

**TÜRKİYE CUMHURİYETİ  
YILDIZ TEKNİK ÜNİVERSİTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



**HADOOP UYGULAMALARI**

16011128 - SEMİH DURMAZ  
16011705 - YUNUS EMRE DEMİR

**Büyük Veri İşleme ve Analizi  
Dönem Projesi**

Mayıs, 2020

Bu proje kapsamında; Booking.com'dan alınan ve kaggle.com da bulunan '515K Hotel Reviews Data in Europe' isimli veri setini kullandık. Bu veri setinde 515.000 müşteri yorumu ve Avrupa çapında 1493 lüks otel puanı bulunmaktadır. Veri seti dosyası 17 alan içerir. Bunlardan bazıları: Hotel address, Reviews and Reviewers Scores vd. gibidir. Mevcut veri seti üzerinde 5 ayrı fonksiyon uygulayacağız. Bunlar:

- 1- Total numbers of reviews (sum)
- 2- The average of scores by hotels (average)
- 3- Lowest and highest score given (min,max)
- 4- The months that most reviews and least reviews are made through the years
- 5- Standart deviation of all scores

**Veri seti linki:** <https://www.kaggle.com/jiashenliu/515k-hotel-reviews-data-in-europe>

### Java Fonksiyonlarının yazılması ve karşılaşılanlar:

5 adet fonksiyonu gerçekleştirebilmek için Mapreduce ile uyumlu bir şekilde 5 adet java kodu yazılmıştır. Bu kodları yazarken yaşanan bir zorluk, Reduce işleminde bir kez döngü yapılabilmesiydi. Reduce işleminde, map işleminde elde edilen veriler 2 kez kullanılması gerektiği zaman ilk döngü içinde ilgili veriyi bir Stringe atıp hepsini bir Array içine attık. İkinci döngüyü bu Array üzerinden döndürdük.

Mapreduce'nin IntWritable ve text veri tiplerine sahip olmasının yanısıra DoubleWritable veri tipine de sahip olduğu öğrenildi ve kullanıldı.

### Amazon üzerinde gerçekleştirilmesi:

Projeyi gerçekleştirmek için kendi bilgisayarlarımız yetersiz kaldığı için Amazon üzerinden gerçeklemeye karar verdik. Amazon Web Servisleri aracılığıyla oluşturduğumuz 4 sanal makine üzerinde dağıtık bir Hadoop sistemi kurmayı ve ardından bu sistem üzerinde yazdığımız Java sınıflarıyla yardımıyla MapReduce işlemlerini gerçekleştirmeyi amaçladık. Bu amaçlar doğrultusunda;

İlk adım olarak sanal makinelerimizi oluşturduk.

	NameNode	i-006fa196662732c2f	t2.micro	us-east-2c	running	2/2 checks ...	None		ec2-3-16-216-12.us-east-2c...
	DataNode001	i-098e23155153c546a	t2.micro	us-east-2c	running	2/2 checks ...	None		ec2-18-224-96-142.us-east-2c...
	DataNode002	i-0aa36c0654e79ab11	t2.micro	us-east-2c	running	2/2 checks ...	None		ec2-18-188-91-152.us-east-2c...
	DataNode003	i-0bd3e319f14e598a9	t2.micro	us-east-2c	running	2/2 checks ...	None		ec2-52-14-167-173.us-east-2c...

Ardından projemiz doğrultusunda kullanacağımız hadoop ve Java kurulumlarını gerçekleştirdik. Konfigürasyon dosyalarında gerekli güncellemeleri yaptık. Hadoop klasöründe bulunan core-site.xml, hadoop-env.sh, hdfs-site.xml, mapred-site.xml, yarn-site.xml dosyalarında gerekli ayarlamaları yaptık.

Sistemin hangi düğümlere erişeceğini bilmesi adına çevre değişkenlerini ekledik.

