

Manuel NewCliTest (2022)

Manuel NewCliTest (2022)

- Présentation
- Lancer les tests de l'application
- Création des tests
- Définition d'un test
 - Test simple sans interaction avec l'utilisateur
 - Test simple avec interaction avec l'utilisateur
 - Test plus complexe avec bloc de définition
- Checks préliminaires
- Annexe
 - Créer l'alias de lancement

Présentation

Les tests **NewCliTest*** permettent de tester des applications en ligne de commande qui utilisent `tty-prompt` pour interagir avec l'utilisateur.

Ou pour tester des applications qui n'interagissent pas du tout avec l'utilisateur.

Lancer les tests de l'application

Note : l'[alias de lancement](#) doit avoir été créé.

Des tests doivent avoir été créés dans le dossier `tests/clitests/`

- ouvrir un Terminal au dossier de l'application,
- jouer la commande `clitests`.

Création des tests

Dans l'application, les tests doivent être placés dans un dossier `tests/clitests/`. C'est là et seulement là que la commande cherchera.

Il faut créer dans ce dossier des fichiers contenant les codes de test.

Définition d'un test

Test simple sans interaction avec l'utilisateur

```
module NewCliTest
  test "Ceci est mon test" do

    # La commande à jouer
    when_run '<commande>'

    # Le résultat attendu
    resultat_is '<le résultat>'

  end
end #/module NewCliTest
```

Par exemple :

```
module NewCliTest

  test "L'addition en mode bash fonctionne" do
    when_run 'expr 2 + 2'
    resultat_is 4
  end

end #/module NewCliTest
```

Remarquer que ces tests doivent impérativement se trouver à l'intérieur d'un bloc `module NewCliTest`.

Test simple avec interaction avec l'utilisateur

```
module NewCliTest

  test "Ceci est mon test" do

    # La commande à jouer
    when_run '<commande>'

    # Les entrées console
    with_inputs [<inputs>]

    # Le résultat attendu
    resultat_is '<le résultat>'

  end

end
```

```
end

end #/module NewCliTest
```

Par exemple :

```
module NewCliTest

test "Test de l'entrée du nom" do
  when_run 'ask_for_name'
  with_inputs ['Phil']
  resultat_is 'Your name is Phil.'
end

end #/module NewCliTest
```

Test plus complexe avec bloc de définition

```
module NewCliTest

test "Ceci est un test plus élaboré" do

  when_run do
    # ... Les choses à faire avant ...
    '<la commande>'
  end

  with_inputs ['...', '...', '...']

  resultat_is do
    # ... les tests à faire ...
    value "Le résultat final"
  end

end

end #/module NewCliTest
```

Checks préliminaires

Si des tests sont à faire avant de jouer le test, on les met dans `pre_check` avec un block. Ce pré-check doit impérativement retourner `true` pour que le test se poursuive.

À l'intérieur, on peut utiliser la formulaire simple suivante qui enregistrera le résultat :

```
condition_vrai || raise("Cette condition est fausse")
```

```
module NewCliTest

test "Un test avec pré-check" do

  # La commande à jouer
  when_run '<commande>'

  # Les entrées console
  with_inputs [<inputs>]

  # Le résultat attendu
  resultat_is `<le résultat>`

  # Le check préliminaire
  pre_check do
    # ... test préliminaire qui doit retourner true
    x + y == z ||
      raise("x + y devrait être égal à z (il vaut #{x+y})")

    # Ou mieux :
    Should.equal(x + y, z, "#{x} + #{y}")
  end
end

end #/module NewCliTest
```

Annexe

Créer l'alias de lancement

```
ln -s /path/to/NewCliTest/run_tests /usr/local/bin/clitests
```

Par exemple, si le dossier `NewCliTest` a été placé dans un dossier `Programmes` du `home` :

```
ln -s ~/Programmes/NewCliTest/run_tests /usr/local/bin/clitests
```

