

Розділ 2

Початок роботи з Pharo

У цьому розділі ви навчитеся запускати Pharo, дізнаєтеся, які файли входять до системи Pharo, та яке їхнє призначення, вивчите різні способи взаємодії з середовищем програмування, ознайомитеся з деякими основними інструментами. У розділі описано, по-перше, як встановити Pharo; по-друге, як запустити Pharo. Ви також матимете нагоду запустити інтерактивний посібник *ProfSteph*, який ознайомить вас з основами синтаксису, допоможе визначити новий метод, створити об'єкт і надіслати йому повідомлення.

2.1. Встановлення Pharo

Застосунок Pharo створено для автономного використання. Він чудово працює без підтримки додаткових бібліотек, тому для запуску Pharo не потрібно нічого інсталиувати у вашій системі. Як буде пояснено пізніше, Pharo складається з віртуальної машини (VM), образу, опису колекції змін і колекції фрагментів програмного коду. Встановити і налаштувати їх на вашому комп'ютері можна кількома різними способами.

Pharo Launcher

Pharo Launcher – це багатоплатформний застосунок, який полегшує управління кількома образами та віртуальними машинами Pharo. Він доступний для безкоштовного завантаження з <http://pharo.org/download>. Оберіть на сайті кнопку вашої операційної системи, щоб завантажити відповідний Pharo Launcher. Він містить все необхідне для запуску Pharo. Докладний опис його використання наведено в параграфі 2.3.

Скрипти автоматичного встановлення

Замість Pharo Launcher для завантаження конкретних версій Pharo можна використовувати скрипти, розміщені на <https://get.pharo.org>. Це справді зручно для автоматизації процесу встановлення Pharo.

Для завантаження найновішої версії системи Pharo 9.0 використовуйте скрипт

```
wget -O- get.pharo.org/90+vm | bash
```

Щоб запустити її на виконання, введіть команду

```
./pharo-ui Pharo.image
```

Станом на нині наведені скрипти працюють в операційних системах macOS і Linux.

2.2. Файли компонентів Pharo

Pharo складається з чотирьох головних файлів. Під час читання цієї книги ви не будете взаємодіяти з ними безпосередньо, проте корисно розуміти їхнє призначення.

1. *Віртуальна машина (ВМ)* – єдиний компонент Pharo, що відрізняється для кожної операційної системи. ВМ є виконавчою системою, подібною до віртуальної машини Java. Кожного разу, коли користувач компілює фрагмент тексту програми, компілятор генерує байт-код Smalltalk. ВМ отримує такий байт-код, перетворює його на команди процесора і виконує їх. Pharo постачається з Cog VM, з дуже швидкою JIT-машиною (JIT – just in time, машина, що компілює на льоту).

Виконуваний файл віртуальної машини називають:

- *Pharo.exe* для Windows;
- *pharo* для Linux;
- *Pharo* для OS X (всередині пакета його також називають *Pharo.app*).

Інші перелічені компоненти можна переносити між операційними системами. Їх можна копіювати і запускати на відповідній ВМ.

2. *Образ системи* є миттєвим знімком стану працюючого Pharo. Він міститься у *.image* файлі, що має крос-платформний формат: образ Pharo з однієї операційної системи можна використовувати в будь-якій іншій з відповідною ВМ. Файл образу зберігає стан усіх об'єктів системи, актуальний на певний момент часу, включно з класами та методами, оскільки вони також об'єкти. Образ є контейнером віртуальних об'єктів.

Файл образу називають відповідно до версії системи, наприклад, *Pharo9.0.image* містить образ Pharo 9.0. Він синхронізований з файлом змін *Pharo9.0.changes*.

