



EBook Gratuito

APPENDIMENTO

pharo

Free unaffiliated eBook created from
Stack Overflow contributors.

#pharo

Sommario

Di.....	1
Capitolo 1: Iniziare con pharo.....	2
Osservazioni.....	2
Examples.....	2
Installazione o configurazione.....	2
Capitolo 2: Creare una classe.....	3
introduzione.....	3
Examples.....	3
Aggiunta di una classe in un browser di sistema.....	3
Codice di creazione della classe banale.....	4
Classe anonima.....	5
Capitolo 3: Mare.....	6
introduzione.....	6
Examples.....	6
droppable.....	6
Titoli di coda.....	8

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [pharo](#)

It is an unofficial and free pharo ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official pharo.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Capitolo 1: Iniziare con pharo

Osservazioni

Pharo fornisce un modo eccellente per [combattere la complessità](#) nello sviluppo del software. Con il suo rapido feedback con la compilazione istantanea del metodo attualmente modificato, gli oggetti dal compilatore allo stack run-time e gli strumenti progettati per essere personalizzabili dal singolo sviluppatore, supporta di nuovo la programmazione di un pasticcio. Dispone di eccellenti strumenti di refactoring e semplifica la creazione di linguaggi specifici di dominio. Valorizza l'innovazione e l'efficacia rispetto all'efficienza di runtime.

Examples

Installazione o configurazione

Installazione semplice

Vai su <http://pharo.org/download> e seleziona un download adatto ed eseguillo.

Dettagli

Esistono molti modi diversi per installare Pharo. Pharo stesso consiste in un vm e un'immagine. Inoltre ha bisogno di fonti e plugin e ha alcune dipendenze:

- È un ambiente multiplatforma, che funziona su OS-X (e iOS), Windows e diverse varianti UNIX (come Ubuntu e Android).
- Funziona su una macchina virtuale in grado di girare su diverse architetture di processore (Intel, ARM). La macchina virtuale è condivisa con Squeak, Cuis e Newspeak. Con Pharo 5 è stato introdotto un nuovo e molto più veloce vm utilizzando un diverso formato di immagine e FFI.
- Esistono varianti a 32 e 64 bit.
- Oltre all'immagine standard, c'è PharoLauncher che si integra con la nostra infrastruttura CI e supporta il download e l'esecuzione di tutti i tipi di immagini, tra cui seaside pre-costruito, magritte e alce, vecchie versioni e le ultime versioni di sviluppo di Pharo.

Leggi Iniziare con pharo online: <https://riptutorial.com/it/pharo/topic/8799/iniziare-con-pharo>

Capitolo 2: Creare una classe

introduzione

Creare una classe in Pharo è semplice come inviare `subclass:` messaggio ad un altro oggetto di classe. Ma la maggior parte delle classi viene creata e modificata in un browser di sistema (attualmente Nautilus).

Examples

Aggiunta di una classe in un browser di sistema

Il modo più comune per aggiungere (e modificare classi) dal browser di sistema Nel browser del sistema Nautilus non è selezionato nulla o selezionare un pacchetto (prima colonna) o una classe (seconda colonna). A seconda della selezione, l'editor di codice mostrerà un modello di classe leggermente diverso:

Selezione	Modello
Nessuna	Modello di classe vuoto
Pacchetto	Un modello di classe con il nome del pacchetto precompilato (basato sul pacchetto selezionato)
Classe	Una definizione effettiva della classe selezionata

L'immagine seguente mostra la finestra di Nautilus con un pacchetto selezionato:



Scoped

Variables

Type: Pkg1|^Pkg2|Pk.*Core\$

- BlueInk-Extras
- BlueInk-Tests
- CodeExport
- CodeImport**
- CodeImportCommandLi
- Collections-Abstract
- Collections-Arithmetic
- Collections-Atomic
- Collections-DoubleLinke
- Collections-DoubleLinke
- Collections-Native
- Collections-Sequenceab
- Collections-Stack

- ChunkFileFormatParser
- ChunkReadStream
- CodeChunk
- BehaviorOwnedChunk
- ClassCommentChunk
- ClassOrganizationChunk
- MethodChunk
- DoltChunk
- StyleChunk
- CodeImporter
- FileCompilerRequestor



Hier.



Class



```
Object subclass: #NameOfSubclass
  instanceVariableNames: ''
  classVariableNames: ''
  package: 'CodeImport'
```

. Se non hai alcuna superclasse particolare nella tua mente, è consigliabile sottoclasse da `Object` ma questo porta una confusione tautologica nell'esempio precedente. Diciamo che vuoi creare una classe `PriorityStack` con la classe `Stack` come una superclasse, quindi devi valutare:

```
Stack subclass: #PriorityStack
```

Classe anonima

È possibile creare classi senza nomi che non sono installati nel sistema inviando `newAnonymousSubclass` a una classe.

Per esempio

```
anonymousSet := Set newAnonymousSubclass
```

assegnerà una sottoclasse anonima di `Set` alla variabile `anonymousSet`. Quindi puoi compilare i metodi in questa classe e creare un'istanza o scambiarla con una classe reale.

Utile per le risorse di test di proxy

Leggi [Creare una classe online](https://riptutorial.com/it/pharo/topic/8834/creare-una-classe): <https://riptutorial.com/it/pharo/topic/8834/creare-una-classe>

Capitolo 3: Mare

introduzione

Seaside è una struttura web per Pharo e altri piccoli slogan. È ideale per applicazioni complesse con un modello di dominio ricco.

Examples

droppable

Un componente Seaside (sottoclasse di `WAComponent`) deve sovrascrivere `#renderContentOn:`. È una classe `smalltalk` in grado di utilizzare tutti i normali metodi di strutturazione di un'applicazione. Qui delega a tre diversi metodi.

```
JQDroppableFunctionalTest>>renderContentOn: html
  self renderInstructionsOn: html.
  self renderInventoryOn: html.
  self renderSelectedOn: html
```

Come parametro ottiene un oggetto `canvas html` che comprende i messaggi rilevanti per la creazione di `html` e `javascript`. Usa un'interfaccia fluente, dove `#with:` è l'ultimo messaggio inviato al contesto corrente della tela.

```
JQDroppableFunctionalTest>>renderInventoryOn: html
  html div class: 'inventory ui-corner-all'; with: [
    self colors do: [ :each |
      html div
        class: each;
        passenger: each;
        script: (html jQuery new draggable
          revert: true) ] ]

JQDroppableFunctionalTest>>renderSelectedOn: html
  html div
    class: 'selected ui-corner-all';
    script: (html jQuery new droppable
      onDrop: (html jQuery this load
        callbackDroppable: [ :event |
          self selected add: (event at: #draggable) ]);
      html: [ :r | self renderSelectedItemsOn: r ]));
    with: [ self renderSelectedItemsOn: html ]

JQDroppableFunctionalTest>>renderSelectedItemsOn: html
  self selected do: [ :each |
    html div
      class: each;
      passenger: each;
      script: (html jQuery new draggable
        onStop: (html jQuery this effect
          percent: 1; puff;
```



```
onComplete: (html jQuery this parent load html: [ :r |  
  self selected remove: each ifAbsent: [ ].  
  self renderSelectedItemOn: r ]))) ]
```

Leggi Mare online: <https://riptutorial.com/it/pharo/topic/8805/mare>

Titoli di coda

S. No	Capitoli	Contributors
1	Iniziare con pharo	Community , Stephan Eggermont
2	Creare una classe	Uko
3	Mare	Stephan Eggermont