



Ćwiczenie 1

Praca z HDSF przy użyciu Ambari UI.

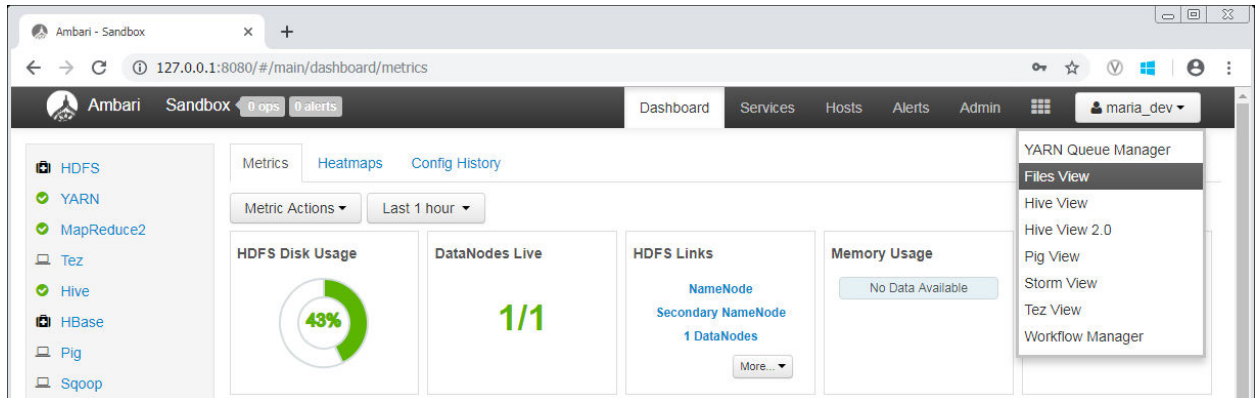
1. Ściągnij i rozpakuj plik: <https://danilewicz.blob.core.windows.net/public/1987.zip>

Plik ten ma następujące kolumny:

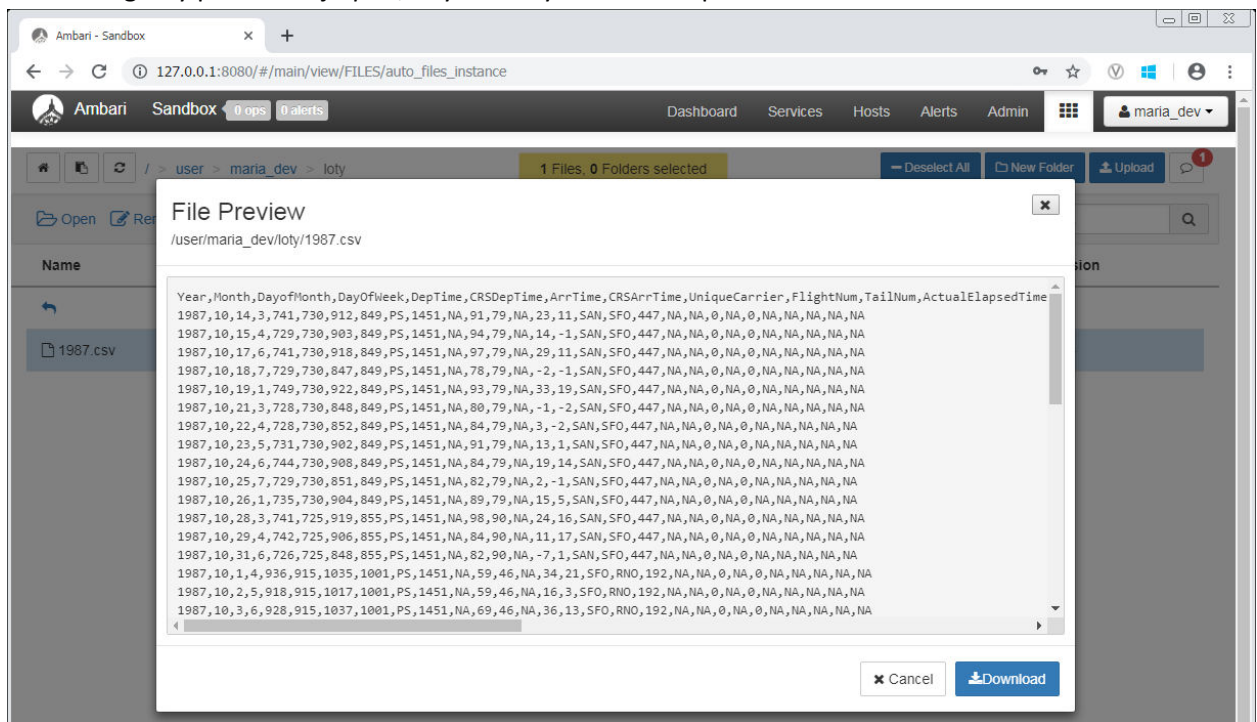
	Name	Description
1	Year	1987-2008
2	Month	01-Dec
3	DayOfMonth	Jan-31
4	DayOfWeek	1 (Monday) - 7 (Sunday)
5	DepTime	actual departure time (local, hhmm)
6	CRSDepTime	scheduled departure time (local, hhmm)
7	ArrTime	actual arrival time (local, hhmm)
8	CRSArrTime	scheduled arrival time (local, hhmm)
9	UniqueCarrier	unique carrier code
10	FlightNum	flight number
11	TailNum	plane tail number
12	ActualElapsedTime	in minutes
13	CRSElapsedTime	in minutes
14	AirTime	in minutes
15	ArrDelay	arrival delay, in minutes
16	DepDelay	departure delay, in minutes
17	Origin	origin IATA airport code
18	Dest	destination IATA airport code
19	Distance	in miles
20	TaxiIn	taxi in time, in minutes
21	TaxiOut	taxi out time in minutes
22	Cancelled	was the flight cancelled?
23	CancellationCode	reason for cancellation (A = carrier, B = weather, C = NAS, D = security)
24	Diverted	1 = yes, 0 = no
25	CarrierDelay	in minutes
26	WeatherDelay	in minutes
27	NASDelay	in minutes
28	SecurityDelay	in minutes
29	LateAircraftDelay	in minutes