3. *Файл змін* зберігає протокол усіх модифікацій вихідного тексту оголошення класів, зокрема, всіх зроблених вами змін під час сеансу програмування у Pharo. Кожна версія середовища постачається з майже порожнім файлом змін, що називається відповідно до версії, наприклад, *Pharo9.0.changes*. Цей файл містить історію змін кожного методу, яку можна використати для відстеження відмінностей або скасування внесених змін. Це означає, що ви зможете відновити всі запрограмовані вами методи з *.changes* файлу навіть тоді, коли образ системи не вдалося зберегти через зависання чи через власну забудькуватість. Файл змін пов'язаний з файлом образу, вони завжди працюють разом. Краще не чіпайте їх. Навіть при тому, що Pharo може працювати без них, ви ризикуєте втратити всю свою роботу, якщо не збережете її за допомогою Git.
4. У *.source* файлі зберігається вихідний код усіх частин системи, що змінюється нечасто. Цей файл важливий, оскільки файл образу зберігає лише об'єкти з компільованими методами та їхніми байт-кодами і не містить вихідних текстів. Зазвичай source-файл генерують один раз під час випуску нової версії Pharo. Для Pharo 9.0 він називається *Pharo9.0.sources*.

Пара Image/Change

Файли образу та змін (*.image* та *.changes* файли) з інсталяційного архіву є початковою точкою живого середовища програмування, яке ви пристосовуватимете для власних потреб. Коли будете працювати з Pharo, ці файли зазнаватимуть змін, тому переконайтеся, що вони розташовані в каталозі, в якому ваша операційна система не забороняє вам записувати. Було б добре також вилучити їх зі списку перевірки своєї антивірусної програми. Завжди зберігайте ці два файли разом, ніколи не перейменовуйте їх і не редагуйте за допомогою редактора текстів, бо *.image* файл зберігає стан пам'яті етапу

виконання об'єктів вашої системи, пов'язаний з відповідними місцями *.changes* файлу задля доступу до вихідного коду. Було б добре зберігати запасну копію завантажених *.image* та *.changes* файлів. Тоді ви завжди зможете розпочати роботу з самого початку і повторно завантажити створений вами код. Проте найефективнішим способом зберігання коду є використання системи контролю версій Iceberg разом з Git, яка забезпечує легке зберігання коду в сховищі та відстежування змін.

Загальне налаштування

Ви можете помістити описані чотири головні компоненти Pharo до одного каталогу, проте частіше віртуальну машину та *.source* файл розташовують в окремому каталозі і надають до нього доступ тільки для читання усім користувачам.

Оберіть будь-який зручний для вас та ефективний для вашої операційної системи варіант. Якщо ви тільки починаєте працювати з Pharo, то ліпше використовувати Pharo Launcher, оскільки він все налаштує сам.

2.3. Запуск Pharo за допомогою Pharo Launcher

Pharo Launcher – це інструмент, який допомагає завантажувати та впорядковувати образи Pharo. Він дуже зручний для отримання нових версій Pharo та для оновлення наявних (для виправлення виявлених помилок). Також він надає вам доступ до попередньо сконфігурованих образів зі спеціальними бібліотеками, що звільняє від ручного встановлення та налаштування таких бібліотек.

