|  |  |  |
| --- | --- | --- |
| **Yapılan Çalışmanın Konusu:** Log4j ve Log4j2 | | |
| **Log4j Nedir?**  Log4j , Log4j Apache Software Foundation ın geliştirmiş olduğu bir loglama kütüphanesidir. (logging framework). İçerdiği kayıt mekanizması sayesinde normal hata ayıklamada bulamayacağımız çeşitli faydalar sağlamaktadır.  **Log4j2**  Apache Log4j 2, önceki sürüm olan Log4j 1.x üzerinde önemli iyileştirmeler sağlayan ve Logback'in mimarisindeki bazı temel sorunları giderirken, Logback'teki birçok iyileştirmeyi sağlayan Log4j'e yapılan bir yükseltmedir.  **Log4j2 Gekiştirilmiş Özellikleri**  Log4j 2, LMAX Disruptor kütüphanesini temel alan yeni nesil Asenkron Logger'ları içerir. Çok iş parçacıklı senaryolarda, Asenkron Zamanlayıcılar, 18 kat daha yüksek verime ve büyüklük sırasına göre Log4j 1.x ve Logback'ten daha düşük gecikme sürelerine sahiptir.  Log4j 2 de bağlam olaylarına, işaretleyicilere, düzenli ifadelere ve Log olayındaki diğer bileşenlere göre filtrelemeyi destekler. Filtreleme, Logger'lara aktarılmadan önce veya Appenders'dan geçerken tüm olaylara uygulanacak şekilde belirlenebilir. Ayrıca, filtreler Logger'larla da ilişkilendirilebilir. , Log4j 2 de bağlam olaylarına, işaretleyicilere, düzenli ifadelere ve Log olayındaki diğer bileşenlere göre filtrelemeyi destekler. Filtreleme, Logger'lara aktarılmadan önce veya Appenders'dan geçerken tüm olaylara uygulanacak şekilde belirlenebilir. Ayrıca, filtreler Logger'larla da ilişkilendirilebilir.  Log4j 2'de [özel log seviyeleri](https://logging.apache.org/log4j/2.x/manual/customloglevels.html) kodda veya konfigürasyonda kolayca tanımlanabilir. Alt sınıflamaya gerek yoktur.  **Log4j2 Filtreleme**  Filtreler, logların değerlendirilip yayımlanmayacağını veya nasıl yayınlanacağını belirlemek için değerlendirilmesine izin verir. (Bir çok filter çeşidi bulunmaktadır. Projede “CompositeFilter” filtreleme kullanıldığından dolayı, compositeFilter filtreleme çeşidine değinilmiştir.)  CompositeFilter, birden fazla filtre belirtmenin bir yolunu sağlar. Konfigürasyona bir filtre elemanı olarak eklenir ve değerlendirilecek diğer filtreleri içerir.    Yukarıdaki örnekde görüldüğü gibi, filteleme yapılarak log mesajlarında sadece info türündeki mesajların gösterilmesi sağlanmıştır. | | |
| **Tarih:** 30 / 09 / 2019 | Kaşe / İmza | **Sayfa No:** 10 |

|  |  |  |
| --- | --- | --- |
| **Yapılan Çalışmanın Konusu:** Log4j ve Log4j2 | | |
| **Filtrelemede Kullanılan Parametreler**   |  |  | | --- | --- | | Parametre Adı | Açıklama | | Level | Filtrelenecek mesajların seviyesi. Varsayılan, WARN. | | onMatch | Filtre eşleştiğinde gerçekleştirilecek eylem. ACCEPT, DENY veya NEUTRAL olabilir. Varsayılan, NEUTRAL | | onMismatch | Filtre eşleşmediğinde gerçekleştirilecek eylem. ACCEPT, DENY veya NEUTRAL olabilir. Varsayılan, NEUTRAL |   **Kodda Özel Günlük Seviyeleri Tanımlama**  Log4J 2 özel günlük seviyelerini destekler. Özel günlük seviyeleri kodda veya konfigürasyonda tanımlanabilir. Kodda özel bir günlük seviyesi tanımlamak için, Level.forName () yöntemini kullanılabilir. Bu yöntem belirtilen ad için yeni bir seviye oluşturur. Bir günlük seviyesi tanımlandıktan sonra, Logger.log () yöntemini çağırarak ve özel günlük seviyesini geçerek mesajları bu seviyede kaydedebilinmektedir:    Özel bir günlük seviyesi tanımlarken, intLevel parametresi (yukarıdaki örnekte 550 ve 350), özel seviyenin Log4J 2'ye yerleşik standart seviyelere göre nerede bulunduğunu belirler. Başvuru için, aşağıdaki tabloda yerleşiklerin intLevel'i gösterilmektedir | | |
| **Tarih:** 30 / 09 / 2019 | Kaşe / İmza | **Sayfa No:** 10 |

|  |  |  |
| --- | --- | --- |
| **Yapılan Çalışmanın Konusu:** Log4j ve Log4j2 | | |
| **Konfigürasyonda Özel Günlük Seviyeleri Tanımlama**  Özel günlük seviyeleri konfigürasyonda da tanımlanabilir. Bu, bir logger filtresinde veya bir appender filtresinde özel bir seviye kullanmak için uygundur. | | |
| **Tarih:** 30 / 09 / 2019 | Kaşe / İmza | **Sayfa No:** 10 |