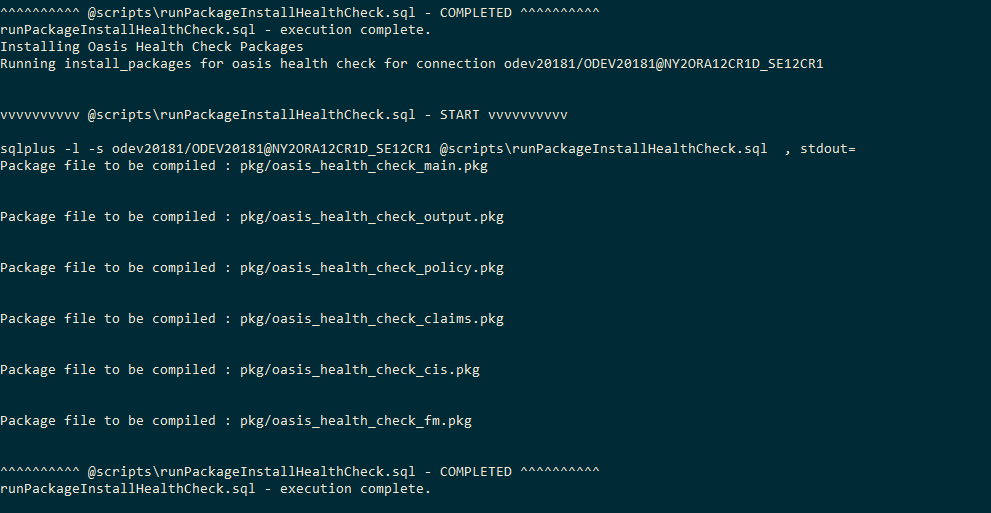
### Steps

1. Download the zip file.
2. Unzip to a local directory. Contents of the directory appear as below:



1. Open command line window and use the cd command to connect to the OASIS health check installation download directory.
2. Run this command:   
   **python OASISHealthCheck.py -c odev20181/ODEV20181@NY2ORA12CR1D\_SE12CR1—install**   
   or  
   **OASISHealthCheck.py -c odev20181/ODEV20181@NY2ORA12CR1D\_SE12CR1 --install**





An install log is created in the root directory.

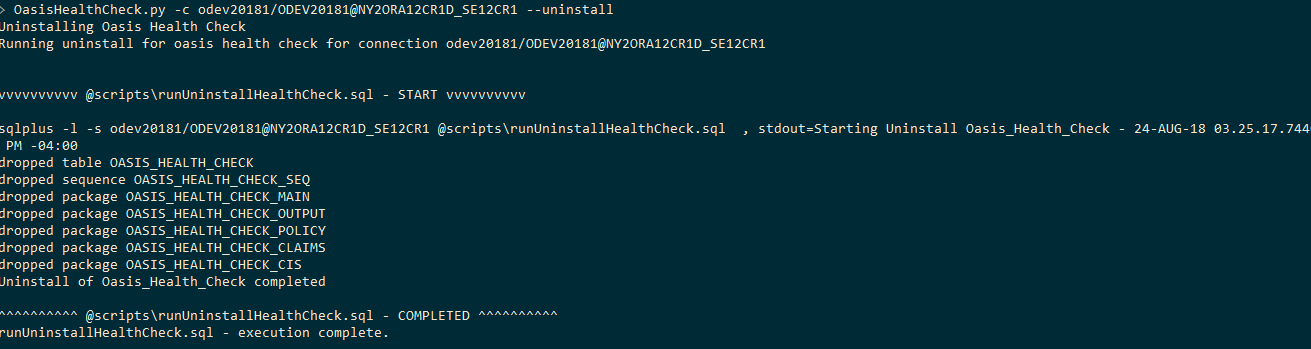
## Uninstallation

Run this command in the command line window:

**OASISHealthCheck.py -c odev20181/ODEV20181@NY2ORA12CR1D\_SE12CR –uninstall**

or

**python OASISHealthCheck.py -c odev20181/ODEV20181@NY2ORA12CR1D\_SE12CR --uninstall**



An uninstall log is created in the root directory.

