Big Data architektúrák és elemző módszerek Gyakorlat

Gombos Gergő

Elérhetőségek

- Gyak.vez: Dr. Gombos Gergő
- honlap: http://ggombos.web.elte.hu
- email: ggombos@inf.elte.hu
- szoba: D. 2-503

Tematika

- Architektúra ismeretek, Batch feldolgozás:
  - Hadoop / MapReduce / HDFS
  - Spark

- Elemző módszerek
  - Python (pandas, sklearn, numpy)
  - Adatvizualizáció (matplotlib)

Követelmények

- Hadoop / MapReduce beadandó
- (Spark Beadandók)
- Spark ZH (min. 50%)
  - Spark Batch adatelemzés, adatfeldolgozás
- Python adatelemzés ZH (min. 50%)
  - Pandas, SKLearn, Adatvizualizáció

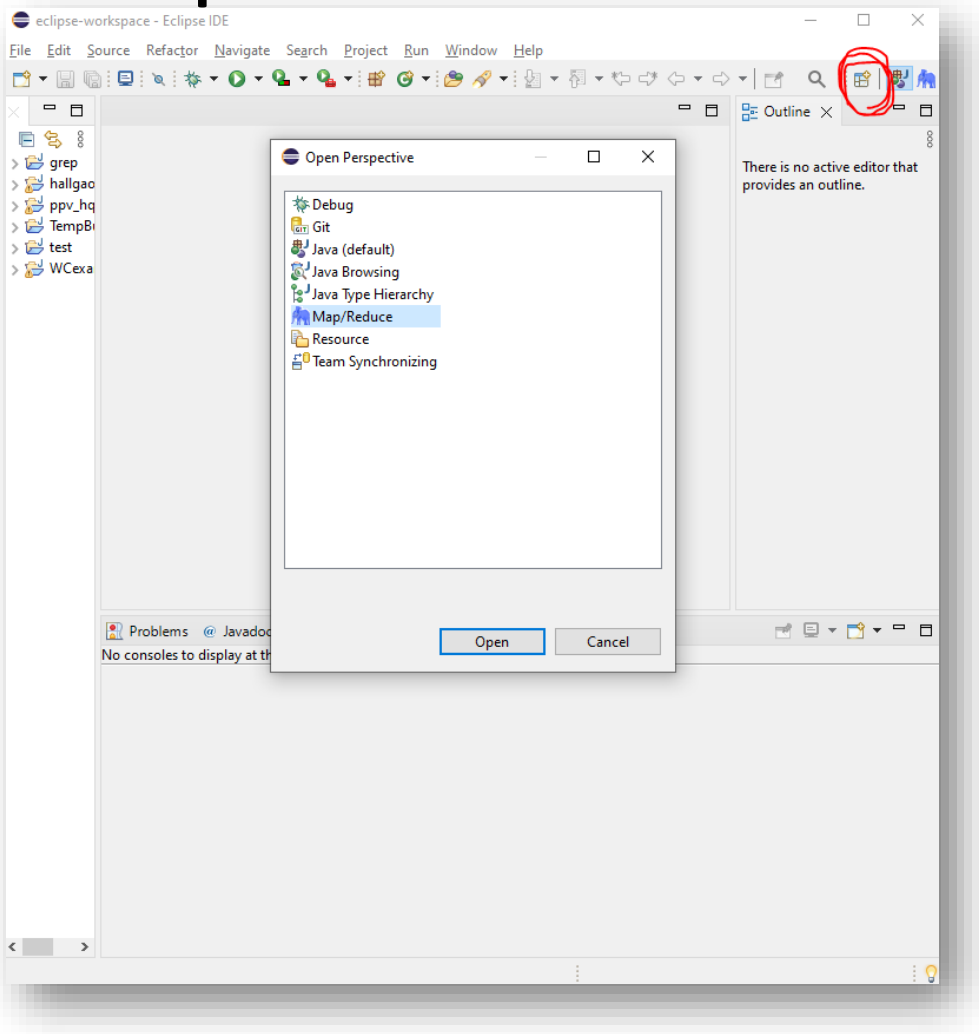
Mai óra célja: környezet kialakítása

- Hadoop opció A:
  1. Töltsd le a BigData.7z és csomagold ki a c:\BigData mappába
  2. Add hozzá a PATH környezeti változóhoz a c:\BigData\hadoop\bin-t

