

compositePrintStream

- -Streams: Collection<PrintStream>
- -error: boolean = false
- + CompositePrintStream()
- + add(st: PrintStream): void
- + checkError(): 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(s: 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(s: String): void
- # setError(): void
- + write(buf: byte[], off: int, len: int): void
- + write(b: int): void
- + write(b: byte[]): void

Runtime EOF Exception

- + Runtime EOF Exception()
- + Runtime EOF Exception(msg: String)

~~RuntimeEOFException~~

~~+ RuntimeEOFException()
+ RuntimeEOFException(msg: String)
at: tencoco.pc.ua~~

Command <<abstract>>

- last: boolean
- title: String
receiver: Receiver = null
validityPredicate^{Receiver}: ValidityPredicate = null
form: Form
display: Display
+ Command(last: boolean, title: String)
+ Command(last: boolean, title: String, receiver: Receiver)
+ Command(last: boolean, title: String, receiver: Receiver, ^{predicate} validityPredicate: ValidityPredicate)
+ Command(title: String, receiver: Receiver)
+ Command(title: String, receiver: Receiver, predicate: ValidityPredicate <Receiver>)
+ title(): String
+ islast(): boolean
+ isValid(): boolean
+ perform-command(): void
+ execute(): void <<abstract>>

~~Constants <final>~~

~~+ MAIN_TITLE: String = "Programação com Objectos"
+ BOOLEAN_WORD_YES: String = "sim"
+ BOOLEAN_WORD_NO: String = "nao"
+ BOOLEAN_CHAR_YES: Char = 's'
+ BOOLEAN_CHAR_NO: Char = 'n'
+ REGEX_BOOLEAN: String = "[nS]"
+ REGEX_NUMBER_INTEGER: String = "[0-9]+"
+ REGEX_NUMBER_REAL: String = "[0-9]+\\.[0-9]+[eE][+-][0-9]+"
+ REGEX_STRING: String = ".*"
Constants()~~

Dialog
<ul style="list-style-type: none"> - ACTION_SWING : String = "swing" - ACTION_TEXT : String = "test" + Dialog IO : Dialog = new Dialog() - subsystem : Interaction - Dialog() + Dialog (action : Interaction) + setTitle (title : String) : void + menu (menu : Menu) : void + form (form : Form) : void + message (d : Display) : void + close () : void

DialogException << abstract >>
<ul style="list-style-type: none"> + getMessage () : String << abstract >> + toString () : String

Menu
<ul style="list-style-type: none"> - title : String - commands : Command <? > [] - menu : Dialog + Menu (title : String, title : String, commands : Command <? > []) + Menu (title : String, commands : Command <? > []) + title () : String + size () : int + entry (n : int) : Command <? > + entries () : Command <? > [] + open () : void

Messages << final >>
<ul style="list-style-type: none"> + invalidOperation (error : String) : String + selectAnOption () : String + invalidOption () : String + exit () : String - Messages()

Property Names << final >>
<ul style="list-style-type: none"> + WRITE_INPUT : String = "writeInput" + BOTH_CHANNELS : String = "both" + INPUT_CHANNEL : String = "in" + OUTPUT_CHANNEL : String = "out" + LOG_CHANNEL : String = "log" + ACTION_CHANNEL : String = "ui" - PropertyNames()

Display

- ui: Dialog
- title: String = ""
- text: String = ""

- + Display()
- + Display(title: String)
- + Display(ui: Dialog, title: String)
- + add(text: String): Display
- + addline(text: String): Display
- + addnewline(text: String, force: boolean): Display
- + getTitle(): String
- + getText(): String
- + display(force: boolean): void
- + display(): void
- + popup(top: Object): void

Form

- ui: Dialog
- title: String
- entries: List<Input<?>>

- + Form(title: String)
- + Form(ui: Dialog, title: String)
- + title(): String
- + entries: Collection<Input<?>>
- + entry(i: int): Input<?>
- + add(im: Input): Input<?>
- + addBooleanInput(label: String): ^{Input<boolean>} ~~Input<String>~~
- + addStringInput(label: String): Input<String>
- + addFloatInput(label: String): Input<float>
- + addIntegerInput(label: String): Input<Integer>
- + parse(): void
- + parse(clean: boolean): void
- + clear(): void

<<abstract>>

Input

- prompt: String
- regex: String
- clear: boolean
- # - value: Type

- + prompt(): String
- # Input()
- # Input(prompt: String, regex: String)
- + set(value: Type): void
- + value(): Type
- + clear(): void
- + dirty(): void
- + cleared(): boolean
- + parse(im: String): boolean
- + regex(): String
- + toString(): String

Input Boolean

- + InputBoolean(prompt: String)
- + parse(im: String): boolean
- + toString(): String

Input Float

- + InputFloat(prompt: String)
- + parse(im: String): boolean

Input Integer

- + InputInteger(prompt: String)
- + parse(im: String): boolean

Input None

- + InputNone(prompt: String)
- + parse(im: String): boolean

InputString

- + InputString(prompt: String)
- + parse(im: String): boolean

<< Interface >>

Interaction

```

+ menu (menu: Menu): void
+ form (form: Form): void
+ message (display: Display): void
+ setTitle (title: String): void
+ close(): void