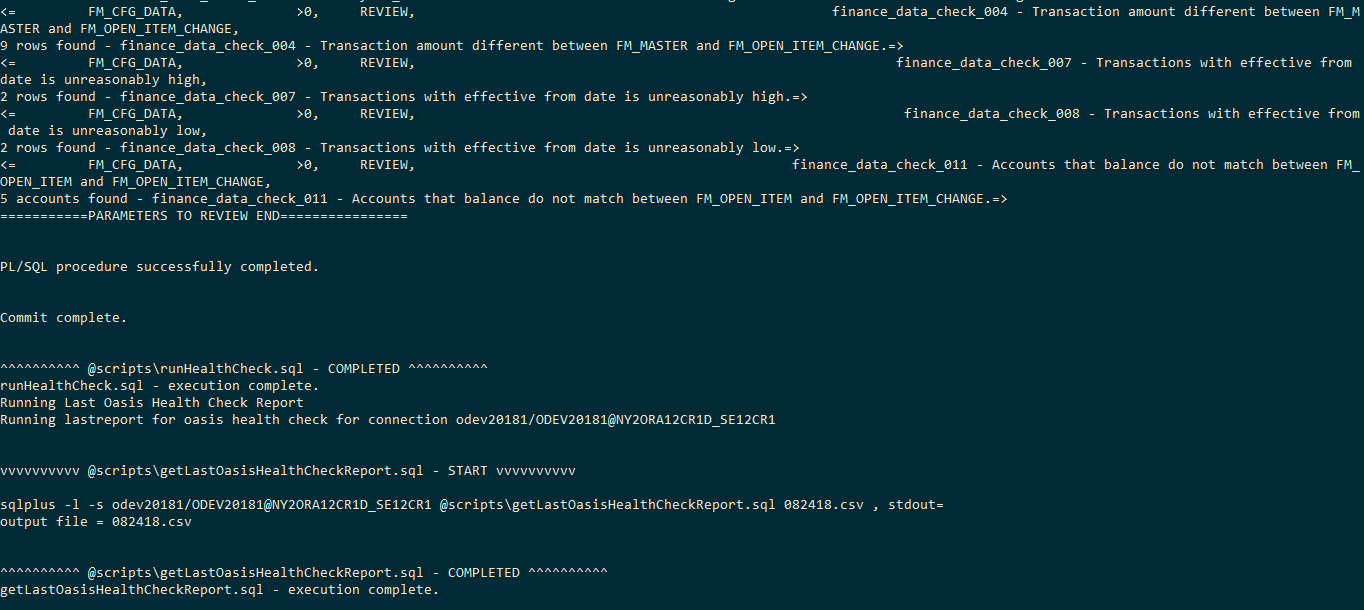
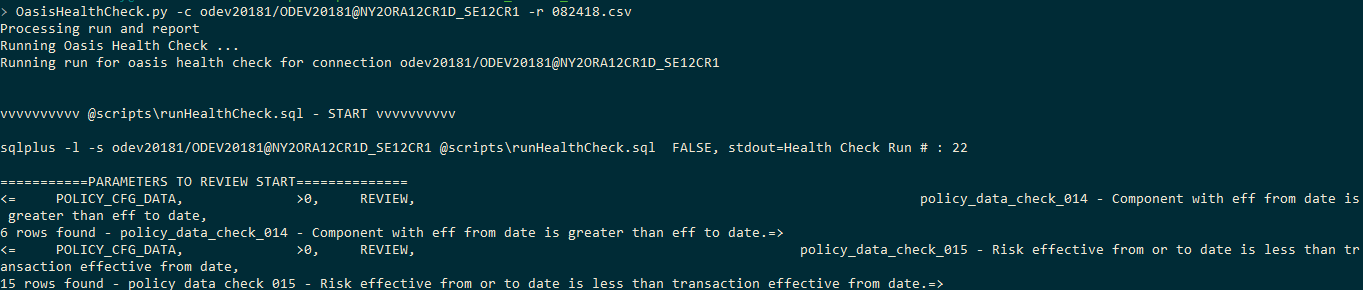
## Run OASIS Health Check – command line

To run OASIS Health Check tool after installation is completed, type in the following command:

