Here's a table of 30 common grep regular expressions along with descriptions and examples:

| **Regular Expression** | **Description** | **Example** |
| --- | --- | --- |
| ^pattern | Matches any line that starts with pattern | grep "^Hello" file.txt |
| pattern$ | Matches any line that ends with pattern | grep "world$" file.txt |
| ^$ | Matches empty lines | grep "^$" file.txt |
| . | Matches any single character except newline | grep "h.t" file.txt |
| \* | Matches zero or more occurrences of the previous character | grep "ho\*t" file.txt |
| .\* | Matches any number of any characters (including none) | grep "ho.\*t" file.txt |
| ^.\*$ | Matches entire lines | grep "^.\*$" file.txt |
| [abc] | Matches any one of the characters a, b, or c | grep "[aeiou]" file.txt |
| [^abc] | Matches any character except a, b, or c | grep "[^0-9]" file.txt |
| [a-z] | Matches any lowercase letter | grep "[a-z]" file.txt |
| [A-Z] | Matches any uppercase letter | grep "[A-Z]" file.txt |
| [0-9] | Matches any digit | grep "[0-9]" file.txt |
| \d | Matches any digit (equivalent to [0-9]) | grep "\d" file.txt |
| \D | Matches any non-digit | grep "\D" file.txt |
| \w | Matches any word character (alphanumeric + underscore) | grep "\w" file.txt |
| \W | Matches any non-word character | grep "\W" file.txt |
| \s | Matches any whitespace character | grep "\s" file.txt |
| \S | Matches any non-whitespace character | grep "\S" file.txt |
| \b | Matches a word boundary | grep "\bword\b" file.txt |
| \B | Matches a non-word boundary | grep "\Bword\B" file.txt |
| (pattern) | Groups patterns | grep "\(hello|world\)" file.txt |
| `pattern1 | pattern2` | Matches pattern1 or pattern2 |
| \? | Matches zero or one occurrence of the previous character | grep "colou\?r" file.txt |
| \+ | Matches one or more occurrences of the previous character | grep "go\+d" file.txt |
| {n} | Matches exactly n occurrences of the previous character | grep "a{3}" file.txt |
| {n,} | Matches n or more occurrences of the previous character | grep "a{3,}" file.txt |
| {n,m} | Matches between n and m occurrences of the previous character | grep "a{3,5}" file.txt |
| (?i) | Case-insensitive match | grep -i "pattern" file.txt |
| \ | Escapes a special character | grep "\." file.txt |
| (?=pattern) | Positive lookahead | grep -P "(?=foo).\*" file.txt |
| (?<=pattern) | Positive lookbehind | grep -P "(?<=foo).\*" file.txt |

**References**

* GNU grep Documentation
* [Regular Expressions 101](https://www.regular-expressions.info/)
* Linux grep Command
* Regular Expression Examples

These examples should help you effectively use grep with regular expressions to search through text files and data.

**Advanced Regular Expressions**

| **Regular Expression** | **Description** | **Example** |
| --- | --- | --- |
| (?<=pattern1)pattern2 | Positive lookbehind: Matches pattern2 only if preceded by pattern1. | grep -P "(?<=hello )world" file.txt |
| (?<!pattern1)pattern2 | Negative lookbehind: Matches pattern2 only if not preceded by pattern1. | grep -P "(?<!hello )world" file.txt |
| (?=pattern1)pattern2 | Positive lookahead: Matches pattern2 only if followed by pattern1. | grep -P "foo(?=bar)" file.txt |
| (?!pattern1)pattern2 | Negative lookahead: Matches pattern2 only if not followed by pattern1. | grep -P "foo(?!bar)" file.txt |
| \bpattern\b | Word boundary: Matches pattern only as a whole word. | grep "\bhello\b" file.txt |
| \Bpattern\B | Non-word boundary: Matches pattern only if not at a word boundary. | grep "\Bhello\B" file.txt |
| `(pattern1 | pattern2 | ...)` |
| pattern{n,} | Matches pattern at least n times. | grep "a{2,}" file.txt |
| pattern{,m} | Matches pattern at most m times. | grep "a{,3}" file.txt |
| [[:alnum:]] | Alphanumeric character: Matches any letter or digit. | grep "[[:alnum:]]" file.txt |
| [[:alpha:]] | Alphabetic character: Matches any letter. | grep "[[:alpha:]]" file.txt |
| [[:digit:]] | Digit: Matches any digit. | grep "[[:digit:]]" file.txt |
| [[:space:]] | Whitespace: Matches any whitespace character. | grep "[[:space:]]" file.txt |
| [[:upper:]] | Uppercase letter: Matches any uppercase letter. | grep "[[:upper:]]" file.txt |
| [[:lower:]] | Lowercase letter: Matches any lowercase letter. | grep "[[:lower:]]" file.txt |
| [[:punct:]] | Punctuation: Matches any punctuation character. | grep "[[:punct:]]" file.txt |
| \Apattern | Matches pattern at the start of the string. | grep "\Ahello" file.txt |
| pattern\Z | Matches pattern at the end of the string. | grep "world\Z" file.txt |
| \K | Resets the start of the match: Discards everything matched so far in the current match. | grep -P "foo\Kbar" file.txt |
| (?<name>pattern) | Named groups: Captures the matched pattern as a named group. | grep -P "(?<word>\w+)" file.txt |
| \g<name> | References a named group captured previously. | grep -P "(?<word>\w+)\s+\g<word>" file.txt |
| \Qpattern\E | Quotes: Treats pattern as a literal string, ignoring any special characters within it. | grep -P "\Q[a-z]\E" file.txt |
| pattern{n,m}? | Non-greedy quantifier: Matches between n and moccurrences of pattern, as few as possible. | grep -P "a{1,3}?" file.txt |

