

# Package ‘easyreporting’

March 30, 2021

**Type** Package

**Title** Helps creating report for improving Reproducible computational Research

**Version** 1.2.0

**Description** An S4 class for facilitating the automated creation of rmarkdown files inside other packages/software, even without knowing rmarkdown language. Best if implemented in functions as ``recursive" style programming.

**License** Artistic-2.0

**Encoding** UTF-8

**biocViews** ReportWriting

**LazyData** false

**BugReports** <https://github.com/drighelli/easyreporting/issues>

**Imports** rmarkdown, methods, tools

**Suggests** BiocStyle, knitr, readxl, edgeR, limma, EDASeq, statmod

**RoxygenNote** 7.0.2

**VignetteBuilder** knitr

**git\_url** <https://git.bioconductor.org/packages/easyreporting>

**git\_branch** RELEASE\_3\_12

**git\_last\_commit** 34555b4

**git\_last\_commit\_date** 2020-10-27

**Date/Publication** 2021-03-29

**Author** Dario Righelli [cre, aut]

**Maintainer** Dario Righelli <dario.righelli@gmail.com>

## R topics documented:

compile . . . . .	2
compile,easyreporting-method . . . . .	3
easyreporting . . . . .	3
getOptionsList . . . . .	4
getOptionsList,easyreporting-method . . . . .	5
getReportFilename . . . . .	5
getReportFilename,easyreporting-method . . . . .	6

makeOptionsList . . . . .	6
mkdCodeChunkCommented . . . . .	7
mkdCodeChunkCommented,easyreporting-method . . . . .	8
mkdCodeChunkComplete . . . . .	9
mkdCodeChunkComplete,easyreporting-method . . . . .	9
mkdCodeChunkEnd . . . . .	10
mkdCodeChunkEnd,easyreporting-method . . . . .	11
mkdCodeChunkSt . . . . .	11
mkdCodeChunkSt,easyreporting-method . . . . .	12
mkdCodeChunkTitledCommented . . . . .	13
mkdCodeChunkTitledCommented,easyreporting-method . . . . .	14
mkdGeneralMsg . . . . .	15
mkdGeneralMsg,easyreporting-method . . . . .	15
mkdGeneralTitledMsg . . . . .	16
mkdGeneralTitledMsg,easyreporting-method . . . . .	17
mkdTitle . . . . .	17
mkdTitle,easyreporting-method . . . . .	18
mkdVariableAssignment . . . . .	19
mkdVariableAssignment,easyreporting-method . . . . .	19
setOptionsList . . . . .	20
setOptionsList,easyreporting-method . . . . .	21

**Index** **23**

---

compile	<i>compile</i>
---------	----------------

---

**Description**

prints the sessionInfo and compiles the produced rmarkdown file into an HTML report.

**Usage**

```
compile(object)
```

**Arguments**

object            an easyreporting class object

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report", title="example_report",
                    author=c("It's me"))
compile(rd)
```

---

```
compile,easyreporting-method
      compile
```

---

**Description**

prints the sessionInfo and compiles the rmarkdown file

**Usage**

```
## S4 method for signature 'easyreporting'
compile(object)
```

**Arguments**

object            an easyreporting class object

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report", title="example_report",
                    author=c("It's me"))
compile(rd)
```

---

```
easyreporting            easyreporting
```

---

**Description**

An S4 class for managing rmarkdown report. Each instance describes an rmarkdown file.

**Usage**

```
easyreporting(filenamePath, title, author, optionList = NULL)
```

**Arguments**

filenamePath    the path with the name of the rmarkdown.  
title            the title of the report section.  
author           the author of the report.  
optionList       a list of options for the general rmarkdown document.

**Value**

an S4 easyreporting class instance

**Slots**

`filenamePath` the path with the name of the rmarkdown.  
`title` the title of the report section.  
`author` the author of the report.  
`documentType` the type of the final document (fixed to "html").  
`optionList` a list of options for the general rmarkdown document.