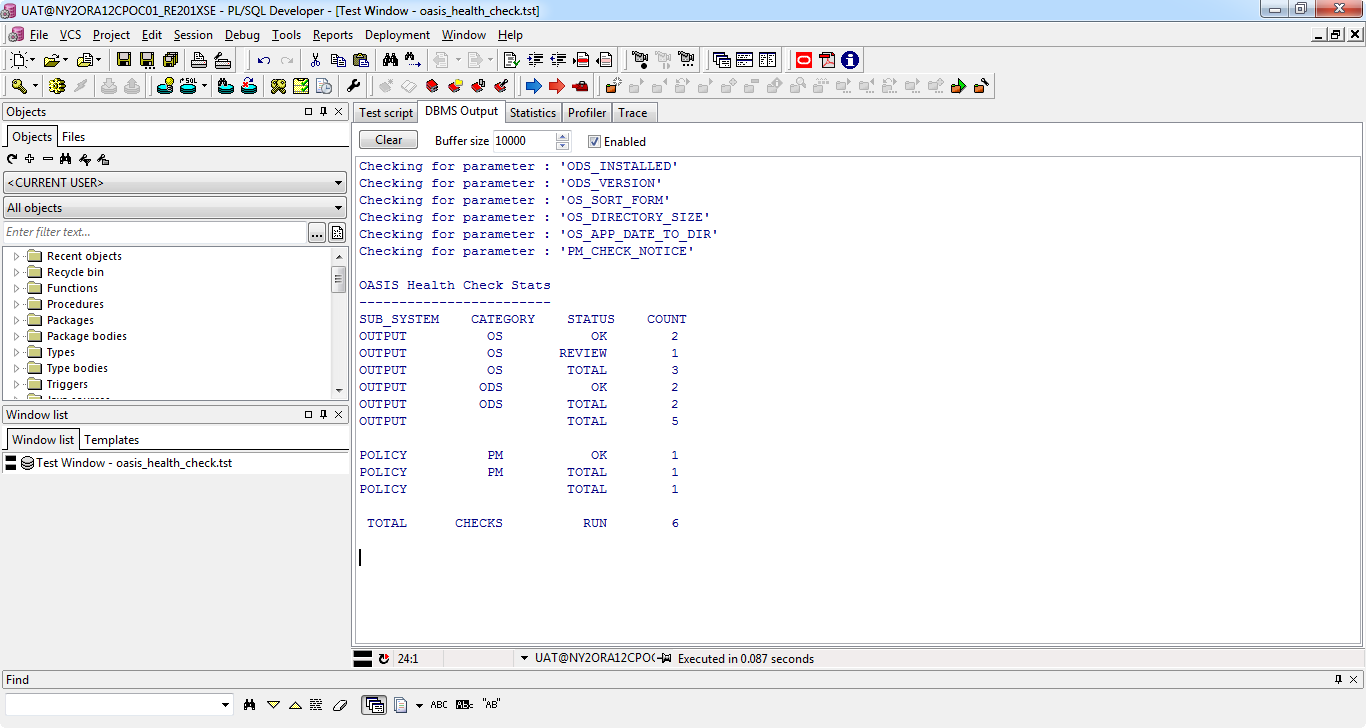
**OASISHealthCheck.py -c odev20181/ODEV20181@NY2ORA12CR1D\_SE12CR1**

or

**python OASISHealthCheck.py -c odev20181/ODEV20181@NY2ORA12CR1D\_SE12CR1**



A test file in “scripts/oasis\_health\_check.tst” can be used to run manually using PL/SQL Developer.



## Run OASIS Health Check – file

OASIS Health Check can be run with the environments listed in a file in the following format

***<Key> <Connection String in the format user/password@host\_sid>***

Refer to “oasishealthcheckenv.txt” in the oasis health check tool directory.

After creating the environment file, the following command would run the health check on the environment pointed by the key,

**OASISHealthCheck.py -f <filename> -k <key>**

If no key is provided, health check would run for all the environments specified in the health check environments file for the command as given below,

**OASISHealthCheck.py -f <filename>**

## Display Health Checks

Heath checks that are run for each subsystem can be listed by using the following PL/SQL block:

**begin**

oasis\_health\_check\_main.display\_health\_checks(subsystem => '');