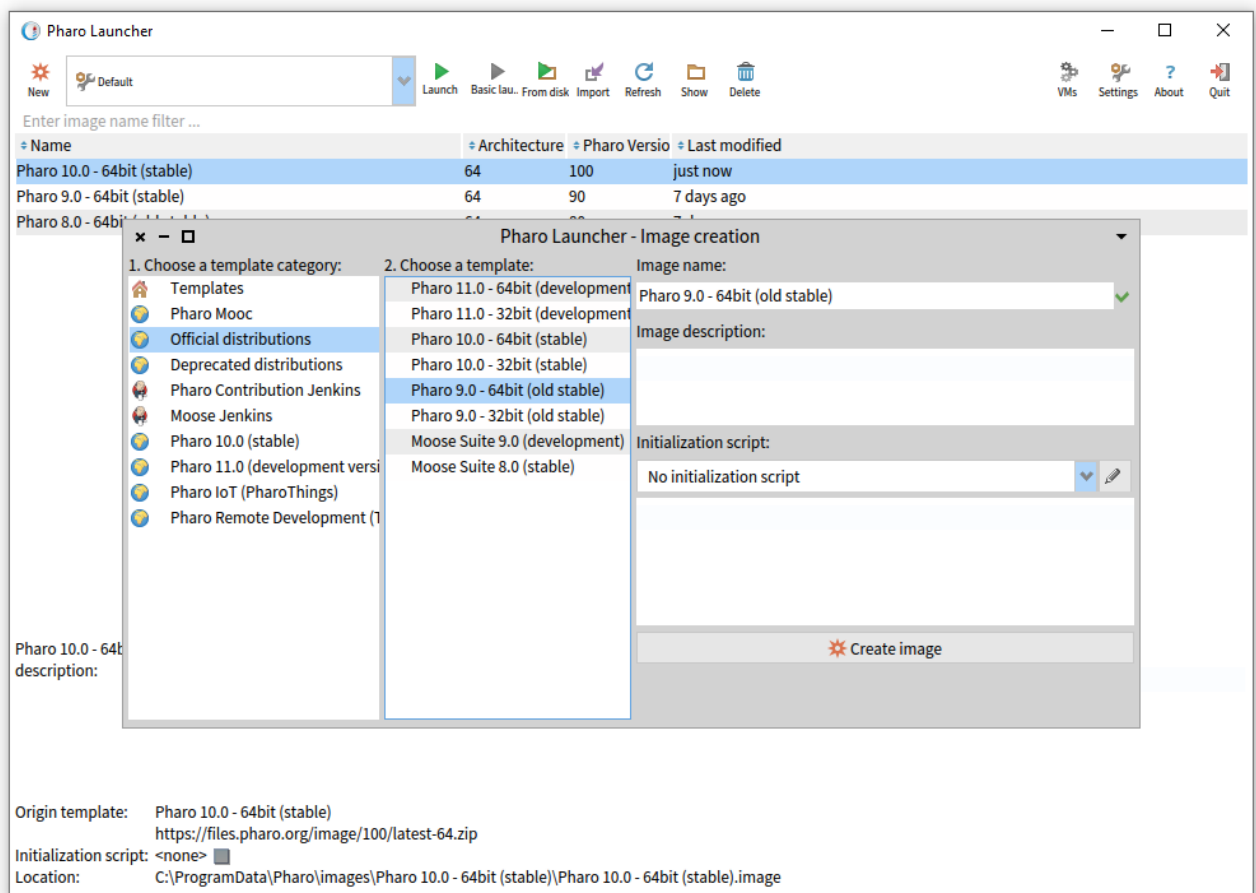


Рис. 2.1. Pharo Launcher – найлегший спосіб керувати образами Pharo

Pharo Launcher можна завантажити з <https://pharo.org/download>. Тут розміщено стислі пояснення щодо завантаження і посилання на версії додатку для різних платформ. Докладні інструкції щодо завантаження, налаштування та використання Pharo Launcher можна знайти на <https://pharo-project.github.io/pharo-launcher/>.

Після встановлення та запуску Pharo Launcher матиме вигляд, подібний до зображеного на рис. 2.1. Щоб завантажити з сервера новий образ Pharo, потрібно клацнути кнопку **New** у лівому верхньому куті вікна. Саме вона відкриває діалог, зображений поверх вікна додатку на рис. 2.1.

Від перекладача. Pharo Launcher завантажує файли образу, змін та вихідного коду і віртуальну машину в папки, вказані в його налаштуваннях. За замовчуванням, вони прив'язані до імені користувача ОС. Наприклад, образ буде завантажено до папки `|Users|ActualUserName|Documents|Pharo|images`, де *ActualUserName* – ім'я користувача. Якщо ваше ім'я задано кирилицею, то Pharo Launcher може мати проблеми з розпізнаванням шляху до завантажених файлів, тому налаштування за замовчуванням варто змінити.



Вгорі праворуч у Pharo Launcher є кнопка *Settings*, що відкриває вікно налаштувань. Місце розташування папок задають у вкладці «*Pharo Launcher*», у пунктах «*Location of template sources file*», «*Location of your images*» тощо. Зазначте тут повний шлях до зручної для вас папки так, щоб він не містив кирилических літер. Налаштування за замовчуванням варто змінити також тоді, коли ви вважаєте за недоречне зберігати Pharo в папці Documents.

