

# Shinydocs Wildcard Index Update Script: Improved wildcard searching

Shinydocs Wildcard Index Update Script (`Shinydocs_Wildcard_Index_Update.ps1`) is a PowerShell script that will add enhanced wildcard search functionality to the selected field(s) in an index. Enhanced wildcard search will allow Discovery Search and Visualizer users to perform Windows-like searches with wildcards.

 During the first part of this process, the index you are applying the enhancements to will be **temporarily unavailable for search and indexing**. Depending on your cluster's hardware, this time should be between **1 - 60 seconds**.

In-house testing with an index of 1.4 million documents resulted in a **downtime of <1 second**.

- [Example of enhanced wildcard search functionality](#)
- [How to use the script Shinydocs\\_Wildcard\\_Index\\_Update.ps1](#)
  - [Requirements:](#)
  - [Script Parameter Requirements](#)
  - [Step-by-step](#)
  - [Validate](#)
- [Logging](#)
- [Previous Versions](#)

Example of enhanced wildcard search functionality

**Scenario:** A search is performed in Discovery Search with the query  
**Green** = Part of the search string

**Purple** = Wildcard, any number of characters (besides special characters)

**Search term:** `ab*d-1*4`

**Resolves to:** `"ab[any characters]d-1[any characters]4"`

**Possible file results with the enhancement:**

- `abcd-1234.docx`
- `ABCd-1234.pptx`
- `abDATAd-1RECORD87234.xlsx`

**Results without enhancement:**

- No results as wildcards in the index do not perform this way normally

How to use the script `Shinydocs_Wildcard_Index_Update.ps1`

**Requirements:**

- An existing Shinydocs index with at least 1 data entry
- Know which field(s) you will be applying this enhancement to
  - We recommend the 'name' field
- The computer running this script must be able to reach the index (ex. `http://server:9200`)
- If you have a local policy preventing PowerShell scripts from being run, that will need to be disabled temporarily while the script runs



Windows has very strict permissions for external PowerShell scripts. You may encounter an error indicating that you are not able to run scripts in PowerShell. Depending on your company's security policies, you may need to either:

- Open PowerShell as Administrator
- Set the appropriate execution policy (See Microsoft Documents: [https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about\\_execution\\_policies?view=powershell-7.2](https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_execution_policies?view=powershell-7.2) & <https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.security/set-executionpolicy?view=powershell-7.2>)

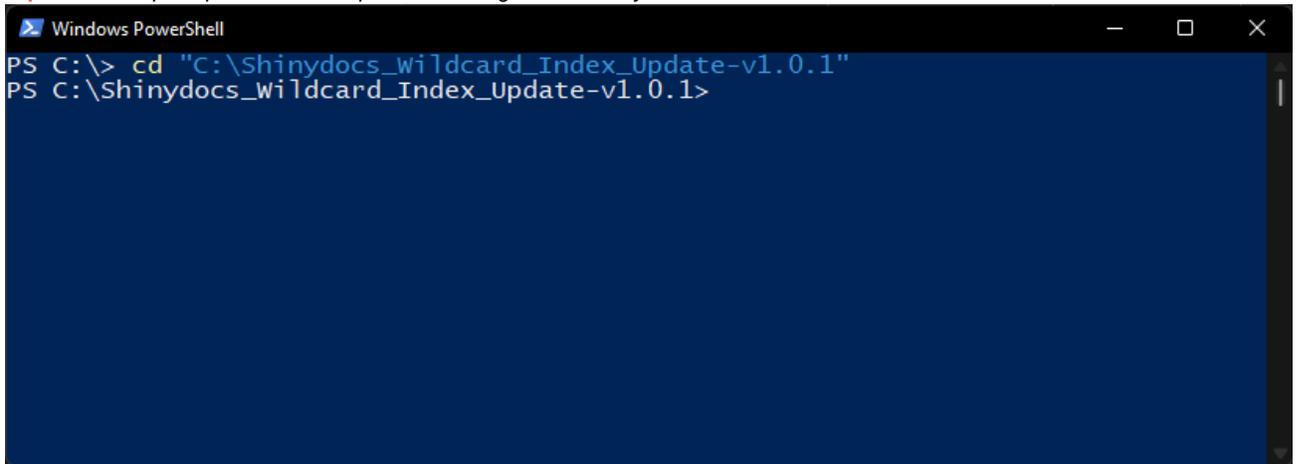
**Script Parameter Requirements**

-indexName	The name of your index to apply the update to
-indexUrl	URL of the index endpoint (example: <a href="http://localhost:9200">http://localhost:9200</a> )
-Fields	Comma-separated values of the field names to apply the enhanced wildcard functionality to (ex. -Fields name,parent)