**end**;

/

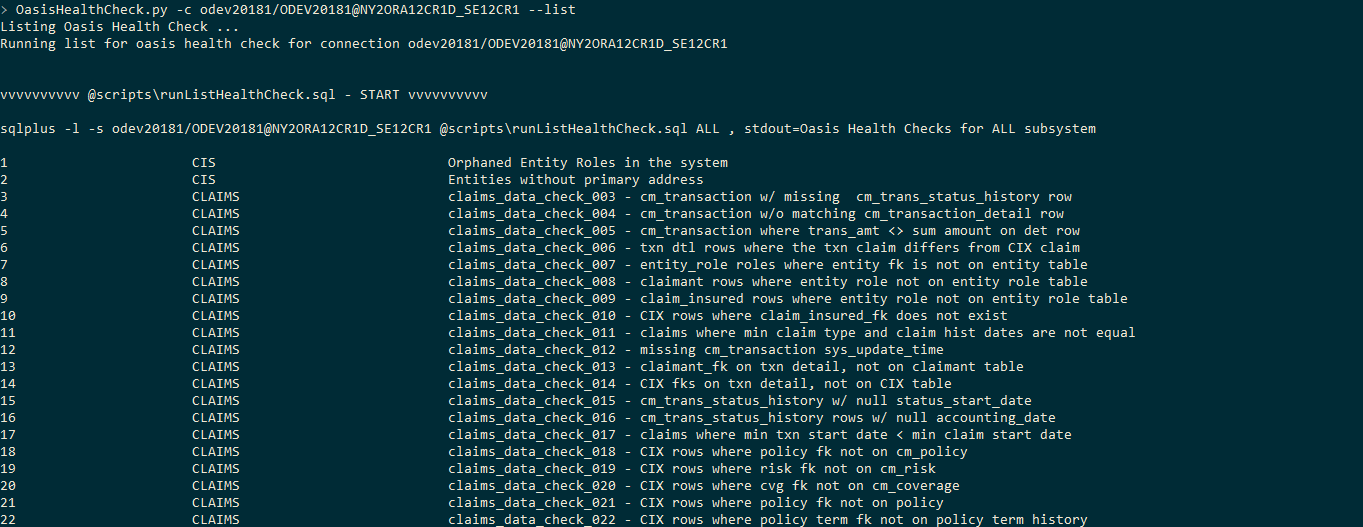
Or

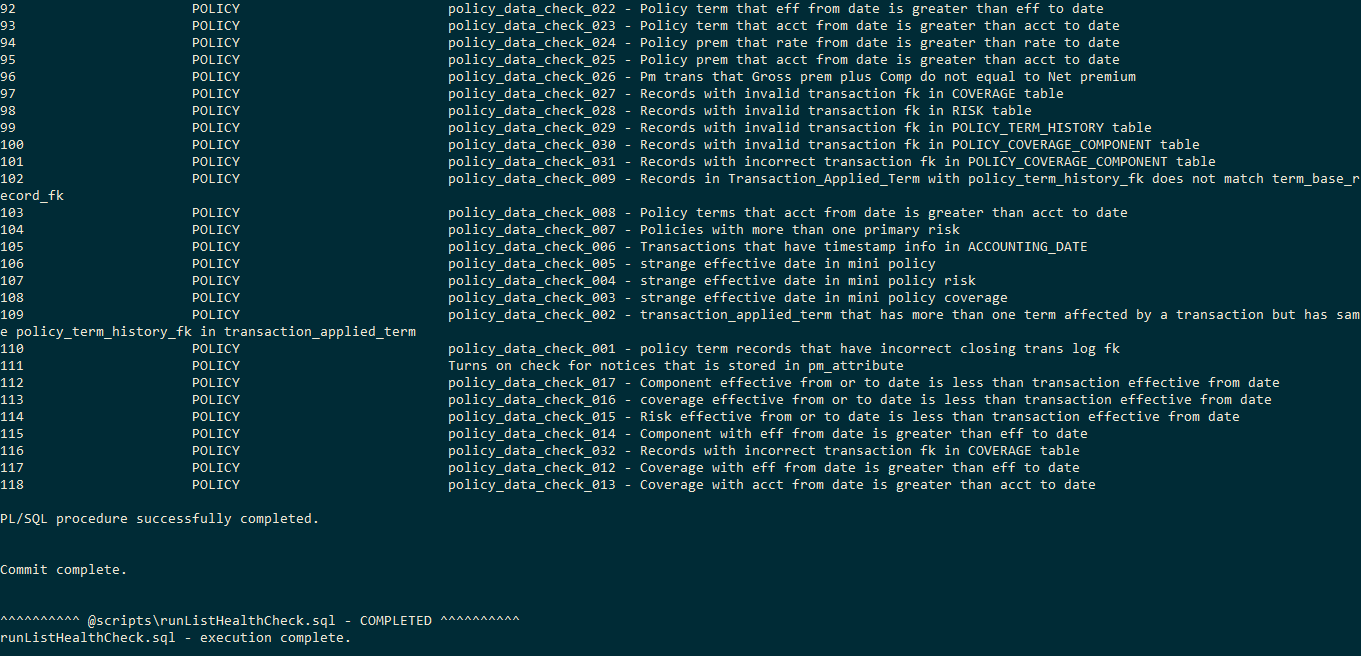
Issue the following command:

