Click to verify



Regex test online

CategoriesFree 219 awesome toolsTest, validate, and debug regular expressions with real-time matching and detailed results. HomeToolsFree Online Regex TesterUnderstanding regex flags and their usageMatches all occurrences of the pattern in a string, not just the first onePattern: catText: cat and catWith flag: Matches both 'cat'sWithout flag: Only matches first 'cat'Makes the pattern: ^startText: start startWith flag: Matches all variations regardless of caseWithout flag: Only matches exact 'hello' and \$ match the start/end of each line, not just the whole stringPattern: ^startText: start startWith flag: Matches 'start' at beginning of each lineWithout flag: Only matches exact 'hello' and \$ match the start/end of each line, not just the whole stringPattern: ^startText: start startWith flag: Matches 'start' at beginning of each lineWithout flag: Only matches exact 'hello' and \$ match the start/end of each line. matches 'start' at beginning of stringDot (.) matches newline characters Pattern: .*Text: line1 line2With flag: Matches up to newline without flag: Matches up to newline onlyEnables Unicode emojiWithout flag: No Unicode supportMatches only from the last index positionPattern: \w+Text: first secondWith flag: Matches only from lastIndexWithout flag: Matches only flag: Matches onl tab, newline\WNot a word characterExample: ab+ matches 'ab', 'abb', but not 'a'?0 or 1 timeExample: ab? matches 'a' or 'ab' {n} Exactly n timesExample: a{3} matches 'aaa' {n,} n or more timesExample: a{2,4} matches 'aaa', 'aaa', 'etc.{n,m} Between n and m timesExample: \bcat\b matches 'hello world'\bWord boundaryExample: \bcat\b matches 'cat' but not 'category'\BNot a word boundaryExample: \Bcat\B matches 'category' but not 'cat'(abc)Capturing groupExample: (\w+)@(\w+).w+) captures username and domain(?:abc)Non-capturing groupExample: (?:b) matches 'category' but not 'cat'(abc)Capturing groupExample: (\w+)@(\w+).w+) lookaheadExample: a(?!b) matches 'a' only if not followed by 'b'^[\w.-]+@([\w-]+\.)+[\w-]{2,}\$Basic email validation\+?\d{1,4}?[-.\s]?\d{1,4} {8,}\$At least 8 characters, upper, lower, number, specialOur Regex Tester tool provides comprehensive features for testing and debugging regular expressions. Test your regex pattern matching behavior. View detailed information about matches, including indices and captured groups. Copy or download match results for further analysis or documentation. Navigate through multiple matches easily with next/previous controls. Instant feedback on regex syntax errors to help debug patterns. Enter your regular expression pattern in the pattern field and select appropriate flags (g, i, m, etc.). Type or paste your test text in the input area. Matches will be highlighted in real-time. View match details, navigate through matches, and use copy/download options for results. What is a Regular Expression (Regex)? How do I use capture groups in regex? How can I test if my regex is performant? What are common regex patterns? / Sign in to persist favorites & patterns. Click the help icon above for info. Any unsaved changes will be lost. Saved patterns & favorites will be migrated to your account. Submitted by anonymous - 8 years ago Page 2A single character in the range: a-zA charac or A-ZAlternate - match either a or bAny non-whitespace character of: a, b or cA character except: a, b or cA character in the range: a-zA character in the range: a-zA character for except: a, b or cA character in the range: a-zA character in the r character in the range: a-zA character not in the range: a-zA character in the range: a-zA character in the range: a-zA character in the range: a-z or A-ZAlternate - match either a or bAny non-whitespace character {{getLabel()}}} {{getDetail()}}} {{getDetail()}}} {{getLabel()}}} {{getLabel()}} {{getLabel()}}} {{getLabel()}} {{ as you type. Roll over a match or expression for details. Save & share expressions with others. Explore the Library for help & examples. Undo & Redo with {{getCtrlKey()}}-Z/Y. Search for & rate Community patterns. Character except newline \w \d \s word, digit, whitespace \W \D \S not word, digit, whitespace [abc] any of a, b, or c [^abc] not a, b, or c [a-g] character between a & g Anchors ^abc\$ start / end of the string \b word boundary Escaped characters \t \r tab, linefeed, carriage return \u00A9 unicode escaped © Groups & Lookaround (abc) capture group \1 backreference to group #1 (?:abc) non-capturing group (?