2. Uruchom wirtualną maszynę z hadoop'em.
3. Otwórz link <http://127.0.0.1:8080> w przeglądarce internetowej.
4. Zaloguj się do Ambari korzystając z loginu *maria_dev* i hasła *maria_dev*.

5. W prawym górnym rogu kliknij ikonę siatki i wybierz *Files View*:



6. Wejdź do folderu: `/user/maria_dev`
7. Kliknij *New Folder* i wpisz swoje nazwisko jako nazwę folderu, zatwierdź klikając *+Add*. Na koniec wejdź do nowo stworzonego folderu.
8. Aby wgrać plik do HDFS, kliknij *Upload* i wybierz plik *1987.csv* ze swojego komputera.
9. Zaznacz wgrany plik i kliknij *Open*, aby zobaczyć zawartość pliku:

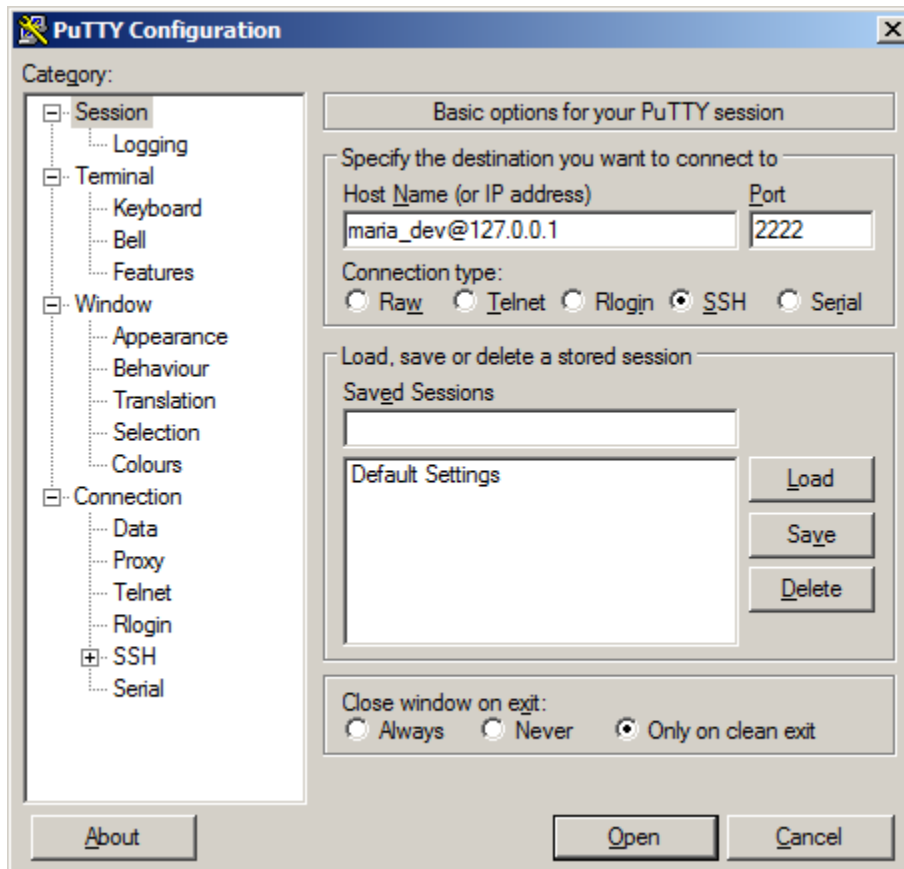


10. Wypróbuj inne opcje, a następnie usuń plik *1987.csv* i folder ze swoim nazwiskiem.

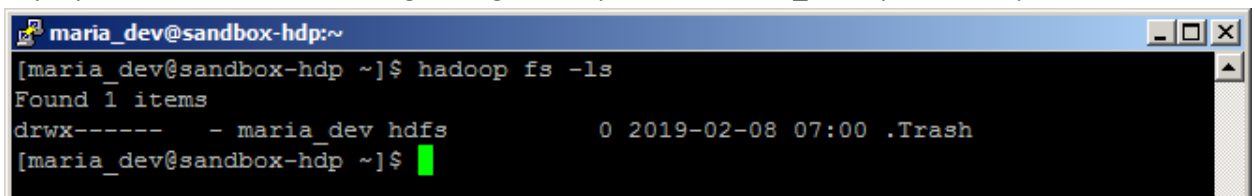
Ćwiczenie 2

Praca z HDSF przy użyciu wiersza poleceń.

1. Ściągnij program *Putty*: <https://the.earth.li/~sgtatham/putty/latest/w64/putty.exe>
2. Uruchom program *putty.exe*. W polu *Host Name* wpisz [maria_dev@127.0.0.1](#), w polu *Port* **2222** i kliknij *Open*.



3. Wpisz hasło: [maria_dev](#)
4. Aby wyświetlić zawartość folderu głównego dla użytkownika *maria_dev* wpisz: [hadoop fs -ls](#)



5. Aby utworzyć folder wpisz: [hadoop fs -mkdir twoje_nazwisko](#)

6. Powtórz krok 4, aby zobaczyć czy folder został utworzony:

```
maria_dev@sandbox-hdp:~  
[maria_dev@sandbox-hdp ~]$ hadoop fs -mkdir danilewicz  
[maria_dev@sandbox-hdp ~]$ hadoop fs -ls  
Found 2 items  
drwx----- - maria_dev hdfs          0 2019-02-08 07:00 .Trash  