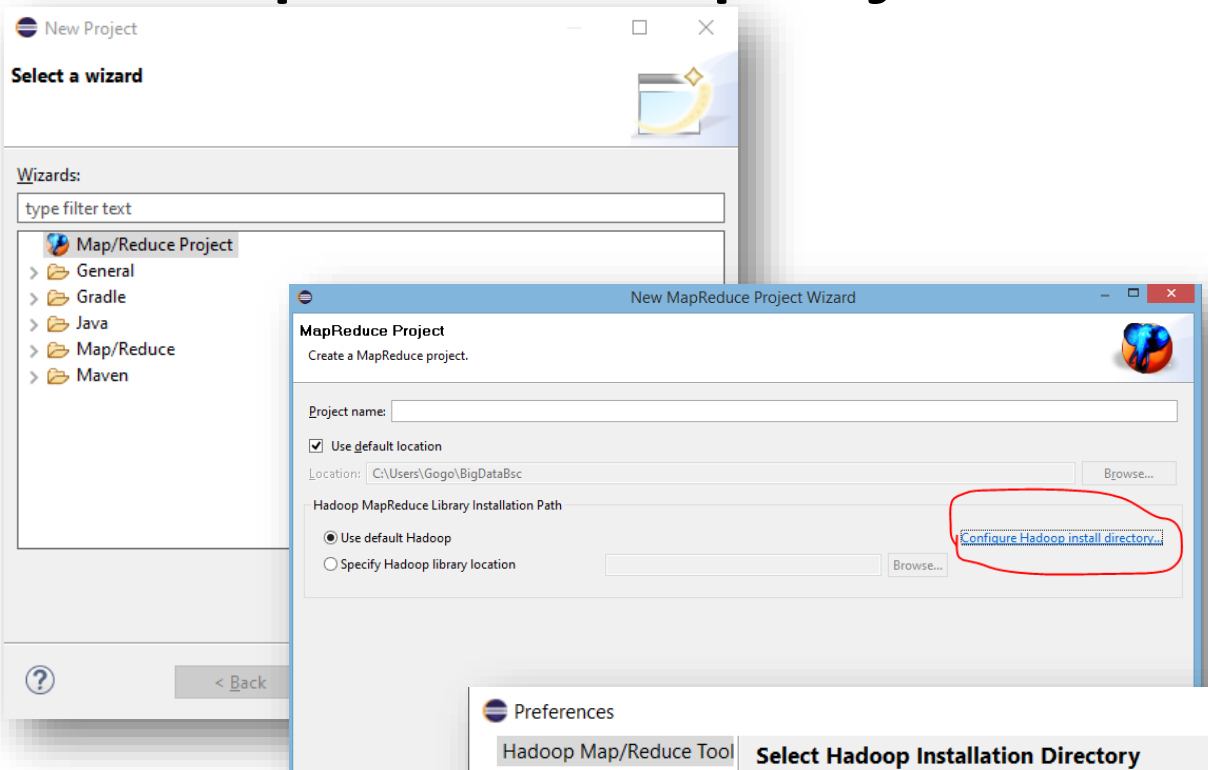
- Hadoop opció B:
  1. Töltsd le és telepítsd az Eclipse-t JRE 20-szal
  2. Töltsd le a Hadoop-ot (pl: 3.3.6)
     - csomagold ki c:\BigData
  3. Töltsd le a Hadoop plug-int
     - Másold be az eclipse\dropins mappába
  4. Töltsd le a winutils, hadoop.dll
     - Másold be c:\BigData\hadoop\bin mappába
  5. Add hozzá a PATH környezeti változóhoz a c:\BigData\hadoop\bin-t
  6. Töltsd le a minimalBigData.zip-et!

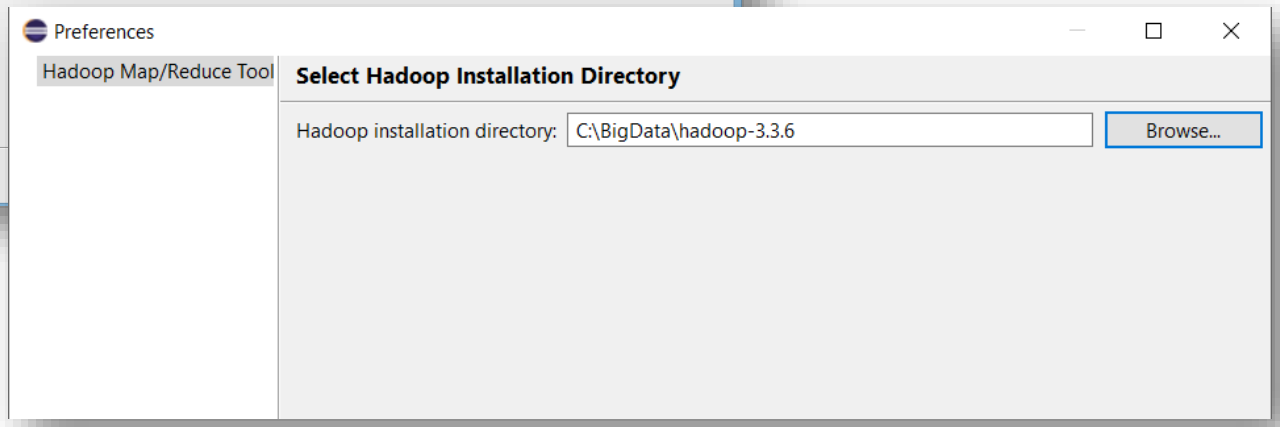
Környezet elindítása, kipróbálása Hadoop

