

```

pi@pi-zero:~$ jshell
Welcome to JShell -- Version 17.0.6
For an introduction type: /help intro

jshell> Properties properties = System.getProperties();
properties --> {java.specification.version=17, sun.management.co... , java.class.version=61.0}
java.specification.version:17
sun.management.compiler:HotSpot Client Compiler
sun.jnu.encoding:UTF-8
java.runtime.version:17.0.6+10-Raspbian-ldeb11u1rpt1
java.class.path:.
user.name:pi
java.vm.vendor:Raspbian
path.separator:
sun.arch.data.model:32
os.version:6.1.15+
java.runtime.name:OpenJDK Runtime Environment
file.encoding:UTF-8
java.vendor.url:unknown
java.vm.name:OpenJDK Client VM
java.vm.specification.version:17
os.name:Linux
  
```

# JShell

## Befehle

- /help: Anzeige der verfügbaren Befehle und deren Funktionen
- /open: Öffnen einer gespeicherten Sitzungsdatei
- /save: Speichern der aktuellen Sitzung in einer Datei
- /exit: Beenden der Java Shell
- /methods: Anzeige der verfügbaren Methoden für ein Objekt oder eine Klasse
- /imports: Anzeige der importierten Klassen und Pakete
- /edit: Bearbeiten von vorherigen Eingaben
- /reset: Zurücksetzen der Java Shell auf den Ausgangszustand
- /vars: Anzeige der definierten Variablen

## Verwendung

- Testen von Codefragmenten und Konzepten
- Exploratives Programmieren und Prototyping
- Erkunden von Klassen und Bibliotheken
- Verstehen des Verhaltens von Java-Konstrukten
- Unterstützung beim Lernen der Programmierung und Verbesserung der Fähigkeiten

## Nachteile ❌

- Nicht für die Entwicklung umfangreicher Projekte geeignet
- Begrenzte Unterstützung für fortgeschrittene IDE-Funktionen
- Keine Integration von Build-Tools oder Versionskontrolle
- Begrenzte Unterstützung für komplexe Debugging-Szenarien
- Fehlende Möglichkeit zur Kompilierung und Erstellung von ausführbaren JAR-Dateien
- Möglicher Mangel an Produktionsumgebungserfahrung

## Java Shell

- Read-Eval-Print Loop (REPL)-Tool
- ab Java 9 standardmäßig in die Java-Entwicklungsumgebung (JDK) integriert
- Ergebnisse unmittelbar sichtbar
- automatische Vervollständigungsfunktion
- Code aus externen Dateien zu importieren und zu verwenden
- interaktive Shell
- Java-Code in Echtzeit auszuführen
- History

## + Vorteile

- Interaktive und experimentelle Umgebung
- Schnelle Ausführung von Java-Codezeilen
- Sofortige Anzeige von Ergebnissen und Fehlermeldungen
- Einfache Überprüfung von Codefragmenten und Prototypen
- Unterstützt automatische Codevervollständigung
- Zugriff auf vorherige Eingaben für iterative Entwicklung
- Import und Nutzung externer Dateien und Bibliotheken
- Unterstützt das Lernen und Verständnis der Java-Sprache
- Beschleunigt den Entwicklungszyklus
- Fehlerbehebung durch sofortiges Feedback