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",  
                    title="example_report", author="It's me")
```

---

<code>getOptionsList</code>	<i>getOptionsList</i>
-----------------------------	-----------------------

---

**Description**

returns the `optionList` from the `easyreporting` class (see the `makeOptionList` function for more details).

**Usage**

```
getOptionsList(object)
```

**Arguments**

`object` an `easyreporting` class object

**Value**

a list of options

**Examples**

```
rd <- easyreporting(filenamePath="./project_report", title="example_report",  
                    author=c("It's me"))  
optList <- getOptionsList(rd)
```

---

`getOptionsList,easyreporting-method`  
*getOptionsList*

---

### **Description**

returns the optionList from the easyreporting class

### **Usage**

```
## S4 method for signature 'easyreporting'  
getOptionsList(object)
```

### **Arguments**

object            an easyreporting class object

### **Value**

a list of options

### **Examples**

```
rd <- easyreporting(filenamePath="./project_report", title="example_report",  
                    author=c("It's me"))  
optList <- getOptionsList(rd)
```

---

`getReportFilename`        *getReportFilename*

---

### **Description**

returns the filename with its path of the report generated by easyreporting

### **Usage**

```
getReportFilename(object)
```

### **Arguments**

object            an easyreporting class object

### **Value**

a string of report file name with path

### **Examples**

```
rd <- easyreporting(filenamePath="./project_report",  
                    title="example_report", author=c("It's me"))  
(rep <- getReportFilename(rd))
```

```
getReportFilename, easyreporting-method  
getReportFilename
```

---

**Description**

returns the report filename with path

**Usage**

```
## S4 method for signature 'easyreporting'  
getReportFilename(object)
```

**Arguments**

object            an easyreporting class object

**Value**

a string of report file name with path

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",  
                    title="example_report", author=c("It's me"))  
(rep <- getReportFilename(rd))
```

---

```
makeOptionsList            makeOptionsList
```

---

**Description**

makes an list of rmarkdown options

**Usage**

```
makeOptionsList(  
  cacheFlag = TRUE,  
  evalFlag = TRUE,  
  echoFlag = TRUE,  
  warningFlag = FALSE,  
  showMessages = FALSE,  
  includeFlag = TRUE  
)
```

**Arguments**

cacheFlag	boolean for caching chunk data (default TRUE)
evalFlag	boolean for evaluating the code chunk in the compiled version (default TRUE)
echoFlag	boolean for showing the code chunk (default TRUE)
warningFlag	boolean for showing the chunk warnings (default FALSE)
showMessages	boolean for showing the chunk warnings in compiled version (default FALSE)
includeFlag	boolean for including the code chunk in the compiled version (default TRUE)

**Value**

list of rmarkdown options

**Examples**

```
optList <- makeOptionsList()
```

---

mkdCodeChunkCommented *mkdCodeChunkCommented*

---

**Description**

it creates a complete code chunk, adding a natural language comment before of it.

**Usage**

```
mkdCodeChunkCommented(
  object,
  commentMsg = NULL,
  codeMsg,
  optionList = getOptionsList(object),
  sourceFilesList = NULL
)
```

**Arguments**

object	an easyreporting class object
commentMsg	a string with the natural language comment for the chunk.
codeMsg	a string within the code.
optionList	a list of options (default is the class options).
sourceFilesList	a optional list of files to source inside the chunk.

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",
                    title="example_report", author=c("It's me"))
optList <- makeOptionsList(includeFlag=TRUE, cacheFlag=TRUE)
mkdCodeChunkCommented(rd,
                      commentMsg="This is the comment of the following code chunk",
                      codeMsg="a <- 1\nb <- 2\n(c <- a+b)\n", optionList=optList,
                      sourceFilesList=NULL)
```

---

*mkdCodeChunkCommented,easyreporting-method*  
*mkdCodeChunkCommented*

---

**Description**

it creates a complete code chunk, adding a natural language comment before of it.

**Usage**