- Indítsd el az Eclipset és állítsd át a nézetet MapReduce-ra!



MapReduce projekt létrehozása

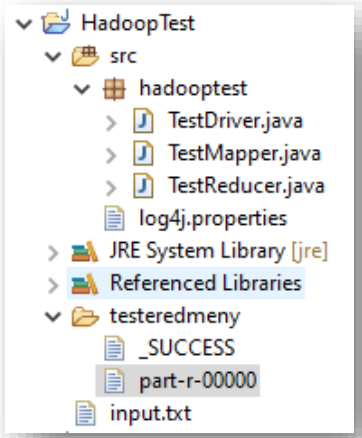


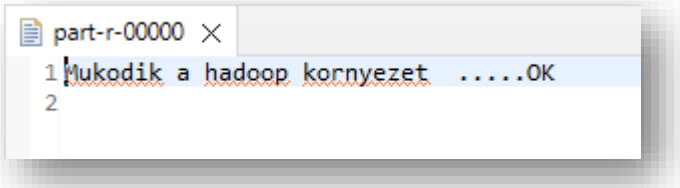


- Hozzunk létre egy MapReduce projektet
- Állítsuk be a Hadoop elérését

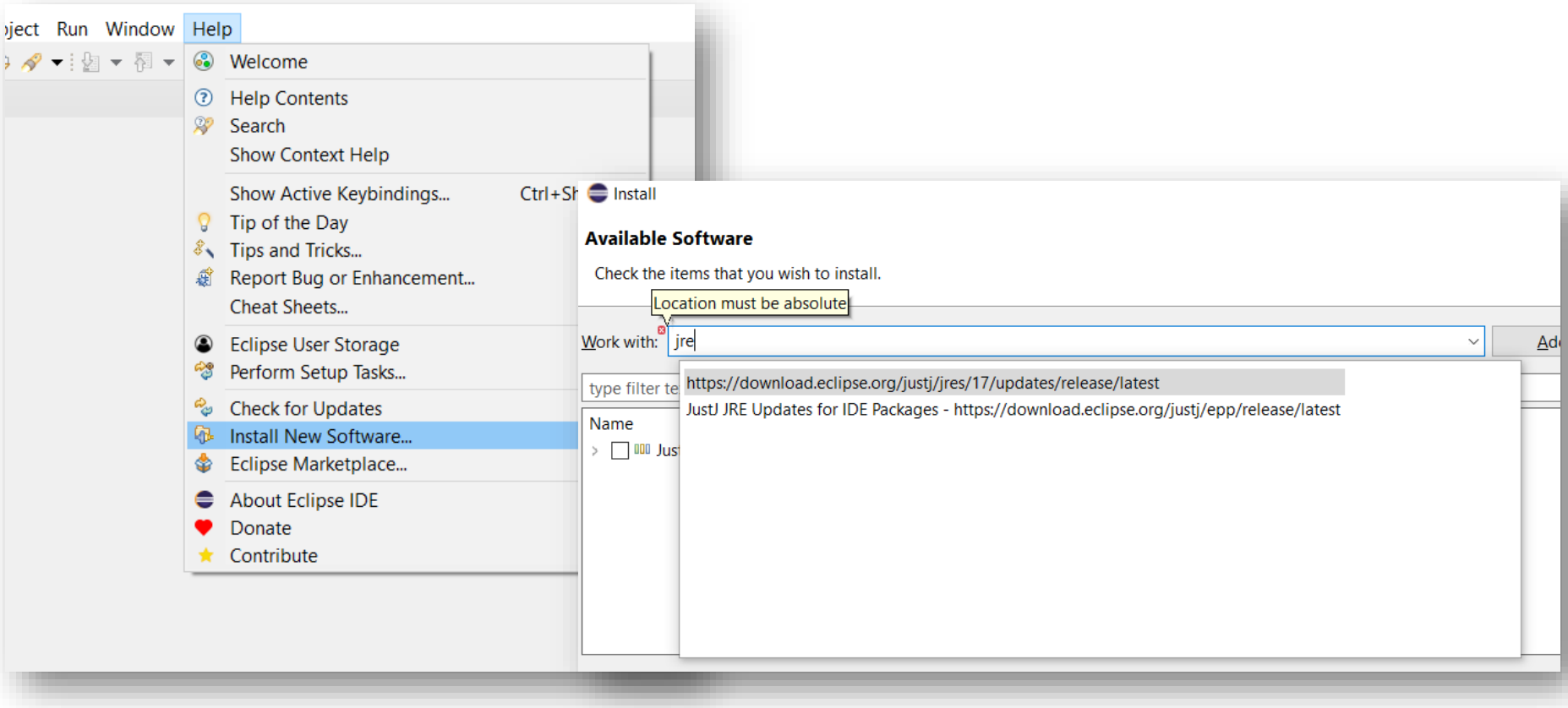
Környezet kipróbálása

- Készítsünk egy hadooptest csomagot (package)
- Másoljuk be a java osztályokat a hadooptest alá
- Másoljuk a log4j.propertiest a src alá
- Másoljuk az input.txt-t a projekt-be
- Indítsuk el „Java Application”-ként