**OASISHealthCheck.py -c odev20181/ODEV20181@NY2ORA12CR1D\_SE12CR1 –list**

Or

**python OASISHealthCheck.py -c odev20181/ODEV20181@NY2ORA12CR1D\_SE12CR1 –list**

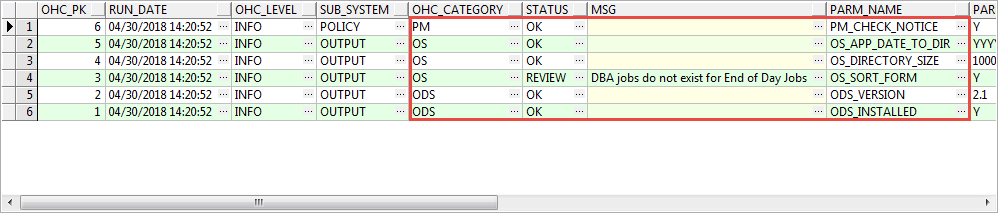




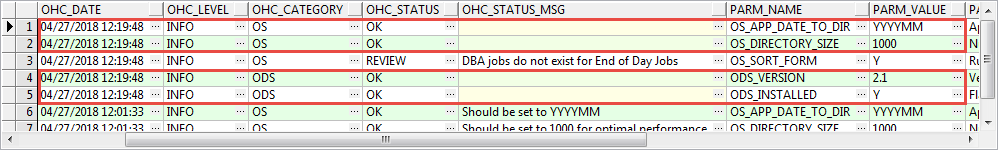
## Validations

Currently, OASIS health check tool checks for two OASIS Documentation System parameters, three operating system parameters and one Policy Management parameter. The table lists all the details of the validation.

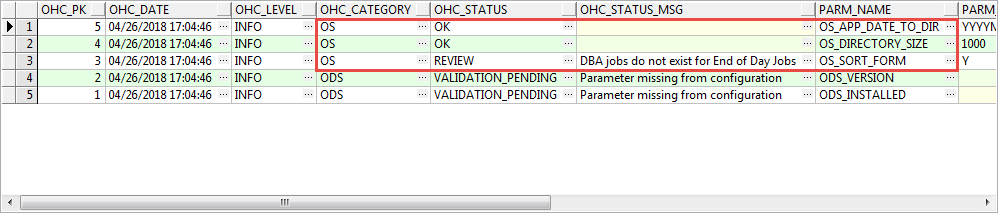
1. Tool reports with three status codes:
   1. OK – Validation was completed and was found to be the expected value.
   2. REVIEW – Validation was completed and found an issue with the configuration. There is a status message field that gives some information on what the problem is for that parameter.
   3. PENDING – Validation was not done due to an issue with the configuration.
2. Three categories of validations as of now.
   1. ODS
   2. OS
   3. PM
3. Validation level can be used later to give priority to a validation. Here the validations are at INFO level.



## Health Check Interpretation



All expected parameter values are displayed with a status of OK wand no status message.

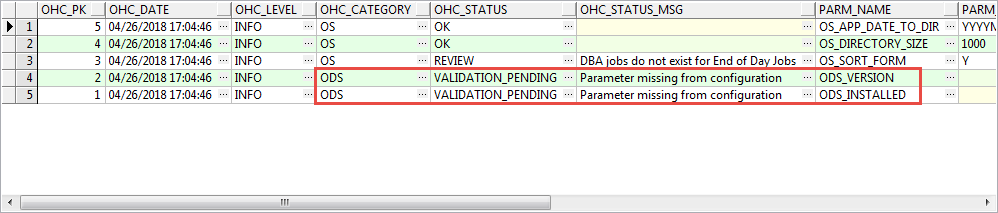


The examples show two types of validations.

