

(November 21st, 2024)

If you know someone who would benefit from being an Insider, feel free to forward this PDF to them so they can sign up [here](#).



Note: As an Insider, you can read all prior Insider newsletters [here](#).

Quick Tips for our Insider friends!

This mini-newsletter comes to you from Camp Savage again, where I survived the ‘bomb cyclone’ overnight on Tuesday, miraculously with zero damage at all! Redmond and surroundings are apparently a disaster zone, but luckily nothing hit our house. I’ll find out when I head south tomorrow.

Black Friday Super-Sale!

It’s the time of year again where companies do crazy sales on the day after Thanksgiving in the US, and we’re doing the same!

Starting at 4pm PST on Thursday, November 28th, we’ll be offering our **lowest prices ever** for our signature [Blackbelt bundle](#) with more than 158 hours of top-quality training (plus all our SQL Server 2022 update/Q&A recordings):

- **One-year access to the Blackbelt bundle: US\$990**
- ***Lifetime* access to the Blackbelt bundle: US\$1,895**

We’ll also have the following special discounts:

- [Our IE0 training class](#) for Accidental and Junior DBAs (over 32hrs; not included in the Blackbelt bundle):
 - **One-year access to IE0: US\$595**
 - ***Lifetime* access to IE0: US\$1,185**
- [Our IEQuery training class](#) on slow queries, inefficient code, and caching/statistics problems (over 11hrs; not included in the Blackbelt bundle) on sale!
 - **One-year access to IEQuery: US\$119**
- [Our IETLB training class](#) on Transactions, Locking, Blocking, Isolation, and Versioning (over 11hrs; not included in the Blackbelt bundle):
 - **One-year access to IETLB: US\$119**
- [Our JumpStart Bundle](#) (all three of the above courses):
 - ***Lifetime* access to Jumpstart Bundle: US\$1,299**
 - Buy it once and I’ll make it Lifetime

- [Our JumpStart+ Bundle](#) (IE0, IEQuery, IETLB, IECS, IEQS, IEVLT):
 - ***Lifetime* access to Jumpstart+ Bundle: US\$1,599**
 - Buy it once and I'll make it Lifetime

From Monday, December 2 12:00am GMT (starting at 4pm PST on Sunday, December 1) and until 12:00am GMT December 17 (ending at 4pm PST on Monday, December 16), we will offer lower than our regular prices but not as deep of discounts as Black Friday. If you've already purchased a bundle or class once, you can purchase the one-year access for a second time and we'll convert your existing access into lifetime access, by removing all the expiration dates.

Any questions, please [let me know](#).

Book Review

One of the books I read earlier this year was [Shōgun, Part One](#) by James Glavell, another book I've always meant to read, and wow - I just loved it! (No I haven't seen the TV adaption and have no desire to.) From Amazon: "After Englishman John Blackthorne is lost at sea, he awakens in a place few Europeans know of and even fewer have seen--Nippon. Thrust into the closed society that is seventeenth-century Japan, a land where the line between life and death is razor-thin, Blackthorne must negotiate not only a foreign people, with unknown customs and language, but also his own definitions of morality, truth, and freedom. As internal political strife and a clash of cultures lead to seemingly inevitable conflict, Blackthorne's loyalty and strength of character are tested by both passion and loss, and he is torn between two worlds that will each be forever changed." Brilliant - highly recommended!

Ponderings...

(From me this time. I was going to write about not reinventing the wheel but I found that I'd already written about that, so now I'm not reinventing the wheel when talking about reinventing the wheel. How meta! ☺)

Something I've seen repeatedly since I started working with SQL Server (on the SQL Server team back in February 1999) is people reinventing the wheel. And I don't just mean with T-SQL scripts, I also saw it over and over inside Microsoft, where developers would want to write their own version of code to do function X rather than re-using someone else's code that already does function X.

Writing code, in whatever language, to do something that someone else has already coded is usually a big waste of time, unless the only example you can find is poorly written or you really want to understand how something works, and writing code to use it would provide that understanding.

A prime example of this, that I see constantly, is writing a query involving DMVs. As you probably already know, many of the DMVs don't provide really useful (or actionable) information unless the results are manipulated in some way or joined with another DMV or catalog view.

An example would be one of my favorite DMVs: *sys.dm_io_virtual_file_stats*. On its own, it gives the cumulative read and write latencies for all I/Os done for all files in the instance, and the files are identified by database ID, file ID, and a hexadecimal file handle.