Під час завантаження Pharo Launcher будьте готові прийняти архів розміром майже 56 Мб, який можна вилучити згодом після встановлення застосунку. Розгорнутий застосунок займе близько 140 Мб. Це досить відчутні затрати дискового простору, але воно того варте, адже Launcher вміє завантажувати образи та віртуальні машини, стежить за оновленнями, повідомляє вас про всі новинки.

У вікні діалогу *Image creation* оберіть потрібний вам образ (стабільну версію) та натисніть кнопку *Create image* – Launcher завантажить і розпакує відповідні файли. Для них потрібно приблизно 100 Мб дискового простору. Перед першим запуском завантаженого образу Launcher завантажить відповідну віртуальну машину. Вона займе ще майже 30 Мб. Архів віртуальної машини застосунок не видаляє, можна зробити це власноруч.

Вікно Pharo Launcher відображає список образів Pharo, що зберігаються локально на вашій машині (зазвичай у спільній системній папці). Ви одразу можете запустити будь-який з них: двічі клацніть на ньому, або позначте і натисніть кнопку **Launch**. Контекстне меню, яке викликають правою кнопкою мишки, містить кілька корисних команд: копіювання та перейменування образів, відшукування їхнього розташування у файловій системі тощо.

Окрім образів, завантажених із сервера, ви можете за допомогою Pharo Launcher запускати власні образи системи, збережені чи перейменовані під час використання Pharo. Для цього імпортуйте в Launcher відповідну пару *.image* та *.change* файлів.

2.4. Запуск Pharo з командного рядка

Якщо ви використовуєте автономну версію Pharo, то запускайте його так, як це зазвичай роблять у вашій операційній системі: перетягніть файл *.image* на значок віртуальної машини, або двічі клацніть по значку *.image* файлу, або надрукуйте в командному рядку ім'я віртуальної машини і слідом – повне ім'я *.image* файлу.

Запуск Pharo з командного рядка

- У системі macOS двічі клацніть пакунок (bundle) *Pharo9.0.app* у розархівованому завантаженні.
- У системі Linux двічі клацніть (або запустіть з командного рядка) виконуваний bash-скрипт *pharo* з розархівованої папки Pharo.
- У системі Windows зайдіть у розархівовану папку Pharo і двічі клацніть *Pharo.exe*.

Загалом Pharo намагається «робити правильні речі». Якщо ви двічі клацнете на ВМ, то віртуальна машина шукатиме у своїй папці файл образу, щоб завантажити його. Pharo дає змогу зберігати образ у файлі з довільним іменем. Це зручно, коли ви провадите різні розробки: кожному проєкту відповідатиме свій образ. Отож запущена ВМ може знайти кілька *.image* файлів. У цьому випадку машина розпочинає діалог вибору файлу, щоб користувач міг вказати потрібний йому образ. Якщо ви двічі клацнете на *.image* файлі, то операційна система спробує запустити відповідну ВМ, щоб відкрити його. Можливо, вам доведеться допомогти їй у цьому.

Якщо на вашому комп'ютері інстальовано кілька віртуальних машин, то операційна система може вибрати не ту. У цьому випадку надійніше перетягти файл образу на значок потрібної віртуальної машини чи скористатися командним рядком.

Загальна схема команди запуску Pharo з консолі має вигляд:

```
<Pharo executable> <path to Pharo image>
```