```
## S4 method for signature 'easyreporting'
mkdCodeChunkCommented(
  object,
  commentMsg = NULL,
  codeMsg,
  optionList = getOptionsList(object),
  sourceFilesList = NULL
)
```

**Arguments**

<code>object</code>	an easyreporting class object
<code>commentMsg</code>	a string with the natural language comment for the chunk.
<code>codeMsg</code>	a string within the code.
<code>optionList</code>	a list of options (default is the class options).
<code>sourceFilesList</code>	an optional list of files to source inside the chunk.

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",
                    title="example_report", author=c("It's me"))
optList <- makeOptionsList(includeFlag=TRUE, cacheFlag=TRUE)
mkdCodeChunkCommented(rd,
                      commentMsg="This is the comment of the following code chunk",
                      codeMsg="a <- 1\nb <- 2\n(c <- a+b)\n", optionList=optList,
                      sourceFilesList=NULL)
```



---

 mkdCodeChunkComplete *mkdCodeChunkComplete*


---

**Description**

it creates a complete code chunk.

**Usage**

```
mkdCodeChunkComplete(
  object,
  message,
  optionList = getOptionsList(object),
  sourceFilesList = NULL
)
```

**Arguments**

object	an easyreporting class object
message	a string containing a function call or the entire code chunk to trace.
optionList	a list of options.
sourceFilesList	a list of files to source.

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",
  title="example_report", author=c("It's me"))
mkdCodeChunkComplete(rd, message="a <- 1\nb <- 2\nc <- a+b\n print(c)")
```

---

 mkdCodeChunkComplete, easyreporting-method  
*mkdCodeChunkComplete*


---

**Description**

it creates a complete code chunk.

**Usage**

```
## S4 method for signature 'easyreporting'
mkdCodeChunkComplete(
  object,
  message,
  optionList = getOptionsList(object),
  sourceFilesList = NULL
)
```

**Arguments**

object            an easyreporting class object  
message          a string containing a function call or the entire code chunk to trace.  
optionList       a list of options.  
sourceFilesList   a list of files to source.

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",  
                      title="example_report", author=c("It's me"))  
mkdCodeChunkComplete(rd, message="a <- 1\nb <- 2\nc <- a+b\n print(c)")
```

---

mkdCodeChunkEnd

*mkdCodeChunkEnd*

---

**Description**

it creates a code chunk end. Always use it after a mkdCodeChunkSt()

**Usage**

```
mkdCodeChunkEnd(object)
```

**Arguments**

object            an easyreporting class object

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",  
                      title="example_report", author=c("It's me"))  
mkdCodeChunkSt(rd)  
## just leaving empty  
mkdCodeChunkEnd(rd)
```

---

mkdCodeChunkEnd,easyreporting-method  
*mkdCodeChunkEnd*

---

### Description

it creates a code chunk end. Always use it after a mkdCodeChunkSt()

### Usage

```
## S4 method for signature 'easyreporting'  
mkdCodeChunkEnd(object)
```

### Arguments

object            an easyreporting class object

### Value

none

### Examples

```
rd <- easyreporting(filenamePath="./project_report",  
                      title="example_report", author=c("It's me"))  
mkdCodeChunkSt(rd)  
## just leaving empty  
mkdCodeChunkEnd(rd)
```

---

mkdCodeChunkSt            *mkdCodeChunkSt*

---

### Description

it creates a code chunk start. A list of options and files to source for the chunk can optionally be passed to the function.

### Usage

```
mkdCodeChunkSt(  
  object,  
  optionList = getOptionsList(object),  
  sourceFilesList = NULL,  
  isComplete = FALSE  
)
```

**Arguments**

object	an easyreporting class object
optionList	a list of options
sourceFilesList	a list of files that can be sourced inside the code chunk.
isComplete	a flag determining if the chunk is already a complete chunk

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",
                    title="example_report", author=c("It's me"))

## no options
mkdCodeChunkSt(rd)
## just leaving empty
mkdCodeChunkEnd(rd)

## setting options
optList <- makeOptionsList(includeFlag=TRUE)
mkdCodeChunkSt(rd, optionList=optList)
## just leaving empty
mkdCodeChunkEnd(rd)
```

---

*mkdCodeChunkSt,easyreporting-method*  
*mkdCodeChunkSt*

---

**Description**

it creates a code chunk start. A list of options and files to source for the chunk can optionally be passed to the function.