=abc) positive lookahead (?!abc) negative lookahead Quantifiers & Alternation a* a+ a? 0 or more, 1 or more, 0 or 1 a {5} a {2,} exactly five, two or more a {1,3} between one & three a+? a {2,}? match as few as possible ab|cd match ab or cd Do you struggle to verify your regular expressions quickly? Are you tired of debugging regex patterns manually? Would you like a tool that instantly tests and validates your regex for accurate results? If you answered yes, our Regex Tester Tool is an online utility that allows you to test, validate, and debug regular expressions effortlessly. It converts your regex patterns and sample text into immediate visual feedback, showing you exactly where matches occur. This tool is an essential Regex Tester Tool for development process. Using our Regex Tester Tool is straightforward. First, paste your regular expression into the designated input field and provide the sample text you wish to test against. Next, select any optional flags (like case-insensitive or global matching) as needed. Then, click the "Test Regex" button, and within seconds, the tool processes your regex and highlights all matching segments in your sample text. This immediate visual feedback helps you fine-tune your patterns for accuracy. Our team of developers at Flexitools continually updates this tool to ensure it remains efficient and reliable. For more details on regex syntax and best practices, you can check out resources like MDN Web Docs on Regular Expressions and W3C Web Standards. Instant Testing: Quickly evaluates your regex pattern against provided sample text and highlights matched text, and errors if any exist. Customizable Flags: Offers options to set flags such as case-insensitive, global, or multiline to simulate different regex behaviors. User-Friendly Interface: Designed with simplicity in mind, making it accessible for both beginners and experts. Secure Processing: Utilizes 256-bit SSL encryption to protect your inputs, and an auto-delete policy ensures your data is removed immediately after testing. (We store only your email and name if you subscribe for notifications.) Mobile Responsive: Fully optimized for use on any device, whether you're at home or on the go. This Regex Tester Tool is invaluable for anyone who works with regular expressions. Web developers and software engineers use it to verify that their regex patterns work correctly before implementation. Digital marketers and content creators also benefit from testing regex-based filters or search queries. By ensuring your regular expressions are accurate, you can avoid bugs in your code and improve overall application performance. Additionally, our tool integrates naturally with other resources available in our Technical Tools category. Using our Regex Tester Tool offers several benefits compared to manual regex testing or using outdated desktop software: Efficiency: Automated regex testing saves significant time compared to manual debugging. Accuracy: Provides consistent, real-time feedback on your regex patterns, reducing errors. Cost-Free: Enjoy a completely free service with no hidden fees or subscription requirements. User-Friendly: Its intuitive interface eliminates the steep learning curve associated with manual regex testing. Key Benefits: Instant results and visual feedback. Customizable regex flags for accurate simulation. Secure and mobile-friendly design. Security Features: 256-bit SSL encryption protects your data. Auto-delete policy ensures no residual data remains. Only minimal user data (email and name) is stored upon subscription. Advantages Over Manual Testing: Automated analysis provides precise results. Reduces human error and saves time. Easily adaptable for various regex patterns and use cases. FeatureRegex Tester ToolManual Testing/Offline SoftwareSpeedInstant automated testingTime-consuming and prone to errorAccuracyConsistent and preciseOften inconsistent and error-proneEase of UseIntuitive, simple interfaceRequires complex setup and manual checksSecurity256-bit SSL encryption and auto-deletionOften lacks robust security measuresCostCompletely freeMay require expensive tools Our Regex Tester Tool empowers you to debug and validate your regular expressions effortlessly, ensuring that your code runs smoothly and accurately every time. Our tool uses 256-bit SSL encryption to protect your regex patterns and sample text during processing, and an auto-delete policy ensures your data is removed immediately after testing. We store only your email and name if you subscribe for notifications. You can test a wide range of regex patterns, including those for validating emails, URLs, phone numbers, and more. The tool supports all standard regex flags to simulate various matching scenarios. Yes, the Regex Tester Tool is fully mobile responsive, so you can debug your regex patterns on any device, whether it's a smartphone, tablet, or desktop. No, our tool is completely free to use with no hidden fees or subscription requirements. Unlike manual testing, which is time-consuming and prone to errors, our tool provides instant, automated, and accurate results, ensuring that your regex patterns are optimized quickly and reliably. Our Regex Tester tool allows you to quickly test and validate your regular expressions. Follow these steps to use the tool: This tool is ideal for developers, data analysts, and anyone working with text processing who needs to quickly test and refine their regular expressions. Regular expressions are powerful tools for pattern matching and text manipulation. Here are some common regex patterns and their uses: Validate Email Addresses [A-Za-z0-9. %+-]+(A-Za-z)(?=.*A-Zanumber.Match Phone Number Formats ^(\+\d{1,2}\s)?\([da-z\.-]+)\.([a-z\.]{2,6})([\w\.-]*)*\/?