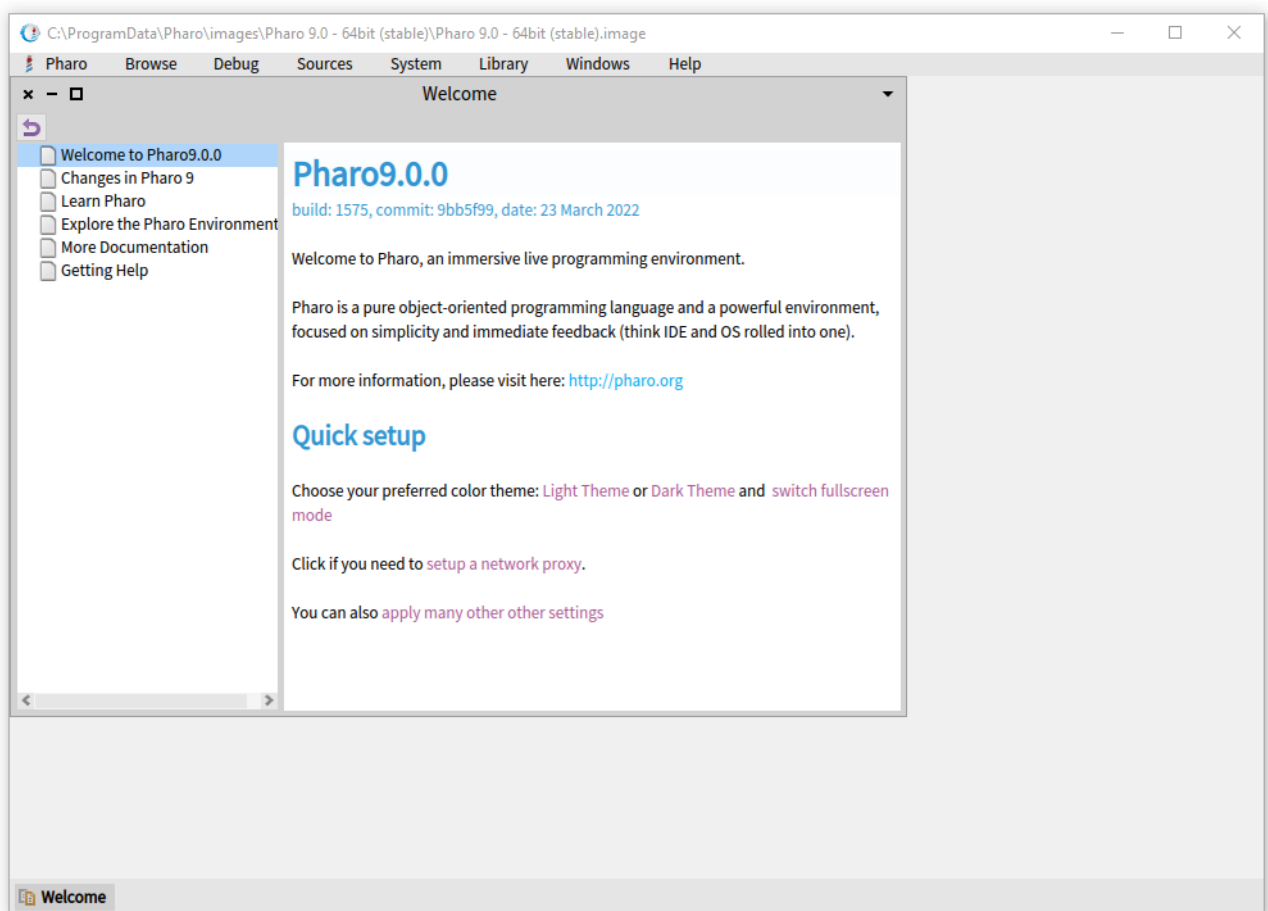


Рис. 2.2. Вікно Pharo після першого запуску

Командний рядок Linux

Припустимо, що ми перебуваємо в папці з файлами Pharo, створеній скриптом автоматичного встановлення.

```
./pharo Pharo.image
```

Командний рядок macOS

Припустимо знову, що Pharo встановили за допомогою скрипту автоматичного встановлення.

```
./pharo Pharo.image
```

Командний рядок Windows

Ми вже зазначали, що скрипти автоматичного встановлення не працюють у Windows, тому ви мусите власноруч завантажити відповідні файли ВМ та образу. Проте запуск також є дуже простим.

```
Pharo.exe Pharo.image
```

Після вдалого запуску ви мали б побачити вікно як на рис. 2.2.