* OS\_APP\_DATE\_TO\_DIR and OS\_DIRECTORY\_SIZE are in the OK status with no status message.
* A ‘REVIEW’ status would accompany with a message with an explanation of which validation failed.

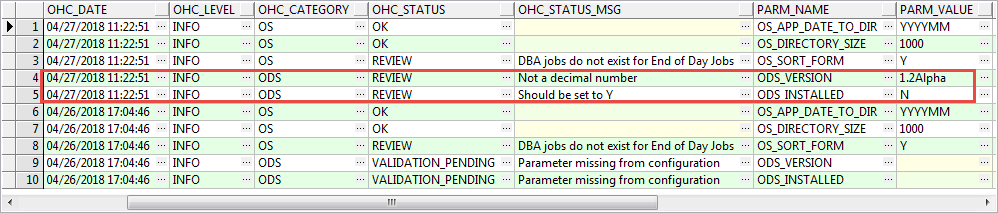
## Health check results intrepretation

### Parameter configuration missing



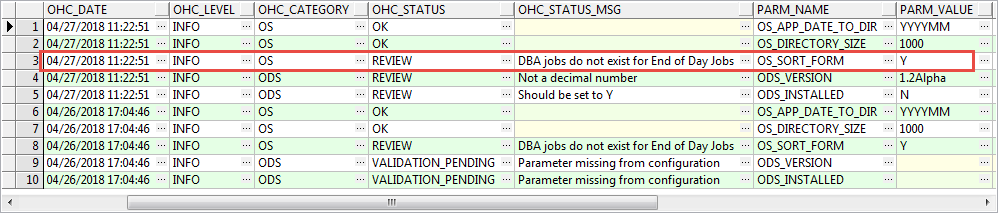
ODS\_INSTALLED and ODS\_VERSION parameters are not set. Status is set to ‘Validation\_Pending’ since validation was not completed for the missing parameters.

### Invalid parameter value



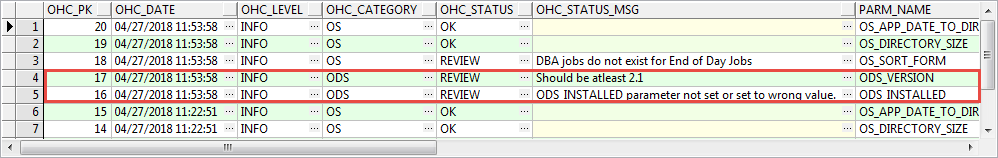
* ODS version should be a float number
* ODS\_INSTALLED is expected to be Y

### Parameter validation failure

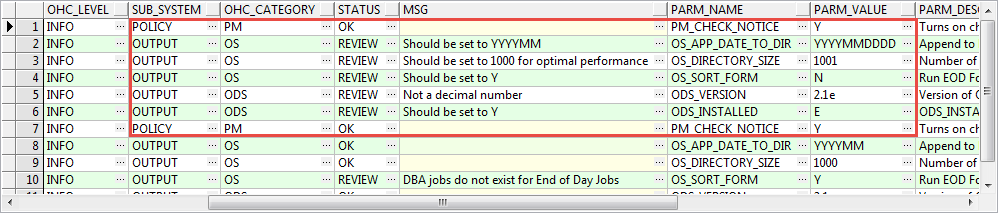


If OS\_FORM is set to ‘Y’, jobs must be present to handle EOD for a form that is configured to be End Of Day. Since it did not find that job, this is set to REVIEW.

### Incorrect parameter value



* ODS\_Version should equal 2.1
* ODS\_INSTALLED is expected to be set to ‘Y’



* OS\_APP\_DATE\_TO\_DIR is expected to be set as ‘YYYYMM’
* OS\_DIRECTORY\_SIZE is expected to be set to 1000
* OS\_SORT\_FORM is expected to be set to Y
* ODS\_VERSION is expected to be 2.1