```

InteractionUsingSwing

```

+ InteractionUsingSwing()
+ close(): void
+ menu (m: Menu): void
+ form (f: Form): void
+ message (d: Display): void
- message (s: String, title: String): void

```

SwingPanel

```

- iopt: int
- -end: boolean
- -ims: Map<Input<?>, JTextField>,

+ actionPerformed (evt: ActionEvent): void
+ sleep (m: int): void
+ await(): void
+ option(): int
+ pause(): boolean

SwingPanel (m: Menu)
SwingPanel (f: Form)
SwingPanel (s: String, title: String)

```

InteractionUsingText

```

- -in: BufferedReader
- -out: PrintStream
- -log: PrintStream = NULL
- -writeInput: boolean

+ InteractionUsingText()
+ closeDown(): void
+ menu (m: Menu): void
+ form (f: Form): void
+ message (display: Display): void
+ setTitle (title: String)
+ close(): void
- readInteger (prompt: String): int
- readString (prompt: String): String
- printLn (text: String): void
- print (text: String): void

```

<< abstract >>

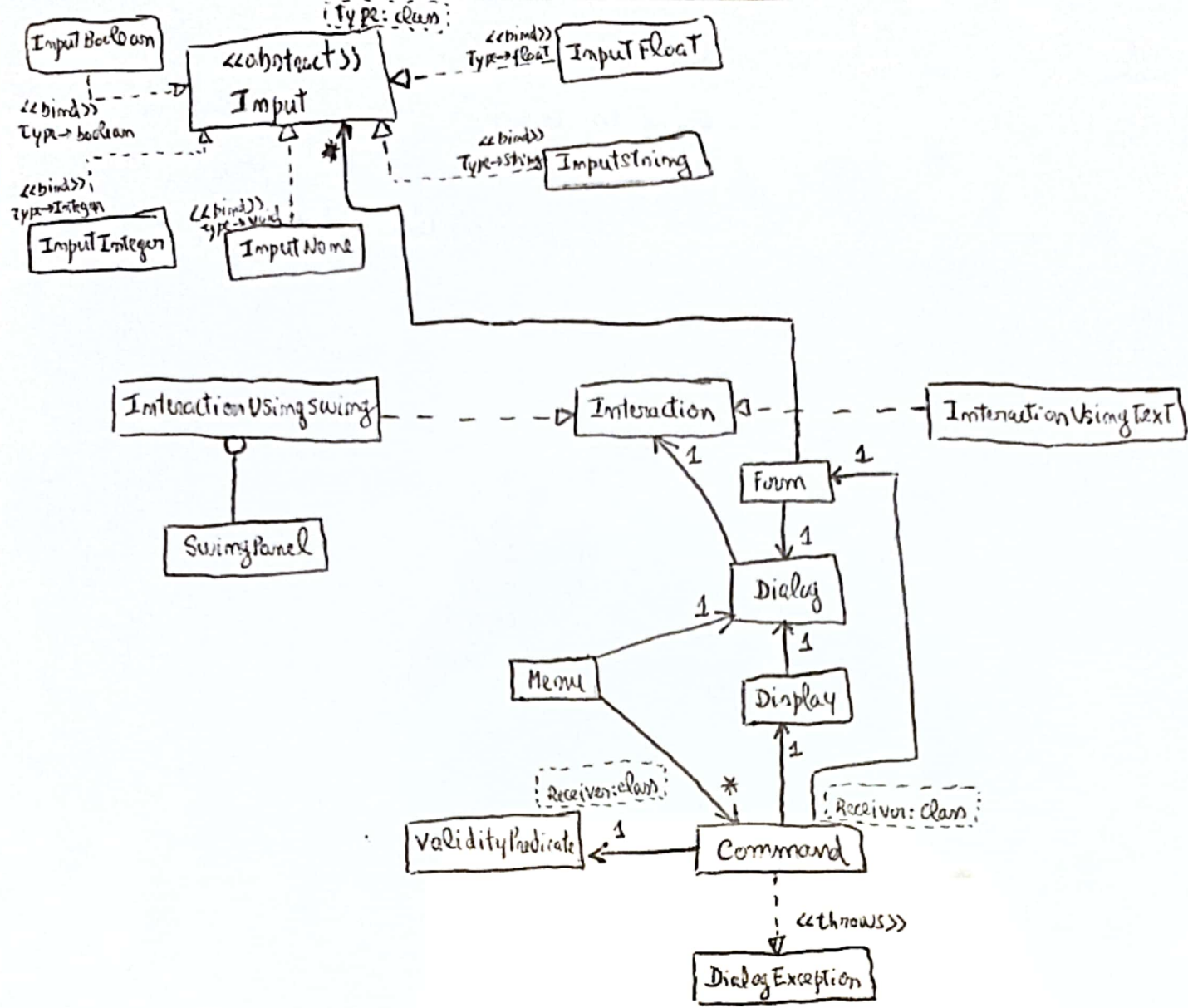
Validity Predicate

```

# - receiver: Receiver

+ Validity Predicate (receiver: Receiver)
+ isValid(): boolean

```



Declaro por minha honra que este diagrama foi realizado apenas pelos elementos que constituem o grupo de projecto.

Eduardo Moreira Miranda

Declaro por minha honra que este diagrama foi realizado apenas pelos elementos que constituem o grupo de projeto.

Rodrigo de Melo Pinto