  - TestDriver-t adjuk meg futtatásra





Ha nincs JRE a gépen



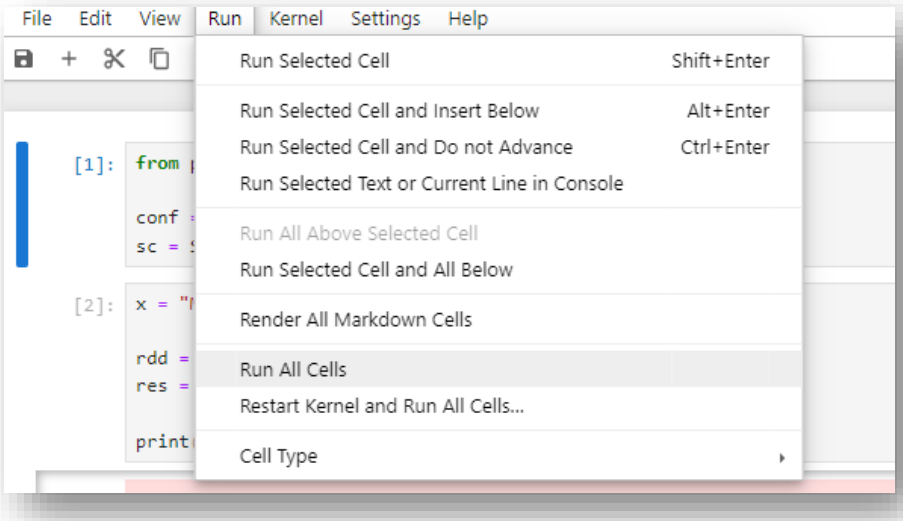
Mai óra célja: környezet kialakítása

- SPARK, pyspark
  1. Nyiss egy parancsori ablakot (cmd)
  2. Add ki a következő utasításokat
     1. pip install pyspark
     2. pip install jupyter
     3. pip install notebook

Környezet elindítása, kipróbálása Spark

1. Indítsuk el a startJupyterSpark.bat
2. Indítsuk el a SparkTest.ipynb-t
3. Futtassuk le az összes mezőt

```
SET PYSPARK_PYTHON=python
SET JAVA_HOME=
python -m notebook
```

