C# - GROUPING CONSTRUCTS

Grouping constructs delineate sub-expressions of a regular expression and capture substrings of an input string. The following table lists the grouping constructs:

Grouping construct	Description	Pattern	Matches
subexpression	Captures the matched subexpression and assigns it a zero-based ordinal number.	\w\1	"ee" in "deep"
? < *name* > *subexpression*	Captures the matched subexpression into a named group.	? < *double* > \w \k< double>	"ee" in "deep"
? < *name1* − *name2* > *subexpression*	Defines a balancing group definition.	((?'*Open*'\([^\ \)] * + (?'*Close* − *Open*'\) [^\)] * +)* ?(*Open*?!)$	"(1 − 3*3 − 1)" in "3+2^(1 − 3*3 − 1)"
?:*subexpression*	Defines a noncapturing group.	Write?:*Line*?	"WriteLine" in "Console.WriteLine "
?*imnsx* − *imnsx*:*subexpression*	Applies or disables the specified options within *subexpression*.	A\d{2}?*i*:\w + \b	"A12xl", "A12XL" in "A12xl A12XL a12xl"
? = *subexpression*	Zero-width positive lookahead assertion.	\w+? = \.	"is", "ran", and "out" in "He is. The dog ran. The sun is out."
?!*subexpression*	Zero-width negative lookahead assertion.	\b?!*un*\w+\b	"sure", "used" in "unsure sure unity used"
? <= *subexpression*	Zero-width positive lookbehind assertion.	? <= 19\d{2}\b	"51", "03" in "1851 1999 1950 1905 2003"
? < !*subexpression*	Zero-width negative lookbehind assertion.	? < !19\d{2}\b	"ends", "ender" in "end sends endure lender"
? > *subexpression*	Nonbacktracking *or* " *greedy* " subexpression.	[13579] ? > *A* + *B* +	"1ABB", "3ABB", and "5AB" in "1ABB 3ABBC 5AB 5AC"