2.5. Зберігання, завершення та повторний запуск сесії Pharo

Вийти з Pharo можна будь-коли. Для цього просто закрийте його вікно так само, як закриваєте інші застосунки. Крім того, можна використати одну з команд *Pharo/Save and quit* або *Pharo/Quit* меню застосунку чи Головного меню Pharo (World Menu), яке з'являється після клацання довільною кнопкою на вільній частині вікна застосунку.

Якщо ви обрали звичайне завершення, то Pharo перепитає вас, чи хочете зберегти образ системи. Якщо ви збережете образ Pharo, то після наступного запуску побачите його *точно в тому стані*, в якому завершили роботу: усі запущені програми, відкриті вікна в тих самих координатах. Так відбувається тому, що файл образу зберігає *всі* об'єкти (редагований текст, розташування вікон, додані методи чи класи – адже *всі* вони є об'єктами), які Pharo завантажує у пам'ять віртуальної машини. Так при виході з Pharo *нічого* не втрачається.

Коли ви запускаєте Pharo вперше, ВМ завантажує вказаний файл образу. Він містить миттєвий знімок багатьох об'єктів, включно з величезним обсягом відкомпільованого коду й інструментами програмування (кожен з яких є об'єктом). Під час роботи з Pharo ви надсилатимете повідомлення цим об'єктам, створюватимете нові об'єкти (за допомогою надсилання повідомлень об'єктам!), деякі з них переставатимуть існувати, а їхня пам'ять буде використана повторно після автоматичного збирання сміття.

Закриваючи Pharo, ви зберігатимете знімок усіх своїх об'єктів. У результаті успішного збереження попередній *.image* файл буде перезаписано новим образом системи. Звичайно, ви можете зберегти образ у файлі з новим іменем.

Як ми вже говорили, *.image* файл працює в парі з *.changes* файлом. Коли ви зберігаєте образ системи, *.image* файл оновлюється миттєвим знімком вашого запущеного Pharo, а *.changes* файл доповнюється журналом усіх змін вихідного коду, зроблених вами після збереження попереднього образу. У більшості випадків ви можете взагалі не

турбуватися про цей файл. Проте, як ми скоро побачимо, *.changes* файл може бути дуже корисним для виправлення помилок чи для відновлення втрачених змін. Але про це – згодом!

Може скластися враження, що образ є ключовим механізмом для зберігання програмних проєктів і керування ними, але це не так. Незабаром ми ознайомимося з набагато кращим засобом для керування кодом та організації командної розробки програмного забезпечення. Образи дуже корисні, але ви звикнете не турбуватися особливо про створення і вилучення образів, оскільки засоби контролю версій, наприклад такі, як менеджер пакетів Iceberg, пропонують набагато ліпші способи керування версіями та поширення коду програмного забезпечення серед команди розробників. Крім того, якщо вам потрібно зберігати об'єкти, можете використати кілька спеціальних систем: *Fuel* (швидкий бінарний серіалізатор об'єктів), або *STON* (текстовий серіалізатор об'єктів), або, навіть, базу даних.