drwxr-xr-x - maria_dev hdfs          0 2019-02-08 07:17 danilewicz  
[maria_dev@sandbox-hdp ~]$
```

7. Zanim wgramy nasz plik do HDFS musimy go jeszcze raz pobrać i rozpakować.
- Wpisz: `mkdir twoje_nazwisko`, aby utworzyć nowy folder w lokalnym systemie plików
 - Wpisz: `cd twoje_nazwisko`, aby wejść do nowo utworzonego folderu
 - Wpisz: `wget https://danilewicz.blob.core.windows.net/public/1987.zip`, aby pobrać plik:

```
maria_dev@sandbox-hdp:~/danilewicz  
[maria_dev@sandbox-hdp danilewicz]$ wget https://danilewicz.blob.core.windows.net/public/1987.zip  
--2019-02-08 06:48:02-- https://danilewicz.blob.core.windows.net/public/1987.zip  
Resolving danilewicz.blob.core.windows.net... 13.95.96.176  
Connecting to danilewicz.blob.core.windows.net|13.95.96.176|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 16379138 (16M) [application/zip]  
Saving to: "1987.zip"  
  
100%[=====>] 16,379,138 2.96M/s in 5.4s  
  
2019-02-08 06:48:08 (2.90 MB/s) - "1987.zip" saved [16379138/16379138]  
[maria_dev@sandbox-hdp danilewicz]$
```

- Następnie wpisz: `unzip 1987.zip`, aby rozpakować archiwum:

```
maria_dev@sandbox-hdp:~/danilewicz  
[maria_dev@sandbox-hdp danilewicz]$ unzip 1987.zip  
Archive: 1987.zip  
  inflating: 1987.csv  
[maria_dev@sandbox-hdp danilewicz]$
```

- Wpisz: `ls`, aby zobaczyć listę lokalnych plików:

```
maria_dev@sandbox-hdp:~/danilewicz  
[maria_dev@sandbox-hdp danilewicz]$ ls  
1987.csv  1987.zip  
[maria_dev@sandbox-hdp danilewicz]$
```

