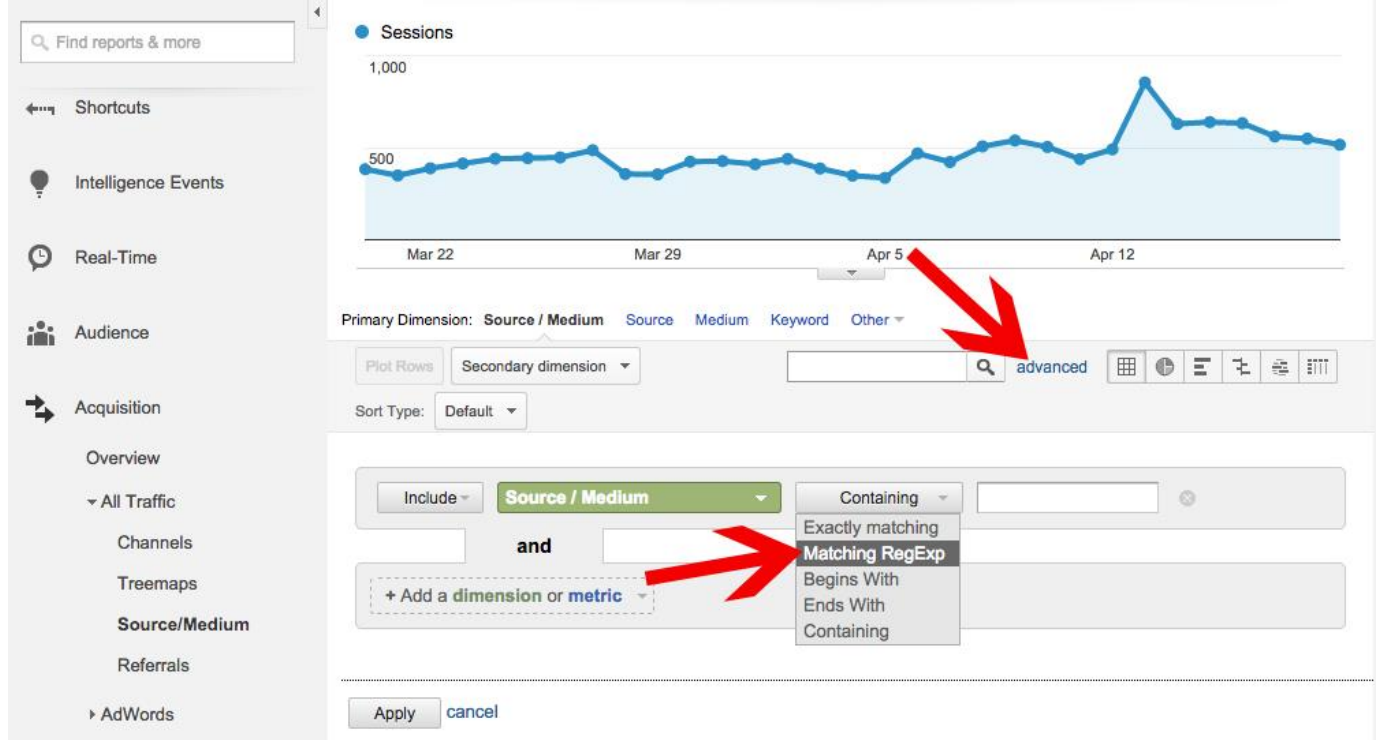


Google Analytics Regular Expression Nedir?



Google Analytics içerisinde kullanabileceğiniz Regular Expression'lar vardır. Regular Expression, Türkçe olarak; "düzenli ifadeler" demektir. Regular Expression, yazılımcıların işi gibi görünse de, dijital pazarlamayla ilgilenen herkesin temel seviyede bilmesi gereken bir konudur.

Regular Expression'lar, hemen hemen kullandığınız tüm araçlarda karşınıza çıkabilir. A/B testi araçları, ısı haritaları veya online anketler yahut dönüşüm toolları gibi çeşitli araçlar örnek verilebilir. İlk bakışta, Regular Expression; biraz karışık gibi görünebilir, fakat biraz pratik yaptığınızda; işinizin ne kadar kolaylaştırdığını görebilirsiniz.

Google Analytics Regular Expression Özellikleri

Google Analytics Regular Expression, birçok programlama dilinde ve Google araçlarında rahatlıkla kullanılabilen; sayısal ve de dizgisel ifadelerdir. İfadeleri, belirlemiş olduğunuz kalıplara göre kontrol edebilmeyi sağlamaktadır. **Google Analytics Regular Expression** kullanılırken, edinilen kullanım deneyimi; hayatımızı pek çok yönden kolaylaştırmaktadır.

Google Analytics Regular Expression Ne Gibi Kolaylıklar Sağlar ve Avantajları Nelerdir?

Google Analytics'de Regular Expression kullanmanın belli başlı avantajları şunlardır:

- Google Analytics Regular Expression ile, birden fazla hedef tamamlama sayfası ayarlayabilirsiniz.
- Google Analytics Regular Expression araçlarını, IP aralığı filtrelemek için de kullanabilirsiniz.
- Google Analytics Regular Expression ile; Google Analytics arayüzünde, çok daha kapsamlı filtrelemeler yapabilirsiniz.
- Google Analytics Regular Expression ile, karmaşık hunileri daha kolay bir şekilde yapılandırabilirsiniz.

Birçok avantajı ve çok yönlü kullanım özellikleriyle, Google Analytics Regular Expression kullanımı her geçen gün daha da yaygınlaşmaktadır

Regular Expression Nasıl Kullanılır?