## Step-by-step

1. Download Shinydocs\_Wildcard\_Index\_Update-v1.x.zip
2. Right-click the downloaded zipper > properties > check 'unlock', click 'OK'
3. Open PowerShell (Windows + R > "PowerShell")
4. Use `cd` to change the current directory in PowerShell to the extracted zip folder from Step 2

**Important:** Wrap the path in "double quotes" to change the directory in PowerShell



```
Windows PowerShell
PS C:\> cd "C:\Shinydocs_Wildcard_Index_Update-v1.0.1"
PS C:\Shinydocs_Wildcard_Index_Update-v1.0.1>
```

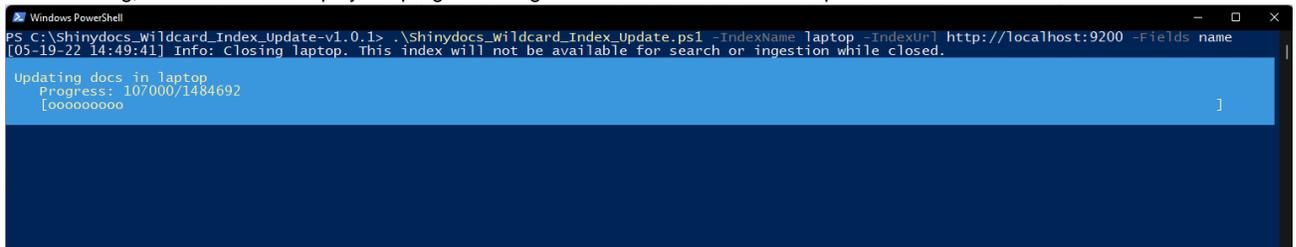
5. Run the following command, substituting your values for the parameters

```
.\Shinydocs_Wildcard_Index_Update.ps1 -IndexName [index_name] -
IndexUrl [http://server:port] -Fields [field1, field2]
```

### Example

```
.\Shinydocs_Wildcard_Index_Update.ps1 -IndexName shinyIndex -
IndexUrl http://localhost:9200 -Fields name
```

6. While running, the console will display the progress along with information about the step it is on



```
Windows PowerShell
PS C:\Shinydocs_Wildcard_Index_Update-v1.0.1> .\Shinydocs_Wildcard_Index_Update.ps1 -IndexName laptop -IndexUrl http://localhost:9200 -Fields name
[05-19-22 14:49:41] Info: Closing laptop. This index will not be available for search or ingestion while closed.

Updating docs in laptop
Progress: 107000/1484692
[oooooooooo ]
```

7. Upon completion, an info message will be displayed (as well as logged to the log file) indicating completion and the number of documents that were updated

```
Index update complete. Updated: 1484692 doc(s)
```

**Note:** The number of docs updated should match with how many items are in your index, if it does not, you can run the script again.

## Validate

To validate that the settings have been applied:

1. Open the Visualizer (ex. <http://server:5601>)
2. Go to Dev Tools
3. Run the following command:

```
GET <index_name>/_mapping
```

4. Click in the response panel, use CTRL+F and enter the name of one of the fields you applied the enhancement to (ex. `name`)

#### Before

```
"name" : {
  "type" : "text",
  "fields" : {
    "keyword" : {
      "type" : "keyword",
      "ignore_above" : 256
    }
  },
  "analyzer" : "sd_name_analyzer"
}
```

#### After

```
"name" : {
  "type" : "text",
  "fields" : {
    "keyword" : {
      "type" : "keyword",
      "ignore_above" : 256
    },
    "name_enhanced_wildcard" : {
      "type" : "text",
      "analyzer" : "sd_wildcard_analyzer"
    }
  },
  "copy_to" : [
    "name_enhanced_wildcard"
  ],
  "analyzer" : "sd_name_analyzer"
},
"name_enhanced_wildcard" : {
  "type" : "text",
  "fields" : {
    "keyword" : {
      "type" : "keyword",
      "ignore_above" : 256
    }
  }
}
```

5. You should now see the additions shown in the **After** sample (including the additional `name_enhanced_wildcard`)

## Logging

`Shinydocs_Wildcard_Index_Update.ps1` will output logs in a folder called "Index Update Logs" in the directory the script was run or called in. If the PowerShell console is in the directory `C:\WINDOWS\system32` (default directory when starting PowerShell), then the log will theoretically be in `C:\WINDOWS\system32` (should permissions allow writing to that directory)

## Previous Versions

Nothing here, yet