8. Teraz możemy wgrać nasz csv plik do folderu `twoje_nazwisko` na HDFS.
Wpisz: `hadoop fs -copyFromLocal 1987.csv twoje_nazwisko/1987.csv`

9. Sprawdźmy czy plik został skopiowany. Wpisz: `hadoop fs -ls twoje_nazwisko`

```

maria_dev@sandbox-hdp:~/danilewicz
[maria_dev@sandbox-hdp danilewicz]$ hadoop fs -ls danilewicz
Found 1 items
-rw-r--r-- 1 maria_dev hdfs 127162942 2019-02-08 06:53 danilewicz/1987.csv
[maria_dev@sandbox-hdp danilewicz]$

```

10. Aby skasować plik na HDFS wpisujemy: `hadoop fs -rm -skipTrash twoje_nazwisko/1987.csv`

```

maria_dev@sandbox-hdp:~/danilewicz
[maria_dev@sandbox-hdp danilewicz]$ hadoop fs -rm danilewicz/1987.csv
19/02/08 06:55:27 INFO fs.TrashPolicyDefault: Moved: 'hdfs://sandbox-hdp.hortonw
orks.com:8020/user/maria_dev/danilewicz/1987.csv' to trash at: hdfs://sandbox-hd
p.hortonworks.com:8020/user/maria_dev/.Trash/Current/user/maria_dev/danilewicz/1
987.csv
[maria_dev@sandbox-hdp danilewicz]$

```

Aby skasować folder na HDFS wpisujemy: `hadoop fs -rmdir twoje_nazwisko`

11. Na koniec skasuj swój folder z lokalnego systemu plików:

- Wyjdź ze swojego folderu: `cd ~`
- Skasuj folder i wszystkie pliki w środku: `rm -r -f twoje_nazwisko`

12. Wpisując `hadoop fs` otrzymamy listę dostępnych operacji. Oto kilka najważniejszych:

<code>-ls <ścieżka></code>	podaje zawartość folderu zdefiniowanego przez ścieżkę; pokazuje nazwę, rodzaj dostępu, właściciela, rozmiar i datę modyfikacji dla każdego pliku/folderu
<code>-lsr <ścieżka></code>	zachowuje się jak <code>-ls</code> , ale wyświetla także wszystkie informacje z podkatalogów
<code>-du <ścieżka></code>	pokazuje użycie dysku w bajtach dla wszystkich plików, które pasują do ścieżki
<code>-dus <ścieżka></code>	zachowuje się jak <code>-du</code> , ale wyświetla użycie dysku dla wszystkich plików/katalogów w ścieżce
<code>-mv <skąd> <dokąd></code>	przenosi plik lub folder wewnątrz HDFS
<code>-cp <skąd> <dokąd></code>	kopiuje plik lub folder wewnątrz HDFS
<code>-rm <ścieżka></code>	kasuje plik lub pusty folder
<code>-rmr <ścieżka></code>	kasuje plik lub folder i wszystkie podfoldery
<code>-put <skąd> <dokąd></code>	kopiuje plik lub folder z lokalnego systemu plików do HDFS
<code>-copyFromLocal <skąd> <dokąd></code>	identyczna do <code>-put</code>
<code>-moveFromLocal <skąd> <dokąd></code>	przenosi plik lub folder z lokalnego systemu plików do HDFS
<code>-moveToLocal <skąd> <dokąd></code>	przenosi plik lub folder z HDFS do lokalnego systemu plików
<code>-get <skąd> <dokąd></code>	kopiuje plik lub folder z HDFS do lokalnego systemu plików
<code>-copyToLocal <skąd> <dokąd></code>	identyczna do <code>-get</code>
<code>-getmerge <skąd> <dokąd></code>	pobiera wszystkie pliki pasujące do ścieżki w HDFS i kopiuje je do pojedynczego, scalonego pliku w lokalnym systemie plików
<code>-cat <nazwa pliku></code>	wyświetla zawartość pliku
<code>-mkdir <nazwa folderu></code>	tworzy nowy folder o podanej nazwie
<code>-test <ścieżka></code>	zwraca 1 jeśli ścieżka istnieje, 0 w przeciwnym wypadku
<code>-stat [format] <ścieżka></code>	pokazuje informacje o ścieżce; format to ciąg, który akceptuje



	rozmiar pliku w blokach (% b), nazwę pliku (% n), rozmiar bloku (% o), replikację (% r) i datę modyfikacji (% y,% Y).
--	---