\$Use this pattern to validate URLs, including those with or without protocol. Use our Regex Tester to experiment with these patterns and create your own custom regular expressions for your specific needs. no description availableSubmitted by garsax - 9 years ago no description availableSubmitted by garsax - 9 years ago Page 2relative to absoluteSubmitted by garsax - 9 years ago page 2relative to absoluteSubmitted by garsax - 9 years ago no description availableSubmitted by garsax - 9 years ago page 2relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 3Page 4relative to absoluteSubmitted by garsax - 9 years ago page 4relative to absoluteSubmitted by garsax - 9 years ago page 4relative to absoluteSubmitted by garsax - 9 years ago page 4relative to absoluteSubmitted by garsax - 9 years ago page 4relative to absoluteSubmitted by garsax - 9 years ago page 4relative to absoluteSubmitted by garsax - 9 years ago page 4relative to absoluteSubmitted by garsax - 9 years ago page 4relative to absoluteSubmitted by garsax - 9 years ago page 4relative to absolute by garsax - 9 years ago page 4relative to absoluteSubmitted by xp prg - 10 years ago no description availableSubmitted by anonymous - 9 years ago no description availableSubmitted by A.ROA - 9 years ago no description availableSubmitted by anonymous - 9 year \alphaSubmitted by anonymous - a year ago testSubmitted by Luca - 10 years ago no description availableSubmi availableSubmitted by robwa - 9 years ago Page 10Extract IDSubmitted by anonymous - 4 years ago routlikeSubmitted by anonymous - 4 years ago routlikeSubmitt strings. Submitted by anmols - 9 years ago Cloud Flare Submitted by damades - 10 years ago no description available Submitted by anonymous - 10 years ago no description available Submitted by damades - 10 years ago no description available Submitted by anonymous - 10 years ago no d description available Submitted by anonymous - 10 years ago Page 14 okok Submitted by anonymous - 4 years ago smth anonymous - 4 years ago Page 16no description availableSubmitted by Sasjaq - 11 years ago no description availableSubmitted by Frusty - 9 years ago no description availableSubmitted by Ka. - 12 years ago Page 17Match an ipv6 addressSubmitted by Frusty - 9 years ago no description availableSubmitted by ka. - 12 years ago Page 18JiraSubmitted by anonymous - 4 years ago Page 19eeSubmitted by anonymous - 4 years ago Page 18JiraSubmitted by ano 4 years ago trino kill querySubmitted by anonymous - 4 years ago Page 20hjSubmitted by anonymous - 4 years ago Page 21/b(+)\b/igSubmitted by regexe - 9 years ago Get iframe srcSubmitted by Sumeet Fefar - 9 years ago no description availableSubmitted by anonymous - 3 years ago rastSubmitted by anonymous - 3 year [aeiou]\$/Submitted by anonymous - 3 years ago Page 24PythonSubmitted by anonymous - 4 years ago Page 25exampleSubmitted by anonymous - 11 years ago no description availableSubmitted by anonymous - 3 years ago Page 26222Submitted by anonymous - 3 years ago Submitted by -3 years ago iCal parserSubmitted by Appeniel - 3 years ago for tpSubmitted by anonymous - 3 years ago for tpSubmitted by Appeniel - 3 years ago for tpSubmitted by anonymous - 3 years ago for tpSubmit ^abc\$ start / end of the string \b word boundary Escaped characters \t.* \\ escaped c a+ a? 0 or more, 1 or more, 0 or 1 a{5} a{2,} exactly five, two or more a{1,3} between one & three a+? a{2,}? match as few as possible ab|cd match ab or cd Please by clicking the link or the button in the left sidebar. You do not have to create an account, just select an existing account from one of the supported service providers. Only the bare minimum information is retrieved and saved in order to identify your account. Welcome to the regex 101 regex quiz! This is an interactive learning tool you can use to improve your understanding of regular expressions, and their different applications. The quiz is divided into a number of different tasks, which each have a short description of the problem for you to solve. Once you have solved each task, you will be presented with statistics over how well others have done in that task, and the next task will be made available to you. After you have finished a task you can continue improving upon your solution, making it shorter, and climb higher into the leader boards. Do note that these tasks do not necessarily promote best practices in real life scenarios, but rather aim to teach you the fundamentals of writing powerful regular expressions. Parsing HTML with regex is seldom a good idea, despite there being a task in the guiz to do just that. The guiz is evaluated using the PCRE2 engine, which is also available in the main editor if you would like to do some testing. The tasks will get progressively harder, and you will receive less and less hand-holding the further you progress. If you encounter any issues, quirky behavior or bugs, please open up an issue on github with all the relevant details: special thanks goes out to the following individuals for their hard work and contributions to the quiz:DrRegexAndreas ZetterlundTimvdeZarthusDavidebyzeroEnjoy! Gets all numbers and floats. By: by Ben - 11 years ago Page 2A single character of: a, b or cA character in the range: a-zA char z or A-ZAlternate - match either a or bAny non-whitespace character of: a, b or cA character in the range: a-zA character in the ran