

## Dialog

```
- <<final>> ACTION_CHANNEL : String = "ur"  
- <<final>> CHANNEL_SWING : String = "swing"  
- <<final>> CHANNEL_NEW_SWING : String = "new swing"  
- <<final>> CHANNEL_TEXT = ""
```

```
+ Dialog (backend : InteractionDriver)  
+ openMenu : Menu : void  
+ fill (form : Form) : void  
+ send (title : String, text : String)  
+ close : void
```

## <<interface>>

### InteractionDriver

```
+ open (menu : Menu) : void  
+ fill (form : Form) : void  
+ send (title : String, text : String)  
+ close : void
```

## Display

## Display

```
- _title : String  
- _text : String Builder
```

```
+ Display (title : String)  
+ Display (ur : Dialog, title : String)  
+ add (toAdd : Object) : Display  
+ addAll (elems : collection<?>) : Display  
+ addLine (toAdd : Object) : Display  
+ addLine (toAdd : Object, force : boolean) : Display  
+ <<final>> display () : void  
+ <<final>> displayText () : void  
+ pop (toPop : Object) : void  
+ pop (toPop : Object) : void  
+ clear () : void
```

Pl. tecnico. utilib

## Form

- title : String

```
+ form()
+ form(title : String)
+ form(ur : Dialog, title : String)
+ title() : String
+ entries() : Collection<Field<?>>
+ field(key : String) : Field<?>
- add(key : String, fn : Field<?>) : void
+ addBooleanField(key : String, label : String) : void
+ addStringField(key : String, label : String) : void
+ addRealField(key : String, label : String) : void
+ addIntegerField(key : String, label : String) : void
+ addOptionField(key : String, label : String, options : String...) : void
- get(key : String, type : String) : Object
+ booleanField(key : String) : Boolean
+ stringField(key : String) : String
+ optionField(key : String) : String
+ realField(key : String) : Double
+ integerField(key : String) : Integer
+ parse() : Form
+ parse(clear : boolean) : Form
+ clear() : void
+ confirm(prompt : String) : Boolean
+ requestInteger(prompt : String) : Integer
+ requestReal(prompt : String) : Double
+ requestString(prompt : String) : String
+ requestOption(prompt : String, options : String...) : String
```

## <<interface>> Message

```
+ keyAlreadyExists(form : String, key : String) : String
+ formNotFilled(form : String) : String
+ keyNotFound(form : String, key : String) : String
+ typeMismatch(form : String, key : String, actualType : String, requestedType : String) : String
```

## FormException

- <<final>> serialVersionUID : long = 202010212136L

+ FormException(msg : String)

Photochemo.uv.kit.focus



<<abstract>> Field	Type: class
- _prompt: String - _clear: boolean # -value: Type	
# Field() # Field(prompt: String) + prompt(): String + set(value: Type): void + value(): Type + clear(): void # dirty(): void + cleared(): boolean + isReadOnly(): boolean + <<abstract>> parse(in: String): boolean + toString(): String	

Field Integer
+ FieldInteger(prompt: String) + parse(in: String): boolean

Field None
+ FieldNone(prompt: String) + parse(in: String): boolean + isReadOnly(): boolean

Field Boolean
- <<final>> BOOLEAN_WORD_YES: String = "yes" - <<final>> BOOLEAN_WORD_NO: String = "no" - <<final>> BOOLEAN_WORD_YES: char = 'y' - <<final>> BOOLEAN_WORD_NO: char = 'n'
+ FieldBoolean(prompt: String) + parse(in: String): boolean + toString(): String

Field Option
- _options: String[]
+ FieldOption(prompt: String, options: String[]) + parse(in: String): boolean