**Usage**

```
## S4 method for signature 'easyreporting'
mkdCodeChunkSt(
  object,
  optionList = getOptionsList(object),
  sourceFilesList = NULL,
  isComplete = FALSE
)
```

**Arguments**

object	an easyreporting class object
optionList	a list of options
sourceFilesList	a list of files that can be sourced inside the code chunk.
isComplete	a flag determining if the chunk is already a complete chunk

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",
                    title="example_report", author=c("It's me"))

## no options
mkdCodeChunkSt(rd)
## just leaving empty
mkdCodeChunkEnd(rd)

## setting options
optList <- makeOptionsList(includeFlag=TRUE)
mkdCodeChunkSt(rd, optionList=optList)
## just leaving empty
mkdCodeChunkEnd(rd)
```

---

```
mkdCodeChunkTitledCommented
      mkdCodeChunkCommented
```

---

**Description**

it creates a complete code chunk, adding a natural language comment before of it.

**Usage**

```
mkdCodeChunkTitledCommented(
  object,
  title = NULL,
  level = 1,
  commentMsg = NULL,
  codeMsg,
  optionList = getOptionsList(object),
  sourceFilesList = NULL
)
```

**Arguments**

<code>object</code>	an easyreporting class object
<code>title</code>	the title to assign to the code chunk section
<code>level</code>	the level of the title (default is 1)
<code>commentMsg</code>	a string with the natural language comment for the chunk.
<code>codeMsg</code>	a string within the code.
<code>optionList</code>	a list of options (default is the class options).
<code>sourceFilesList</code>	a optional list of files to source inside the chunk.

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",
                    title="example_report", author=c("It's me"))
optList <- makeOptionsList(includeFlag=TRUE, cacheFlag=TRUE)
mkdCodeChunkCommented(rd,
                      commentMsg="This is the comment of the following code chunk",
                      codeMsg="a <- 1\nb <- 2\n(c <- a+b)\n", optionList=optList,
                      sourceFilesList=NULL)
```

---

```
mkdCodeChunkTitledCommented,easyreporting-method
      mkdCodeChunkTitledCommented
```

---

**Description**

it creates a complete code chunk, adding a natural language comment before of it.

**Usage**

```
## S4 method for signature 'easyreporting'
mkdCodeChunkTitledCommented(
  object,
  title = NULL,
  level = 1,
  commentMsg = NULL,
  codeMsg,
  optionList = getOptionsList(object),
  sourceFilesList = NULL
)
```

**Arguments**

<code>object</code>	an easyreporting class object
<code>title</code>	the title to assign to the code chunk section
<code>level</code>	the level of the title (default is 1)
<code>commentMsg</code>	a string with the natural language comment for the chunk.
<code>codeMsg</code>	a string within the code.
<code>optionList</code>	a list of options (default is the class options).
<code>sourceFilesList</code>	an optional list of files to source inside the chunk.

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",
                    title="example_report", author=c("It's me"))
optList <- makeOptionsList(includeFlag=TRUE, cacheFlag=TRUE)
mkdCodeChunkTitledCommented(rd, title="Title Example", level=1,
                             commentMsg="This is the comment of the following code chunk",
                             codeMsg="a <- 1\nb <- 2\n(c <- a+b)\n", optionList=optList,
                             sourceFilesList=NULL)
```

---

mkdGeneralMsg

*mkdGeneralMsg*


---

**Description**

It appends a general message to the report, useful for describing a code chunk or a part of the report. Useful for adding natural language comments.

**Usage**

```
mkdGeneralMsg(object, message)
```

**Arguments**

object	an easyreporting class object
message	the message to append to the report

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",
                    title="example_report", author=c("It's me"))
mkdGeneralMsg(rd, message="Writing a paragraph to describe my code chunk")
```

---

mkdGeneralMsg, easyreporting-method

*mkdGeneralMsg*


---

**Description**

It appends a general message to the report. Useful for adding natural language comments.

