* **to write a search query in Splunk you need to create index first**
* **Use Splunk Web to create an index**
* If you like using a UI, use Splunk Web to create an index.
* In Splunk Web, go to **Settings > Indexes**.
* On the Indexes page, click **New Index**.
* On the New Index page, in the **Index Name** field, enter **devtutorial**. Keep the other default settings.
* Click **Save**.

### After creating index, go to add data and upload data

* how to add data in Splunk
* Adding data to Splunk can be done in several ways, depending on your needs and the type of data you want to index. Here are the main methods:
* **Upload Data**
* This method is suitable for adding a single file or a small number of files. Here’s how to do it:
* **Log into Splunk Web** and go to the Home page.
* **Click on "Add Data"** under the Settings tab.
* **Choose "Upload"** and select the file or archive from your computer.
* **Set the source type** and index for the data.
* **Submit** the data for indexing.

### After upload data go to search bar and search query

### Basic search commands in search bar:

splunk search query commands

Splunk's Search Processing Language (SPL) provides a rich set of commands to help you analyze and visualize your data effectively. Here are some of the most commonly used search query commands:

**Basic Search Commands**

1. **index**:
   * **Command**: index=<index\_name>
   * **Description**: Specifies the index to search within.
   * **Example**: index=main
2. **source**:
   * **Command**: source=<source\_path>
   * **Description**: Filters events from a specific source.
   * **Example**: source="/var/log/syslog"
3. **sourcetype**:
   * **Command**: sourcetype=<sourcetype\_name>
   * **Description**: Filters events of a specific source type.
   * **Example**: sourcetype=access\_combined
4. **host**:
   * **Command**: host=<hostname>
   * **Description**: Filters events from a specific host.
   * **Example**: host=server1

**Filtering and Data Transformation Commands**

1. **search**:
   * **Command**: search <criteria>
   * **Description**: Filters events based on specified criteria.
   * **Example**: search error OR warning
2. **where**:
   * **Command**: where <condition>
   * **Description**: Filters events based on a conditional expression.
   * **Example**: where response\_time > 500
3. **stats**:
   * **Command**: stats <stat\_function> by <field>
   * **Description**: Performs statistical calculations on the search results.
   * **Example**: stats count by status
4. **eval**:
   * **Command**: eval <new\_field>=<expression>
   * **Description**: Calculates an expression and assigns it to a new field.
   * **Example**: eval response\_time\_ms=duration\*1000
5. **table**:
   * **Command**: table <fields>
   * **Description**: Displays specified fields in a table format.
   * **Example**: table host, sourcetype, status
6. **dedup**:
   * **Command**: dedup <field>
   * **Description**: Removes duplicate events based on a specified field.
   * **Example**: dedup host
7. **rename**:
   * **Command**: rename <old\_field> as <new\_field>
   * **Description**: Renames a specified field.
   * **Example**: rename \_raw as raw\_data

**Time-Based Commands**

1. **timechart**:
   * **Command**: timechart <stat\_function> by <field>
   * **Description**: Creates a time series chart.
   * **Example**: timechart count by status
2. **bucket**:
   * **Command**: bucket <field> span=<time\_interval>
   * **Description**: Groups events into buckets of a specified time interval.
   * **Example**: bucket \_time span=1h

**Advanced Commands**

1. **join**:
   * **Command**: join <field>
   * **Description**: Joins results from a subsearch to the main search based on a common field.
   * **Example**: search index=main | join user\_id [search index=other user\_data]
2. **lookup**:
   * **Command**: lookup <lookup\_name> <input\_field> OUTPUT <output\_field>
   * **Description**: Enriches search results with data from a lookup table.
   * **Example**: lookup user\_lookup user\_id OUTPUT user\_name, user\_email
3. **transaction**:
   * **Command**: transaction <fields>
   * **Description**: Groups related events into transactions.
   * **Example**: transaction session\_id