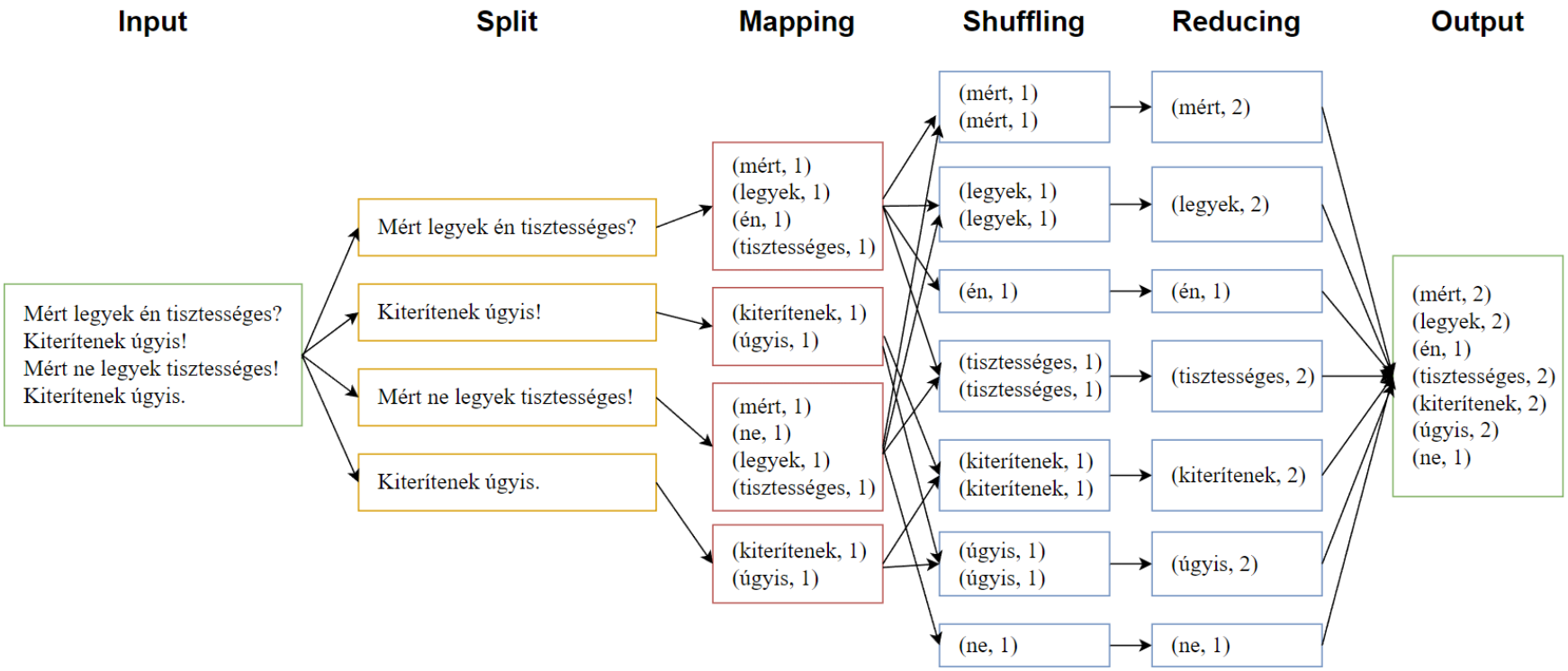



```
[1]: from pyspark import SparkConf, SparkContext

     conf = SparkConf()
     sc = SparkContext(conf=conf)

[2]: x = "MWűGkyösdwinko Qau fSJpDaTrykv!w"

     rdd = sc.parallelize([x[i:i+2] for i in range(0, len(x), 2)])
     res = rdd.map(lambda x: x[0]).collect()

     print("".join(res))

     Működik a Spark!
```

Hadoop

- Nyílt forráskódú keretrendszer
- Lehetőséget ad nagy adathalmazok feldolgozására számítógép klaszterek használatával
- Fő részei
  - Hadoop Distributed File System (HDFS) elosztott fájlrendszer
  - MapReduce programozási modell
  - YARN erőforrás menedzser

Word Count


Input
Split
Mapping
Shuffling
Reducing
Output
Mért legyek én tisztességes?
Kiterítenek úgyis!
Mért ne legyek tisztességes!
Kiterítenek úgyis.
Mért legyek én tisztességes?
Kiterítenek úgyis!
Mért ne legyek tisztességes!
Kiterítenek úgyis.
(mért, 1)
(legyek, 1)
(én, 1)
(tisztességes, 1)
(kiterítenek, 1)
(úgyis, 1)
(mért, 1)
(ne, 1)
(legyek, 1)
(tisztességes, 1)
(kiterítenek, 1)
(úgyis, 1)
(mért, 1)
(mért, 1)
(legyek, 1)
(legyek, 1)
(én, 1)
(tisztességes, 1)
(tisztességes, 1)
(kiterítenek, 1)
(kiterítenek, 1)
(úgyis, 1)
(úgyis, 1)
(ne, 1)
(mért, 2)
(legyek, 2)
(én, 1)
(tisztességes, 2)
(kiterítenek, 2)
(úgyis, 2)
(ne, 1)
(mért, 2)
(legyek, 2)
(én, 1)
(tisztességes, 2)
(kiterítenek, 2)
(úgyis, 2)
(ne, 1)

Köszönöm a Figyelmet!