**Usage**

```
## S4 method for signature 'easyreporting'
mkdGeneralMsg(object, message)
```

**Arguments**

object	an easyreporting class object
message	the message to append to the report

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",
                    title="example_report", author=c("It's me"))
mkdGeneralMsg(rd, "Writing a paragraph to describe my code chunk")
```

---

`mkdGeneralTitledMsg`    *mkdGeneralTitledMsg*

---

**Description**

It appends a a titled section followed by a general message to the report. Useful for adding natural language comments.

**Usage**

```
mkdGeneralTitledMsg(object, title = NULL, level = 1, message)
```

**Arguments**

object	an easyreporting class object
title	the title of the report section.
level	the level (1 to 6) of the title (default is 1)
message	the message to append to the report

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",
                    title="example_report", author=c("It's me"))
mkdGeneralTitledMsg(rd, title="Generic SubTitle for this message", level=2,
                    message="Writing a paragraph to describe my code chunk")
```



---

```
mkdGeneralTitledMsg,easyreporting-method
      mkdGeneralTitledMsg
```

---

**Description**

It appends a a titled section followed by a general message to the report. Useful for adding natural language comments.

**Usage**

```
## S4 method for signature 'easyreporting'
mkdGeneralTitledMsg(object, title = NULL, level = 1, message)
```

**Arguments**

object	an easyreporting class object
title	the optional title to give to the message section
level	the level (1 to 6) of the title (default is 1)
message	the message to append to the report

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",
                    title="example_report", author=c("It's me"))
mkdGeneralTitledMsg(rd, title="Generic SubTitle for this message", level=2,
                    message="Writing a paragraph to describe my code chunk")
```

---

```
mkdTitle          mkdTitle
```

---

**Description**

Inserts an rmarkdown title inside the report, useful for create distinct sections inside the final resport.

**Usage**

```
mkdTitle(object, title, level = 1)
```

**Arguments**

object	an easyreporting class object
title	a string within the title.
level	a numeric from 1 to 6 (default is 1).

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",
                    title="example_report", author=c("It's me"))

mkdTitle(rd, "First Level Title")
mkdTitle(rd, "Sub-Title", level=2)
```

---

*mkdTitle,easyreporting-method*  
*mkdTitle*

---

**Description**

Inserts an rmarkdown title inside the report.

**Usage**

```
## S4 method for signature 'easyreporting'
mkdTitle(object, title, level = 1)
```

**Arguments**

object	an easyreporting class object
title	a string within the title.
level	a numeric from 1 to 6 (default is 1).

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",
                    title="example_report", author=c("It's me"))

mkdTitle(rd, "First Level Title")
mkdTitle(rd, "Sub-Title", level=2)
```

---

 mkdVariableAssignment *mkdVariableAssignment*


---

**Description**

it includes a variable assignment in the report. NB: a call to the "mkdCodeChunkSt" has to be done before using it.

**Usage**

```
mkdVariableAssignment(object, variable.name, variable.object.name, show)
```

**Arguments**

object	an easyreporting class object
variable.name	a string indicating the name of the variable to store in the report. (This can be changed here, but further uses of the variable needs to take into account the variable name change).
variable.object.name	the name of the already existing variable. (This cannot be changed.)
show	a boolean indicating if to show the message before writing it into the rmarkdown file.

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",
                    title="example_report", author=c("It's me"))
## leaving the default options to the code chunk
mkdCodeChunkSt(rd)
## adding a variable assignment
variable <- 1
mkdVariableAssignment(rd, "variable", "variable", show=TRUE)
mkdCodeChunkEnd(rd)
```

---

 mkdVariableAssignment, easyreporting-method  
*mkdVariableAssignment*


---

**Description**

it includes a variable assignment in the report. NB: a call to the "mkdCodeChunkSt" has to be done before using it.

**Usage**

```
## S4 method for signature 'easyreporting'
mkdVariableAssignment(
  object,
  variable.name,
  variable.object.name,
  show = FALSE
)
```

**Arguments**

`object` an easyreporting class object

`variable.name` a string indicating the name of the variable to store in the report. (This can be changed here, but further uses of the variable needs to take into account the variable name change).