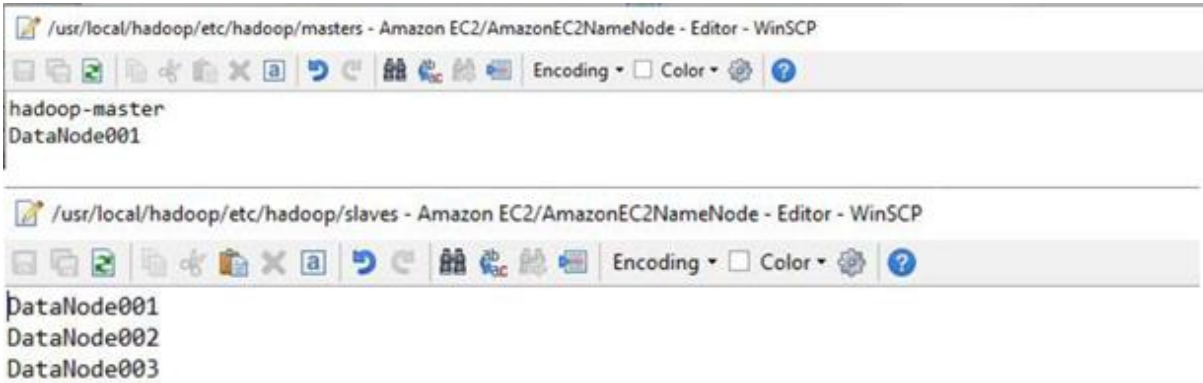
```
export NameNodeDNS="ec2-3-16-216-12.us-east-2.compute.amazonaws.com"
export DataNode001DNS="ec2-18-224-96-142.us-east-2.compute.amazonaws.com"
export DataNode002DNS="ec2-18-188-91-152.us-east-2.compute.amazonaws.com"
export DataNode003DNS="ec2-52-14-167-173.us-east-2.compute.amazonaws.com"
export NameNodeIP="172.31.32.203"
export DataNode001IP="172.31.40.226"
```

```
export DataNode002IP="172.31.40.248"
export DataNode003IP="172.31.40.215"
export IdentityFile=~/.ssh/hadoop-clusterkeypair.pem"
```

Namenode ve datanode bağlantılarının gerçekleşmesi adına SSH bağlantılarını oluşturduk.

```
Using username "ubuntu".
Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 5.3.0-1017-aws x86_64)
```

.masters ve .slaves dosyalarını oluşturduk.



```
/usr/local/hadoop/etc/hadoop/masters - Amazon EC2/AmazonEC2NameNode - Editor - WinSCP
hadoop-master
DataNode001

/usr/local/hadoop/etc/hadoop/slaves - Amazon EC2/AmazonEC2NameNode - Editor - WinSCP
DataNode001
DataNode002
DataNode003
```

Dağıtık sistemi yönetmek için kullandığımız master yani NameNode makinamızdan sistemi ayağı kaldırmadan önce formatladık. Ardından hdfs ve YARN'ı başlattık. Sistemin durumunu kontrol ettiğimizde ana makinamız aracılığıyla diğer 3 slave makinaya sorunsuz bir şekilde bağlanmış olduğunu teyit ettik.

DFS Remaining:	21.06 GB (63.41%)
Block Pool Used:	48 KB (0%)
DataNodes usages% (Min/Median/Max/stdDev):	0.00% / 0.00% / 0.00% / 0.00%
Live Nodes	3 (Decommissioned: 0)

Bir sonraki adım olarak yazdığımız fonksiyonları MapReduce fonksiyonlarını Hadoop üzerinde çalıştırmayı denedik.

```
20/05/14 12:25:07 INFO client.RMPProxy: Connecting to ResourceManager at /0.0.0.0:8032
20/05/14 12:25:08 INFO input.FileInputFormat: Total input files to process : 1
20/05/14 12:25:09 INFO mapreduce.JobSubmitter: number of splits:1
20/05/14 12:25:09 INFO Configuration.deprecation: yarn.resourcemanager.system-metrics-publisher.enabled is deprecated. Instead, use yarn.system-metrics-publisher.enabled
20/05/14 12:25:13 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1589458911973_0003
20/05/14 12:25:13 INFO impl.YarnClientImpl: Submitted application application_1589458911973_0003
20/05/14 12:25:13 INFO mapreduce.Job: The url to track the job: http://ec2-3-16-216-12.us-east-2.compute.amazonaws.com:8088/proxy/application_1589458911973_0003/
20/05/14 12:25:13 INFO mapreduce.Job: Running job: job_1589458911973_0003
```

Bu kısımdan sonra ilerleyemedik. Hadoop sisteminin kendi fonksiyonlarını da denedik fakat sistem "running job" aşamasında takılı kaldı. Bu problemi çözmek için 3 gündür girmediğimiz forum sitesi, denemediğimiz yöntem kalmadı. Çözemediğimiz için de performans testlerini ve oradan sonraki adımları gerçekleştiremedik.