

Exercice 3 :

1. a) Créons une variable de type Int, attribuons une valeur, puis voir quelles méthodes sont disponibles pour ce type :

- Création de la variable

```
scala> val age=20  
age: Int = 20
```

- Méthodes disponibles pour ce type

```
scala> age.  
!= / >= ceil getClass isPosInfinity isWhole shortValue toDegrees toOctalString underlying  
% < >> compare intValue isValidByte longValue signum toDouble toRadians until  
& << >>> compareTo isInfinite isValidChar max to toFloat toShort  
* <= ^ doubleValue isInfinity isValidInt min to toHexString unary_+  
+ == abs floatValue isNaN isValidLong round toByte toInt unary_-  
- > byteValue floor isNegInfinity isValidShort self toChar toLong unary_~
```

b) Créons une variable de type String, attribuons une valeur, puis voir quelles méthodes sont disponibles pour ce type :

- Création de la variable

```
scala> val nom="bigdata"  
nom: String = bigdata
```

- Méthodes disponibles pour ce type

```
scala> nom.  
* compare forall lastIndexOf reduceOption splitAt toIterator  
+ compareTo foreach lastOption reduceRight toList  
++ compareToIgnoreCase format length reduceRightOption toLong  
++: compose formatLocal lengthCompare regionMatches strip toLowerCase  
+: concat genericBuilder lift repeat stripLeading toMap  
/: contains getBytes lines replace stripLineEnd toSeq  
:+ containsSlice getChars linesIterator replaceAll stripMargin toSet  
:\ contentEquals groupBy linesWithSeparators replaceAllLiterally stripPrefix toShort  
< copyToArray grouped map matches repr stripSuffix toStream  
<= copyToBuffer hasDefiniteSize reverse stripTrailing toString  
> corresponds hashCode max reverse subSequence toTraversable  
>= count head maxBy reverseIterator substring toUpperCase  
addString diff headOption min reverseMap sum toVector  
aggregate distinct indexOf minBy runWith tail transpose  
andThen drop indexOfSlice mkString sameElements tails trim  
apply dropRight indexWhere nonEmpty scan scanLeft take union  
applyOrElse dropWhile indices offsetByCodePoints scanLeft takeRight unzip  
canEqual endsWith init orElse scanRight takeWhile unzip3  
capitalize equals inits padTo segmentLength to updated  
charAt equalsIgnoreCase intern par self toArray view  
chars exists intersect partition seq to toBoolean withFilter  
codePointAt filter isBlank patch size toBuffer zip  
codePointBefore filterNot isDefinedAt permutations slice toByte zipAll  
codePointCount find isEmpty prefixLength sliding toCharArray zipWithIndex  
codePoints flatMap isTraversableAgain product sortBy toDouble  
collect flatten iterator r sorted toFloat  
collectFirst fold last reduce sorted toIndexedSeq  
combinations foldLeft lastIndexOf reduceLeft span toInt  
companion foldRight lastIndexOfSlice reduceLeftOption split toIterable
```

2. explorons les autres options disponibles dans scala repl :

Les options disponibles sont :

```
brice@wambdevps:~/spark$ scala
Welcome to Scala 2.11.12 (OpenJDK 64-Bit Server VM, Java 11.0.16).
Type in expressions for evaluation. Or try :help.

scala> :help
All commands can be abbreviated, e.g., :he instead of :help.
:edit <id>|<line>      edit history
:help [command]        print this summary or command-specific help
:history [num]         show the history (optional num is commands to show)
:h? <string>          search the history
:imports [name name ...] show import history, identifying sources of names
:implicits [-v]        show the implicits in scope
:javap <path|class>    disassemble a file or class name
:line <id>|<line>      place line(s) at the end of history
:load <path>          interpret lines in a file
:paste [-raw] [path]   enter paste mode or paste a file
:power                enable power user mode
:quit                 exit the interpreter
:replay [options]      reset the repl and replay all previous commands
:require <path>        add a jar to the classpath
:reset [options]       reset the repl to its initial state, forgetting all session entries
:save <path>          save replayable session to a file
:sh <command line>    run a shell command (result is implicitly => List[String])
:settings <options>   update compiler options, if possible; see reset
:silent              disable/enable automatic printing of results
:type [-v] <expr>    display the type of an expression without evaluating it
:kind [-v] <expr>   display the kind of expression's type
:warnings            show the suppressed warnings from the most recent line which had any
```

