

## Arbeitsblatt: PSPP

Name:

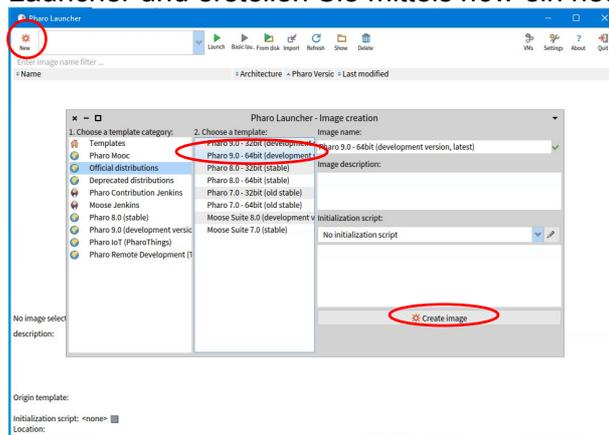
Kurznamen:

## Programmierung in Smalltalk

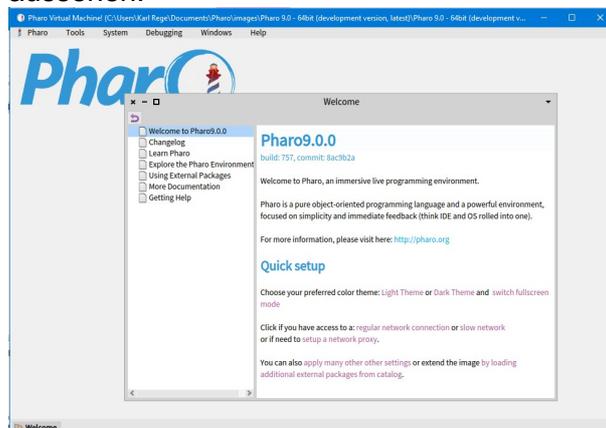
### Installation der Pharo Umgebung

Installieren Sie sich zuerst die Pharo Umgebung.

Variante 1: installieren Sie Pharo über <https://pharo.org/> Seite. Starten Sie den Launcher und erstellen Sie mittels *new* ein neues Image, z.B. Pharo 9.0 64 Bit.



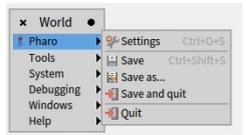
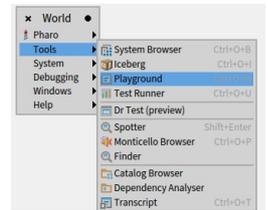
Danach kann Pharo über den Launcher gestartet werden und sollte wie unten aussehen.



Variante 2: Entpacken Sie das Pharo6.ZIP File in ein neues Verzeichnis und starten Sie es mit pharo.exe (ohne Launcher)

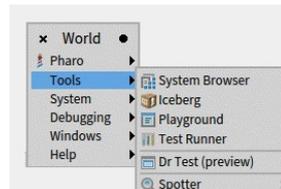
## Aufgabe 1 Kennenlernen der Umgebung

- Öffnen Sie ein neues Playground Fenster (*Linksklick* auf Desktop)
- Öffnen Sie ein Transcript Fenster
- Schreiben Sie Hello World auf das Transcript Fenster
- Speichern Sie den Zustand des Desktops und laden Sie in zurück
- Führen Sie ein paar einfache Berechnungen aus dem Skript von Hand aus um mit dem System vertraut zu werden

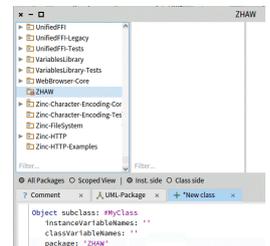


## Aufgabe 2 Erstellen einer neuen Klasse

1. Starten Sie dafür den System Browser

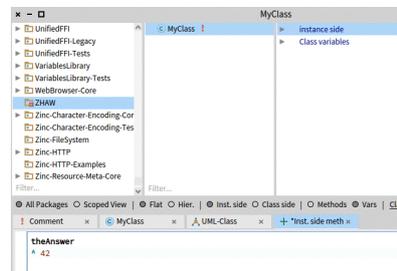
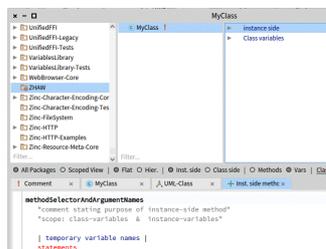


2. Wählen Sie eine beliebige neue Klasse aus und schreiben Sie die gewünschten Änderungen ins untere Fenster. Wählen Sie als package ZHAW. Mit `<ctrl> s` wird die Klasse gespeichert (Veränderte Klassen erkennen Sie am Orangen Dreieck oben rechts).

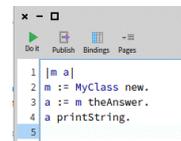


3. Wählen Sie InstanceSide aus und schreiben Sie die Methode:

z.B. `theAnswer` ergibt 42.



4. Instanzieren Sie das Objekt im Playground und rufen Sie die neue Methode auf



### Aufgabe 3

Schreiben Sie ein Smalltalk Programm, das  $\pi$  mittels der früher kennengelernten Monte Carlo Methode bestimmt. Die Berechnung soll direkt im Playground durchgeführt werden.

#### Hinweis

- Klasse Random verwenden um Zufallszahlen zwischen 0 und 1 zu erzeugen.
- Falls Sie eine Endlosschleife programmieren, dann reagiert das Smalltalk System nicht mehr; speichern Sie Ihren Desktop von Zeit zu Zeit ab.

#### Abgabe:

Praktikum: **PS14.3**

Filename: **montecarlo.st**

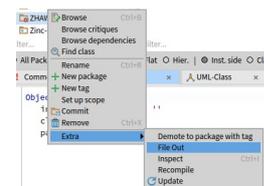
### Aufgabe 4

Schreiben Sie eine Klasse Montecarlo mit folgenden Methoden

- count: n legt die Anzahl Schritte für die Berechnung fest.
- pi liefert die Näherung für  $\pi$

#### Hinweis:

- Schreiben Sie das File via System Browser → File Out



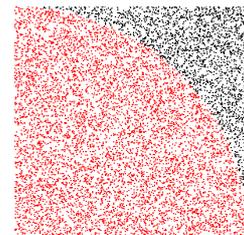
#### Abgabe:

Praktikum: **PS14.4**

Filename: **montecarlo.st**

### Aufgabe 5

Zeichnen Sie die folgende Graphik. Verwenden Sie dafür die Klasse CanvasMorph, die aber vorher noch ins System geladen werden muss.



#### Hinweis:

- CanvasMorph.st via Tools → File Browser → FileIn ins System laden

#### Abgabe:

Praktikum: **PS14.5**

Filename: **montecarlo.st**