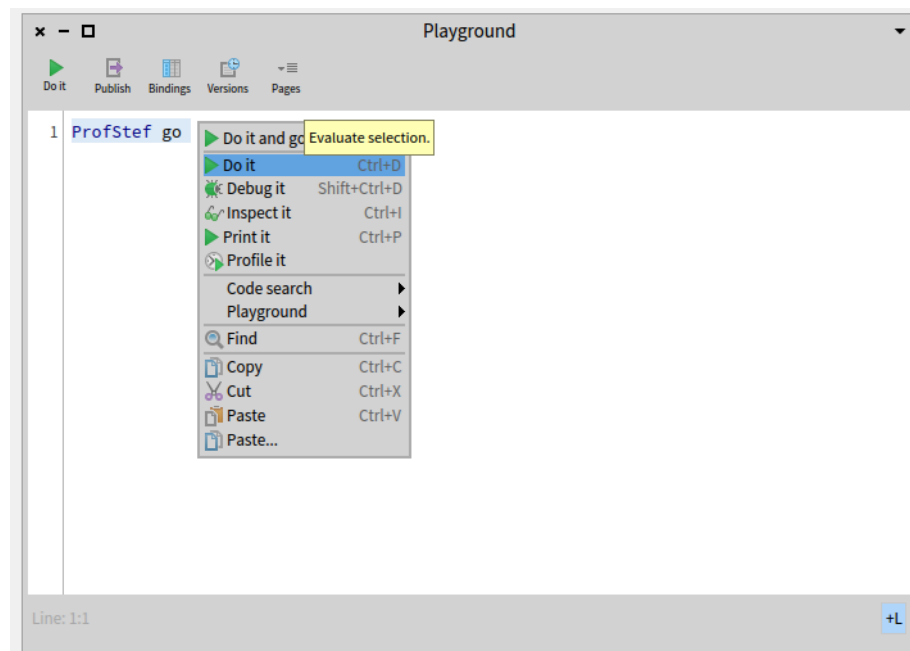


Рис. 2.3. Виконати вираз можна командою «Do it» контекстного меню

2.6. Для швидких і нетерплячих

Іноді може видатися нудним читання довгих розділів про систему, наділену настільки багатьма інструментами, як Pharo. Якщо на вашому комп'ютері уже встановлена робоча система Pharo, і ви трохи уявляєте, як все працює, можете захотіти вивчити основи синтаксису за допомогою *ProfStef*, а потім перейти до наступного розділу – посібника для визначення класу простого лічильника і пов'язаних з ним модульних тестів. Ви також можете переглянути відповідне відео на <http://mooc.pharo.org>.

Щоб запустити *ProfStef*, відкрийте вікно *Playground* відповідною командою розділу *Browse* головного меню, надрукуйте в ньому вираз

```
ProfStef go
```

і виконайте його командою «Do it», як показано на рис. 2.3.

Не хвилюйтеся, якщо все це для вас нічого не означає – скоро зрозумієте. Просто читайте далі...

Цей вираз запустить навчальну програму *ProfStef* (рис. 2.4) – чудовий посібник, щоб почати вивчення синтаксису Pharo.

Вітаємо, ви щойно надіслали ваше перше повідомлення! Pharo опирається на концепцію надсилання повідомлень об'єктам. Об'єкти Pharo схожі на солдатів, готових одразу виконати ваш наказ, якщо тільки вони його зрозуміють. Трохи згодом побачимо, як саме об'єкти розпізнають повідомлення.

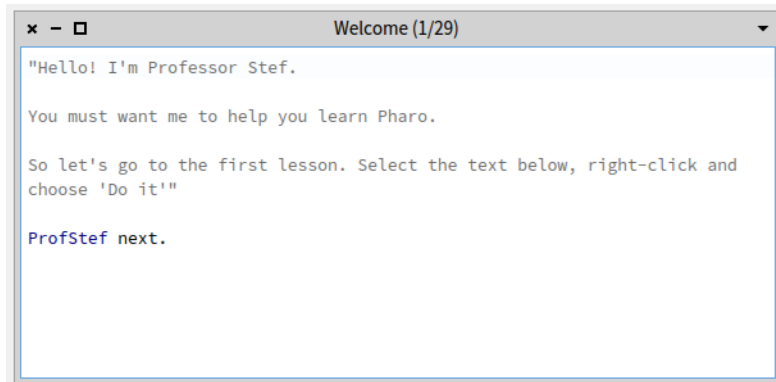


Рис. 2.4. *ProfStef* – простий інтерактивний посібник для вивчення синтаксису Pharo

2.7. Підсумки розділу

- Керувати системами Pharo на вашому комп'ютері можна за допомогою застосунку Pharo Launcher. Завантажити Pharo можна також за допомогою скриптів автоматичного встановлення.
- Робоча система Pharo складається з віртуальної машини, файлу вихідного коду (*.sources*), файлу образу (*.image*) і файлу змін (*.changes*). Останні два з них змінюються, коли ви зберігаєте образ – миттєвий знімок запущеної системи.
- Коли ви відкриваєте образ Pharo, отримуєте середовище в тому ж стані (тобто з тими ж працюючими об'єктами), у якому ви востаннє зберегли образ системи.