



Commandes Selenium prises en charge



<https://pandorafms.com/manual/!778/>

Permanent link:

https://pandorafms.com/manual/!778/fr/documentation/pandorafms/technical_annexes/17_compatible_selenium_commands
12/03 19:30



Commandes Selenium prises en charge

Liste des commandes prises en charge

Ces commandes sont utilisées dans le « [Contrôle de l'expérience utilisateur UX et WUX](#) ».

Commandes internes

- `run` : Il exécute un *test*.
- `build_transaction` : Il pré-construit une transaction basée sur le passé *test* (interne).
- `get_transaction` : Il renvoie la transaction en cours, toujours utilisée pour récupérer les résultats, l'ordre des phases et les variables.
- `sessions` : Il récupère la liste des sessions présentes dans *hub*.
- `kill_sessions` : Supprimer toutes les sessions existantes dans *hub*.

Commandes supplémentaires

- `extract` : Il combine le code source et le code propre pour l'extraction.
- `storeExtraction` : Il enregistre le résultat de `extract` dans une variable.
- `dispatchEvent` : Il déclenche un événement JavaScript sur un élément donné.
- `takeElementScreenshot` : Il effectue une capture d'écran d'un élément spécifique.
- `phase_start` : Il initie une phase et transforme le test en une transaction WUX.
- `phase_end` : Terminer une phase.
- `clickAndWait` : Il simule le comportement de cette commande Selenium 2 dans Selenium 3.
- `waitForPageToLoad` : Il simule le comportement de cette commande Selenium 2 dans Selenium 3.

Commandes qui génèrent des modules

- `getValue` : Il extrait une valeur.

`getValue;nom_du_module;type_module;REGEX_capture`

- `getVariable` : (Version NG 753 ou ultérieure) Il extrait une valeur spécifique d'une variable.

`getVariable;nom_du_module;type_module;nom_du_variable`

- `getScreenshot` : Il extrait une capture d'écran.

`getScreenshot;nom_du_module`

Toutes les commandes propres doivent apparaître dans le fichier de test (`.side`) sous forme de commentaires. Sinon, le test échouera dans l'extension Selenium.

Commandes fournies par l'extension d'enregistrement (v3)

- `addSelection`
- `answerOnNextPrompt`
- `assert`
- `assertAlert`
- `assertChecked`
- `assertConfirmation`
- `assertEditable`
- `assertElementPresent`
- `assertElementNotPresent`
- `assertNotChecked`
- `assertNotEditable`
- `assertNotSelectedValue`
- `assertNotText`
- `assertPrompt`
- `assertSelectedValue`
- `assertSelectedLabel`
- `assertText`
- `assertTitle`
- `assertValue`
- `check`
- `chooseCancelOnNextConfirmation`
- `chooseCancelOnNextPrompt`
- `chooseOkOnNextConfirmation`
- `click`
- `clickAt`
- `close`
- `doubleClick`
- `doubleClickAt`
- `dragAndDropToObject`
- `echo`
- `editContent`
- `else`
- `elseif`
- `end`
- `executeScript`
- `executeAsyncScript`
- `if`

- mouseDown
- mouseDownAt
- mouseOver
- mouseUp
- mouseUpAt
- open

- pageLoadTimeout
- pause
- removeSelection
- select
- selectFrame
- selectWindow
- sendKeys
- setSpeed
- setWindowSize
- store
- storeAttribute
- storeJson
- storeText
- storeTitle
- storeValue
- storeWindowHandle
- storeXPathCount
- submit
- type
- uncheck
- verify
- verifyChecked
- verifyEditable
- verifyElementPresent
- verifyElementNotPresent
- verifyNotChecked
- verifyNotEditable
- verifyNotSelectedValue
- verifyNotText
- verifySelectedLabel
- verifySelectedValue
- verifyText
- verifyTitle
- verifyValue
- waitForElementEditable
- waitForElementNotEditable
- waitForElementNotPresent
- waitForElementNotVisible
- waitForElementPresent
- waitForElementVisible
- waitForText (*Version NG 752 ou ultérieure*)
- webdriverAnswerOnVisiblePrompt
- webdriverChooseCancelOnVisibleConfirmation
- webdriverChooseOkOnVisibleConfirmation



[Retour à l'index de la documentation de Pandora FMS](#)