*	Yıldız Kendinden bir önceki ifadeyi yok sayar veya tekrarlar	-	Tire Aralık belirtmek için kullanılır		Düz Çizgi Veya karakteridir
\	Ters Slash Kaçış karakteridir	\$	Dolar İşareti Satır sonunu ifade eder	+	Artı Kendinden önceki ifadenin bir veya daha fazla kullanılmasını sağlar
^	Düzeltilme İşareti Satır başını ifade eder	.	Nokta Tek bir karakteri ifade eder	?	Soru İşareti Kendinden önceki karakterin stringte olup olmasını sağlar
{ }	Süslü Parantez Önceki karakterin ne kadar tekrar edeceğini gösterir	()	Parantez Gruplamaları gösterir	[]	Köşeli Parantez Belirli karakterleri veya aralığı ifade eder

Düz çizgi İşareti (|)

Kısaca veya anlamına gelmektedir.

Örnek: /haber/|/futbol/

Yukarıdaki düz çizgi sembolü uzantısı /haber/ veya /futbol/ olan sayfaları size getirecektir. Aynı zamanda bunların alt sayfaları da gelebilir /haber/turkiye şeklinde.

Örnek2: (mert|dinc)

Yukarıdaki örnekte içinde mert veya dinc geçen her kelime getirilebilir.

1. mert
2. abcmert

3. mertabc
4. mertdinc

Nokta İşareti (.)

Tek bir karakteri ifade eder. Bulunduğu yere bütün karakterler gelebilir.

Örnek: .para ifadesi aşağıdakilerle eşleşir

1. npara
2. spara
3. 1para

Yıldız İşareti (*)

Kendinden bir önceki ifadeyi yok sayar veya tekrarlanan bir biçimde kullanılmasını sağlar.

Örnek: mert* ifadesi,

mer, mert, mertt, merttt şeklinde eşleşebilir.

Artı İşareti (+)

Kendinden bir önceki ifadenin bir veya birden fazla bir şekilde eşleşmesini sağlar.

Örnek: merhaba+ ifadesi

merhaba, merhabaa, merhabaaa şeklinde eşleşebilir.

Ters Slash İşareti (\)

Kaçış karakteridir. Kendinden sonraki karakterin string'e dahil edilmesini sağlar. Meta karakterleri normal karakterlere dönüştürür.

Örnek: burhanmertdinc\.com ifadesinde nokta işaretinin önüne ters slash eklenmesiyle birlikte noktanın işlevi yok sayılmış ve string değerine dönüştürülmüştür.

Örnek2: /thankyou\?1234 ifadesinde başka bir meta karakter olan ? işaretinin işlevini yitirmesi için kaçış karakteri kullanmalıyız.

Süslü parantez ({ })

Önceki karakterin bir veya daha fazla tekrar etmesiyle eşleşebilir.

Örnek:

1. m{2} ifadesi mm ile eşleşir
2. m{1-3} ifadesi m, mm, mmm ile eşleşebilir.
3. [a-z]{2} ifadesi, aa, bb, cc gibi tekrar eden harflerle eşleşebilir.

Tire İşareti (-)

Aralık belirtmek, daha geniş listeler oluşturmak için kullanılır.

Örnek:

[A-Z] A'dan Z'ye kadar tüm harfler ile eşleşir.

[0-9] 0 ile 9 arasındaki tüm sayılarla eşleşir

[a-z] - a'dan z'ye kadar tüm harfler ile eşleşir

Örnek:

apple[1-6] ifadesi, apple1, apple2, apple3, apple4, apple5, apple6 ile eşleşir.

Köşeli Parantez İşareti ([])

Belirtilmiş karakterlerden veya belirtilen aralık içindeki karakterlerden birisi anlamına gelir.

Örnek: r[aeo]r ifadesi, rar, rer, ror şeklinde getirilebilir.

Biraz daha fazla aralık vermek için "-" kullanabilirsiniz.

** Bunların tersini uygulamak için ise köşeli parantezi düzeltme işareti ile kullanmalısınız. Belirtmiş olduğunuz karakterlerin dışında kalan karakterler getirilir.

Örnek3: [^a] ifadesi,

1. /aksesuar/apple/ ifadesi ile eşleşmez.
2. /aksesuar/samsung/ ifadesiyle eşleşir.

Örnek4: kampanya=[^Ysk] ifadesi,

1. ?kampanya=Yaz ile eşleşmez
2. ?kampanya=sonbahar ile eşleşmez
3. ?kampanya=ilkbahar ile eşleşir.

Parantez İşareti ()

Gruplama yapmanızı sağlar. Örnek:

1. /aksesuar/iphone5/kilif/
2. /aksesuar/s3/kilif/

3. /aksesuar/iphone6/kilif

^/aksesuar/(iphone5|s3|iphone6)/kilif/\$

Soru İşareti (?)

Kendinden önceki karakterin stringde olmasını ya da olmamasını sağlar. Genelde yazım hatalarını görebilmek için eklenir.

Örnek: merr?t ifadesi,

mert veya mertr olarak eşleşir.

Örnek2: ^bura?da\$ ifadesi,

burada veya burda olarak eşleşir.

Dolar İşareti (\$)

Satır sonunu ifade eder. Şununla biten anlamına gelir.

kravat\$ ifadesi kırmızı kravat'ı tetikler.

Örnek2: \.php\$ ifadesi .php ile biten tüm sayfaları getirecektir.

Örnek3: kravat\.php\$ ifadesi ise kravat.php ile biten sayfaları getirir

Düzeltilme İşareti (^)

Satır başını ifade eder. Şununla başlayan anlamına gelir.

Örnek: ^kravat ifadesi

kravat, kravat al, kravatta gibi kelimeleri ile eşleşebilir.