**Explanation of Some Advanced Concepts**

* **Lookahead and Lookbehind**: These assertions match a pattern only if it is followed (lookahead) or preceded (lookbehind) by another pattern. Positive lookahead ensures the presence of a pattern, while negative lookahead ensures its absence. Similarly, positive and negative lookbehinds work with preceding patterns.
* **Word Boundaries (\b)**: These are used to match whole words. A word boundary asserts the position where a word character is not followed or preceded by another word character.
* **Character Classes ([:...:])**: These are shorthand for common character sets. For instance, [[:digit:]]matches any digit, and [[:alpha:]] matches any letter.

These advanced regular expressions enhance the capability to search and manipulate text with precision, enabling complex text processing tasks.

**References**

* [Regular Expressions 101](https://www.regular-expressions.info/)
* GNU Grep Manual
* Grep Command in Linux/Unix

| **Regular Expression** | **Description** | **Example** |
| --- | --- | --- |
| ^pattern | Matches any line that starts with pattern | grep "^Hello" file.txt |
| pattern$ | Matches any line that ends with pattern | grep "world$" file.txt |
| . | Matches any single character except newline | grep "h.t" file.txt |
| \* | Matches zero or more occurrences of the previous character | grep "ho\*t" file.txt |
| .\* | Matches any number of any characters (including none) | grep "ho.\*t" file.txt |
| ^.\*$ | Matches entire lines | grep "^.\*$" file.txt |
| [abc] | Matches any one of the characters a, b, or c | grep "[aeiou]" file.txt |
| [^abc] | Matches any character except a, b, or c | grep "[^0-9]" file.txt |
| [a-z] | Matches any lowercase letter | grep "[a-z]" file.txt |
| [A-Z] | Matches any uppercase letter | grep "[A-Z]" file.txt |
| [0-9] | Matches any digit | grep "[0-9]" file.txt |
| \d | Matches any digit (equivalent to [0-9]) | grep "\d" file.txt |
| \D | Matches any non-digit | grep "\D" file.txt |
| \w | Matches any word character (alphanumeric + underscore) | grep "\w" file.txt |
| \W | Matches any non-word character | grep "\W" file.txt |
| \s | Matches any whitespace character | grep "\s" file.txt |
| \S | Matches any non-whitespace character | grep "\S" file.txt |
| \b | Matches a word boundary | grep "\bword\b" file.txt |
| \B | Matches a non-word boundary | grep "\Bword\B" file.txt |
| (pattern) | Groups patterns | grep "\(hello|world\)" file.txt |
| `pattern1 | pattern2` | Matches pattern1 or pattern2 |
| \? | Matches zero or one occurrence of the previous character | grep "colou\?r" file.txt |
| \+ | Matches one or more occurrences of the previous character | grep "go\+d" file.txt |
| {n} | Matches exactly n occurrences of the previous character | grep "a{3}" file.txt |
| {n,} | Matches n or more occurrences of the previous character | grep "a{3,}" file.txt |
| {n,m} | Matches between n and m occurrences of the previous character | grep "a{3,5}" file.txt |
| (?i) | Case-insensitive match | grep -i "pattern" file.txt |
| \ | Escapes a special character | grep "\." file.txt |
| (?=pattern) | Positive lookahead | grep -P "(?=foo).\*" file.txt |
| (?<=pattern) | Positive lookbehind | grep -P "(?<=foo).\*" file.txt |