`variable.object.name` the name of the already existing variable. (This cannot be changed.)

`show` a boolean indicating if to show the message before writing it into the rmarkdown file.

**Value**

none

**Examples**

```
rd <- easyreporting(filenamePath="./project_report",
                    title="example_report", author=c("It's me"))
## leaving the default options to the code chunk
mkdCodeChunkSt(rd)
## adding a variable assignment
variable <- 1
mkdVariableAssignment(rd, "variable", "variable", show=TRUE)
mkdCodeChunkEnd(rd)
```

---

setOptionsList

*setOptionsList*

---

**Description**

set an optionList to the class, it automatically sets the options for the rmarkdown file. Are useful for the interpretation/execution of the code chunks.

**Usage**

```
setOptionsList(
  object,
  cacheFlag = TRUE,
  evalFlag = TRUE,
  echoFlag = TRUE,
```

```
warningFlag = FALSE,
showMessages = FALSE,
includeFlag = TRUE
)
```

### Arguments

object	an easyreporting class object
cacheFlag	boolean for caching chunk data (default TRUE)
evalFlag	boolean for evaluating the code chunk in the compiled version (default TRUE)
echoFlag	boolean for showing the code chunk (default TRUE)
warningFlag	boolean for showing the chunk warnings (default FALSE)
showMessages	boolean for showing the chunk warnings in compiled
includeFlag	boolean for including the code chunk in the compiled version (default TRUE)

### Value

none

### Examples

```
rd <- easyreporting(filenamePath="./project_report",
                    title="example_report", author=c("It's me"))

## setting default option
setOptionsList(rd)

## modifying only some options
rd <- setOptionsList(rd, warningFlag=TRUE,
                    showMessages=TRUE, includeFlag=TRUE)
```

---

setOptionsList,easyreporting-method  
*setOptionsList*

---

### Description

set an optionList to the class

### Usage

```
## S4 method for signature 'easyreporting'
setOptionsList(
  object,
  cacheFlag = TRUE,
  evalFlag = TRUE,
  echoFlag = TRUE,
  warningFlag = FALSE,
  showMessages = FALSE,
  includeFlag = TRUE
)
```



# Index

[compile](#), [2](#)  
[compile](#), [easyreporting-method](#), [3](#)

[easyreporting](#), [3](#)  
[easyreporting-class](#) ([easyreporting](#)), [3](#)

[getOptionsList](#), [4](#)  
[getOptionsList](#), [easyreporting-method](#), [5](#)  
[getReportFilename](#), [5](#)  
[getReportFilename](#), [easyreporting-method](#),  
[6](#)

[makeOptionsList](#), [6](#)  
[mkdCodeChunkCommented](#), [7](#)  
[mkdCodeChunkCommented](#), [easyreporting-method](#),  
[8](#)  
[mkdCodeChunkComplete](#), [9](#)  
[mkdCodeChunkComplete](#), [easyreporting-method](#),  
[9](#)  
[mkdCodeChunkEnd](#), [10](#)  
[mkdCodeChunkEnd](#), [easyreporting-method](#),  
[11](#)  
[mkdCodeChunkSt](#), [11](#)  
[mkdCodeChunkSt](#), [easyreporting-method](#),  
[12](#)  
[mkdCodeChunkTitledCommented](#), [13](#)  
[mkdCodeChunkTitledCommented](#), [easyreporting-method](#),  
[14](#)  
[mkdGeneralMsg](#), [15](#)  
[mkdGeneralMsg](#), [easyreporting-method](#), [15](#)  
[mkdGeneralTitledMsg](#), [16](#)  
[mkdGeneralTitledMsg](#), [easyreporting-method](#),  
[17](#)  
[mkdTitle](#), [17](#)  
[mkdTitle](#), [easyreporting-method](#), [18](#)  
[mkdVariableAssignment](#), [19](#)  
[mkdVariableAssignment](#), [easyreporting-method](#),  
[19](#)

[setOptionsList](#), [20](#)  
[setOptionsList](#), [easyreporting-method](#),  
[21](#)