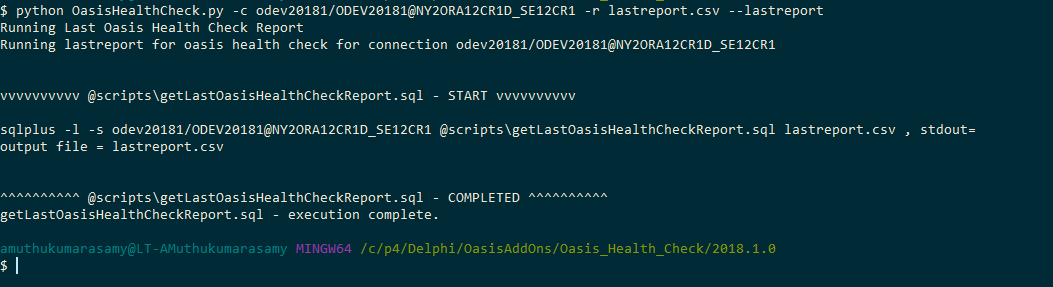
## Report

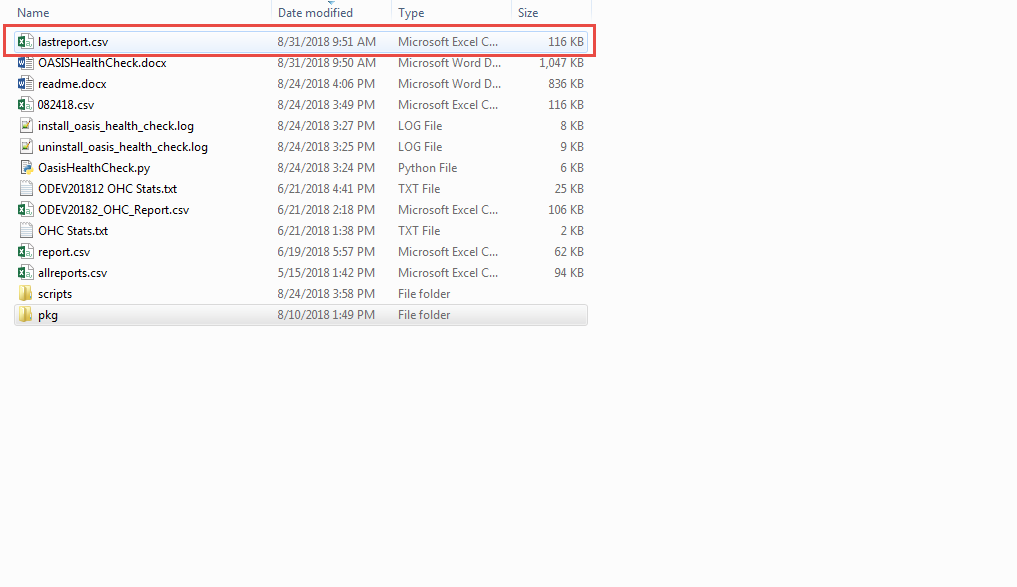
### Last Run Report

To get a last run report in a comma-separated file, type in the following command: **python OASISHealthCheck.py -c odev20181/ODEV20181@NY2ORA12CR1D\_SE12CR1 –lastreport**

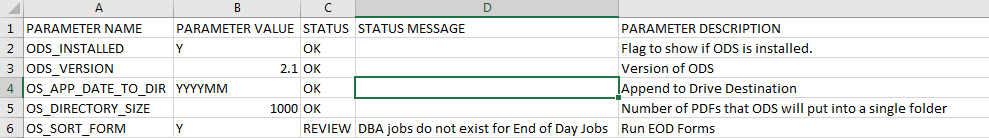
**Or**

**python OASISHealthCheck.py -c odev20181/ODEV20181@NY2ORA12CR1D\_SE12CR1 –lastreport**





Example of a report:



### Report of a Particular Timestamp

To get a report for a timestamp in a comma-separated file, type in the following command:  **OASISHealthCheck.py -c odev20181/ODEV20181@NY2ORA12CR1D\_SE12CR1 -r datereport.csv –datereport “04/26/2018 14:39:28”**

**Or**

**python OASISHealthCheck.py -c odev20181/ODEV20181@NY2ORA12CR1D\_SE12CR1 -r datereport.csv –datereport “04/26/2018 14:39:28”**

### All Reports

To get all health check reports in a comma-separated file, type in the following command:   
**OASISHealthCheck.py -c odev20181/ODEV20181@NY2ORA12CR1D\_SE12CR1 -r allreport.csv –allreport**

**Or**

**python OASISHealthCheck.py -c odev20181/ODEV20181@NY2ORA12CR1D\_SE12CR1 -r allreport.csv –allreport**

## Examples of checks performed, and remedies made in environments

### ODEV20181SE

* Run health check
* Illustrate an error problem scenario
* Fix an error
* Run health check
* Illustrate the error as resolved