**Server Configuration Checklist**

* **CPU**:
  + How many logical cpus?
  + Check in the task manager 🡪 performance 🡪 CPU for sockets , cores, hyper-threading (virtalization enabled)
  + Logical or physical server
  + In the instance properties – make sure CPU affinity is NOT checked
* **Memory**:
  + How much memory does the server have?
  + In the instance properties - check max server memory (right click on instance for properties)– should not be higher than the recommended value calculated by this script:  
    
  + Minimum server memory – if sql server is the only thing running on the server maybe you should configure this to reduces cases when an external process takes memory from sql server.
* **Instance Configurations (right click instance 🡪 properties):**
  + Change SA user's name and disable it.
  + Allow remote connections is checked
  + Auditing should be only for failed logins unless there is a good reason so the log won't blow with unneeded messages.
  + Authentication mode – know what it is configured to. Best practice is windows.
  + Check if default fill factor is not 0 – make a note if it is different.
  + Check compress backup is checked.
  + Recovery interval should be 0. Otherwise affects the checkpoint timing.
  + File default locations should be changed to the proper locations.
  + Advanced tab for the instance:
    - Optimize for ad-hoc workloads – should be checked for workloads that have a lot of ad-hoc queries. When turned on, once used plans are not taking up space in the cache.
    - Network packet size – usually the app decides on connection, here it is more for linked servers. Best for bulk insert. Bigger packet size is cutting the data to less chunks.
    - Cost threshold for parallelism – when queries cost is over this threshold parallelism will be considered. Best practice start at 50 for OLTP workload and check from there.
    - Max degree of parallelism – how many threads parallelism will use – we don't want one query to take it all and there is a managing overhead. Best practice is 8. For new versions can be configured at the database level.
    - **These options could clear the plan cache!**
* **Database level configurations**:
  + Database owner should be SA (even if disabled).
  + Files:
    - Initial size – File size should be big enough for the future but not too big so it requires too much storage.
    - Autogrowth – should not be percent and should grow by a reasonable amount (not too much not too little)
  + Recovery model – if FULL requires log backups. When in SIMPLE data can be lost up to the last full backup.
  + Autoshrink should be off
  + Autoclose should be off
  + Auto update statistics can be disabled (clears plan cache) and we can run it as a job
  + Cardinality estimator – make note which estimator is being used.
  + Compatibility level – make note which one is being used.
  + Parameter sniffing should be on
* **Instant file initialization** – when new storage needs to be given to a file the operating system will zero it out first (initialize). If is set ON there are will be no initializing with zeros which is faster. Under Security🡪 users🡪 give instant file initialization permission to the user that is running SQLServer. If creating a large file is fast then instant file initialization is ON.
* **Tempdb**:
  + 4 data files or 8 with the exact same size and same autogrowth. More files will remediate contention on IAM, SGAM page when creating temp objects.
  + 1117,1118 trace flags (requires restart) turned ON.
  + Check collation is the same with other databases and instance. Otherwise collation changes must be made when using temp objects.
* **Service level (Configuration Manager):**
  + SQL Server and Agent service user – should not be a local system account because it allows access to all disks. Agent and Engine services should have different users.
  + For security - change default port as well.
* **Maintenance:**
  + Avoid running shrink on files regularly.
* **Tools:**
  + Configuration Manager
  + Task Manager
  + Sql query Stress test – simulates stress on databases. Can be used to check tempdb contention (if there are current waiting tasks on pages with database\_id = 2).
  + Snippets:
    - Show current running queries
    - Waiting tasks