Field Real
+ FieldReal(prompt: String) + parse(in: String): boolean

Field String
+ FieldString(prompt: String) + parse(in: String): boolean

pt.technico.utlib.forus



<pre> &lt;&lt;abstract&gt;&gt; Command Receiver: class </pre>	
<pre> - _last: boolean - _title: String # &lt;&lt;final&gt;&gt; - receiver: Receiver # _valid: Predicate&lt;Receiver&gt; </pre>	
<pre> + Command(last: boolean, title: String) + Command(last: boolean, title: String, receiver: Receiver) + Command(last: boolean, title: String, receiver: Receiver, valid: Predicate&lt;Receiver&gt;) + Command(title: String, receiver: Receiver) + Command(title: String, receiver: Receiver, valid: Predicate&lt;Receiver&gt;) + title(): String + isLast(): boolean + isValid(): boolean + addBooleanField(key: String, prompt: String): void + addRealField(key: String, prompt: String): void + addIntegerField(key: String, prompt: String): void + addStringField(key: String, prompt: String): void + addOptionField(key: String, prompt: String, options: String...): void + booleanField(key: String): Boolean + realField(key: String): Double + integerField(key: String): Integer + stringField(key: String): String + optionField(key: String): String + &lt;&lt;final&gt;&gt; performCommand(): void # &lt;&lt;abstract&gt;&gt; execute(): void </pre>	

<pre> CommandException </pre>	
<pre> - &lt;&lt;final&gt;&gt; serialVersionUID: long = 2020101500000000000L </pre>	
<pre> + CommandException(message: String) + CommandException(message: String, throwable: Throwable) + &lt;&lt;final&gt;&gt; toString(): String </pre>	
<pre> DoOpenMenu </pre>	
<pre> + DoOpenMenu(label: String, receiver: Menu) # &lt;&lt;final&gt;&gt; execute(): void </pre>	

AbstractLibMenu

Menu
- title: String
+ Menu(ur: Dialog, title: String, commands: Command<?>...) + Menu(title: String, commands: Command<?>...) + title(): String + size(): int + entry(n: int): Command<?> + entries(): Command<?>[] + open(): void

<< interface >> Message
+ formatInfo(msg: String): String

At demo-urlib.Menu



## Concrete PrintStream

-- errors: boolean

+ <<final>> add(rs: PrintStream): void

+ checkErrors(): boolean

+ close(): void

+ flush(): void

+ print(b: boolean): void

+ print(c: char): void

+ print(c: char[]): void

+ print(d: double): void

+ print(f: float): void

+ print(i: int): void

+ print(l: long): void

+ print(obj: Object): void

+ print(str: String): void

+ println(b: boolean): void

+ println(c: char): void

+ println(c: char[]): void

+ println(d: double): void

+ println(f: float): void

+ println(i: int): void

+ println(l: long): void

+ println(obj: Object): void

#+ println(str: String): void

#+ setErrors(): void

+ write(buf: byte[], off: int, len: int): void

+ write(b: int): void

+ write(b: byte[]): void

## <<interface>>

### Message

+ invalidOption(errors: String): String

+ invalidOption(): String

+ errorEOF(e: EOFException): String

+ errorIO(e: IOException): String

+ errorInvalidNumber(e: NumberFormatException): String

+ errorREOF(e: RuntimeEOFException): String

+ inputError(e: FileNotFoundException): String

+ errorClosingInput(e: IOException): String

+ endOfInput(): String

+ outputError(e: FileNotFoundException): String

+ logError(e: FileNotFoundException): String

## <<interface>>

### Prompt

+ option(): String

+ exit(): String

## <<interface>>

### Property

+ WRITE\_INPUT: String = "writeInput"

+ BOTH\_CHANNELS: String = "both"

+ INPUT\_CHANNEL: String = "in"

+ OUTPUT\_CHANNEL: String = "out"

+ LOG\_CHANNEL: String = "log"

Pl. de caso, utilib. nt



### RuntimeException

- <<Perid>> serialVersionUID: long = 2006102916551

+ RuntimeException()

+ RuntimeException(msg: String)

### TextInteraction

- \_writeInput: boolean

+ TextInteraction()

+ close(): void

+ open(menu: Menu): void

+ fill(focus: Focus): void

+ send(title: String, text: String): void

- readString(prompt: String): String

- readInteger(prompt: String): int

pt. decou write text