This information isn't really useful in that form. By doing some math on the output, basically converting the cumulative latencies into per-read/write latencies, and joining with *sys.master_files* to get file names and paths, the output becomes readable and easily understood.

Did I write the code to do that? No, I used some code that Jimmy May wrote a long time ago and whenever I show it when teaching, I give credit to Jimmy for it.

There's no shame in using or adapting code that someone else wrote and published. People in the SQL Server community publish code on blogs specifically so you can use it to save yourself time and get useful information from SQL Server.

If I'm ever going to write a new DMV-based query, I always check first to see if someone has already done the work – especially if I know it's going to involve joining with a few other DMVs and system catalogs and I'm going to have to play around to figure out which columns to join on.

I don't like reinventing the wheel – I'm too busy to spend my time doing that, and I think you are too. Here are a few links to scripts on SQLskills you may find helpful:

- Kimberly's super-duper [sp_helpindex rewrite](#)
- Kimberly's script to [identify duplicate indexes](#) (with [background reading](#))
- Jonathan's scripts to [identify problems using analysis of the plan cache](#)
- Erin's collection of scripts for [producing a performance baseline](#)
- Some of my more useful scripts:
 - [Wait statistics main script](#) and lots of links to other scripts
 - [I/O latencies](#)
 - [How much has changed since the last full backup](#)
 - [Is a database really in the full recovery mode](#)
 - [All open transactions for a database](#)
 - [Wasted space in the buffer pool](#)
 - [Index counts for all tables](#)
 - [Potential space savings from changing a cluster key](#)
 - [Identifying queries with SOS_SCHEDULER_YIELD waits](#)

And I remember the very first public SQL Server script I published, which was Example E in the Books Online entry for *DBCC SHOWCONTIG*, which I'd just rewritten for SQL Server 2000, and the script made use of the new *DBCC INDEXDEFRAG* I'd written. It's still there in the [official docs](#) even today in 2024! Nowadays I'd always recommend someone uses (or uses code from) Ola Hallengren's excellent [Maintenance Solution](#) (yes, other people have published scripts to do similar, but Ola's are the gold standard IMHO so I won't recommend anything else).

Call to action: Any time you're trying to figure out a script to do X or Y or use DMV Z, if you just want the job done and you're not trying to learn the ins-and-outs of the DMV or how SQL Server works, do a quick Google or Bing search to see if someone already written a script, and then if you can't find anything ask on X (Twitter) using the #sqlhelp tag and if a script exists, someone will point you to it. Don't reinvent the wheel if you really don't need to.

#TBT

(Turn Back Time...) Blog posts we've published since the previous newsletter plus some older resources we've referred to recently that you may find useful.

In this TBT I want to point you at a 30-part blog post series that the team did a few years ago that covers a lot of introductory topics, aimed at accidental/involuntary DBAs. Rather than listing all 30 posts, [here](#) is a link to the main page for the series. The series covers these main areas:

- Setup and configuration
- Backups
- Implementing maintenance and security
- Troubleshooting performance
- High availability
- General troubleshooting
- Although the series was aimed at accidental DBAs, these posts are all worth reading to make sure your basic understanding of topics is current and complete.

I hope you find these useful and interesting!

Video Demo

The video this time is taken from my Pluralsight course [SQL Server: Understanding and Performing Backups](#). In the demo I explain why a full backup sometimes needs to include extra transaction log if there's an active transaction at the time the full backup performs its initial checkpoint operation.

The video is 5.5 minutes long and you can get it in MP4 format [here](#).

The demo code is [here](#).

Enjoy!

Upcoming SQLskills Events

Given how little changed with SQL Server 2022, we decided to record a series of updates relevant to our class material rather than running full classes. These are done as part of our Blackbelt Base Camp series and are available in our shop, and are free with a Blackbelt bundle.

With our streaming system, you can now choose to attend a live, online event or purchase a recording to watch at your leisure, either individually or as part of a bundle. And all attendees of live events get lifetime access to the class recordings too!

To help your boss understand the importance of focused, technical training, we've also added a few items to help you justify spending your training dollars with us:

- [Letter to your boss explaining why SQLskills training is worthwhile](#)
- [Community blog posts about our classes](#)
- [Immersion Event FAQ](#)

You can get all the details on our [training options page](#) or just go directly to our [shop](#).

Summary

I hope you've enjoyed this issue – I really enjoy putting these together. If there's anything else you're interested in, I'd love to hear from you - [drop me a line](#).

Thanks,
Paul

Paul@SQLskills.com