- « :sh » exécuter une commande shell (le résultat est implicitement => Liste[String])

```
scala> :sh mkdir folderdatas
res0: scala.tools.nsc.interpreter.ProcessResult = `mkdir folderdatas` (0 lines, exit 0)

scala> :sh touch folderdata/folder
res1: scala.tools.nsc.interpreter.ProcessResult = `touch folderdata/folder` (1 lines, exit 1)

scala> :sh touch folderdatas/datas
res2: scala.tools.nsc.interpreter.ProcessResult = `touch folderdatas/datas` (0 lines, exit 0)

scala> :sh ls folderdatas
res3: scala.tools.nsc.interpreter.ProcessResult = `ls folderdatas` (1 lines, exit 0)

scala> res3.line foreach println
l
s

f
o
l
d
e
r
d
a
t
a
s

scala> res3.lines foreach println
datas
```

- « :save » sauvegarde de la session rejouable dans un fichier

```
brice@wambdevps:~$ scala
Welcome to Scala 2.11.12 (OpenJDK 64-Bit Server VM, Java 11.0.16).
Type in expressions for evaluation. Or try :help.

scala> 1
res0: Int = 1

scala> 2
res1: Int = 2

scala> 3
res2: Int = 3

scala> :save xxx

scala> 
```

- « :load » interpréter les lignes d'un fichier

```
scala> 1
res0: Int = 1

scala> 2
res1: Int = 2

scala> 3
res2: Int = 3

scala> :save xxx

scala> :load xxx
Loading xxx...
res3: Int = 1
res4: Int = 2
res5: Int = 3

scala> 
```

- « :reset »

```

brice@wambdevps:~$ scala
Welcome to Scala 2.11.12 (OpenJDK 64-Bit Server VM, Java 11.0.16).
Type in expressions for evaluation. Or try :help.

scala> 1
res0: Int = 1

scala> res0 + 1
res1: Int = 2

scala> :save xxx

scala>
brice@wambdevps:~$
brice@wambdevps:~$ scala
Welcome to Scala 2.11.12 (OpenJDK 64-Bit Server VM, Java 11.0.16).
Type in expressions for evaluation. Or try :help.

scala> 7
res0: Int = 7

scala> :reset
Resetting interpreter state.
Forgetting this session history:

7

Forgetting all expression results and named terms: $intp

scala> :load xxx
Loading xxx...
res0: Int = 1
res1: Int = 2

scala> █

```

3. Recherchons le spark-shell et voyons si les mêmes commandes et fonctionnalités y sont également disponibles.

- Recherche et installation de spark

```

brice@wambdevps:~/spark$ spark-shell
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
22/10/26 14:32:18 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Spark context Web UI available at http://wambdevps:4040
Spark context available as 'sc' (master = local[*], app id = local-1666791140961).
Spark session available as 'spark'.
Welcome to

  ____
 /  __ \
/   /  \
/_____/  version 3.3.1

Using Scala version 2.12.15 (OpenJDK 64-Bit Server VM, Java 11.0.16)
Type in expressions to have them evaluated.
Type :help for more information.

```

- voyons si les mêmes commandes et fonctionnalités y sont également disponibles.

```

brice@wambdevps:~/spark$ spark-shell
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
22/10/26 15:42:18 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Spark context Web UI available at http://wambdevps:4040
Spark context available as 'sc' (master = local[*], app id = local-1666795340509).
Spark session available as 'spark'.
Welcome to

  ____      _
 / ___|    / \
| |  | |  / _ \
| |  | | / ___ \
| |  | || |___| \
| |  | || |___| \
| |  | || |___| \
|_|  |_| \____/

version 3.3.1

Using Scala version 2.12.15 (OpenJDK 64-Bit Server VM, Java 11.0.16)
Type in expressions to have them evaluated.
Type :help for more information.

scala> :help
All commands can be abbreviated, e.g., :he instead of :help.
:completions <string>      output completions for the given string
:edit <id>|<line>          edit history
:help <command>            print this summary or command-specific help
:history <num>             show the history (optional num is commands to show)
:h? <string>              search the history
:imports <name name ...>   show import history, identifying sources of names
:implicits [-v]            show the implicits in scope
:javap <path>|<class>      disassemble a file or class name
:line <id>|<line>          place line(s) at the end of history
:load <path>               interpret lines in a file
:paste [-raw] <path>       enter paste mode or paste a file
:power                     enable power user mode
:quit                     exit the interpreter
:replay <options>          reset the repl and replay all previous commands
:require <path>            add a jar to the classpath
:reset <options>           reset the repl to its initial state, forgetting all session entries
:save <path>               save replayable session to a file
:sh <command line>        run a shell command (result is implicitly => List[String])
:settings <options>       update compiler options, if possible; see reset
:silent                   disable/enable automatic printing of results
:type [-v] <expr>         display the type of an expression without evaluating it
:kind [-v] <type>         display the kind of a type. see also :help kind
:warnings                 show the suppressed warnings from the most recent line which had any

```

En conclusion, nous constatons que les mêmes commandes et fonctionnalités sont également disponibles.

4. essayons d'augmenter la mémoire utilisée par le shell scala.