

Algorithmen und Datenstrukturen

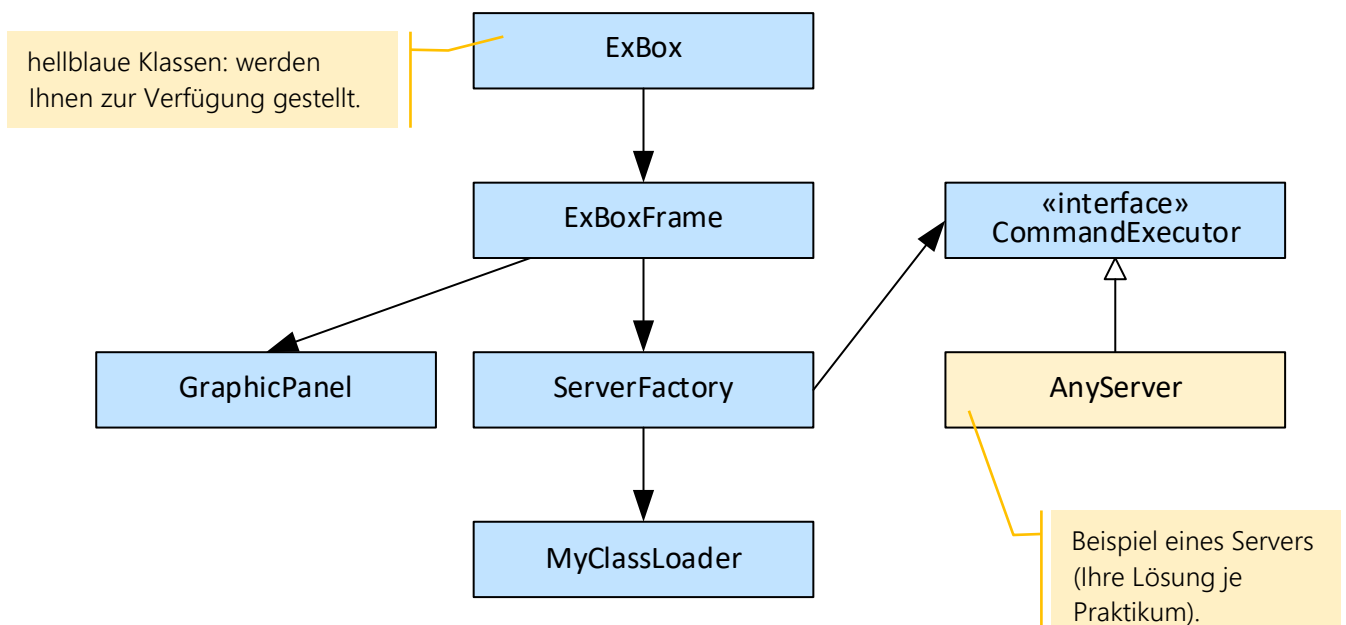
Der Experimentierkasten

In diesem Semester werden wir eine Reihe von Algorithmen kennenlernen und mit diesen Experimente durchführen. Dafür benötigen wir einen geeigneten «Experimentierkasten». Dieser soll in einem sogenannten Client-Server-Arrangement realisiert werden:

- Client: Der Client stellt die Benutzerschnittstelle zur Verfügung. Er wird Ihnen von uns zur Verfügung gestellt (ExBox.java, ExBoxFrame.java, ServerFactory.java, MyClassLoader.java, CommandExecutor.java und GraficPanel.java). Diese werden Sie in jedem Praktikum wieder verwenden.
- Server: der Server enthält jeweils die Experimente, die Implementierung Ihrer Lösung der Aufgaben. Sie erhalten als Beispiel/Vorlage den AnyServer (AnyServer.java), dieser stellt einen Beispiel-Server dar, welcher das Interface CommandExecutor implementieren muss.

Der Client ist so entworfen worden, dass beliebige Server auch nachträglich (während der Laufzeit) hinzugefügt werden können (wir werden in den folgenden Aufgaben immer wieder neue Server schreiben). Dabei muss der Client lediglich wissen, wie der Server aufgerufen werden kann, er muss also die Server-Schnittstelle kennen. Die Schnittstelle wird als Interface (das CommandExecutor-Interface) realisiert, welche vom Server (Bsp. AnyServer) implementiert werden muss.

Zusätzlich müssen wir zur Laufzeit Server nachladen, bzw. Klassen instanzieren können, von denen wir lediglich den Namen (als String) kennen. Für diesen Zweck gibt es die Klasse ServerFactory, mit der Objekte beliebiger Klassen, die den CommandExecutor implementieren, instanziiert werden können.

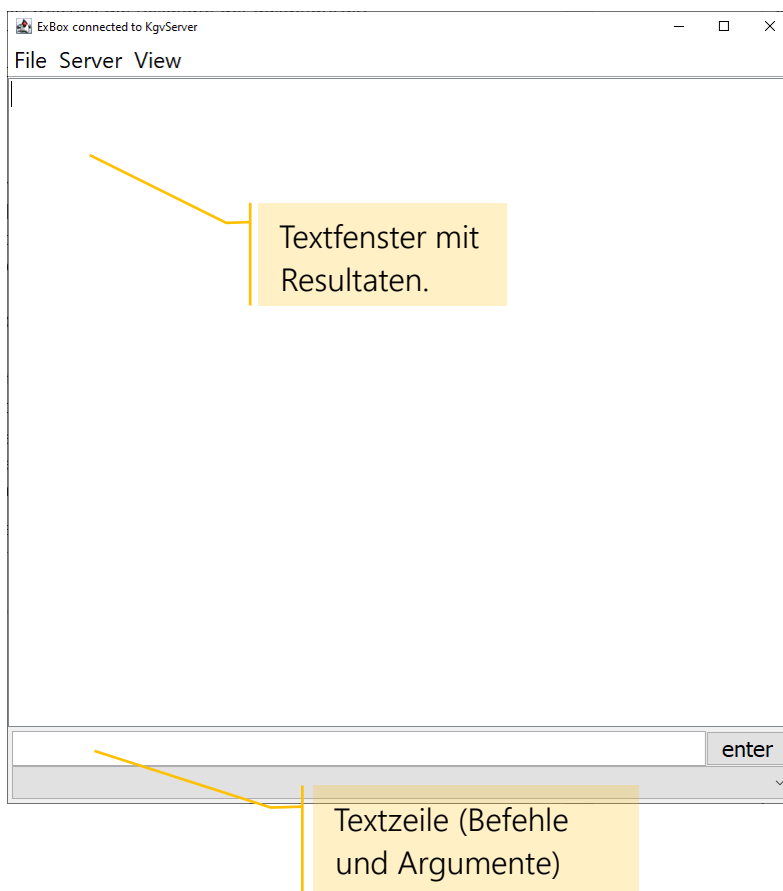


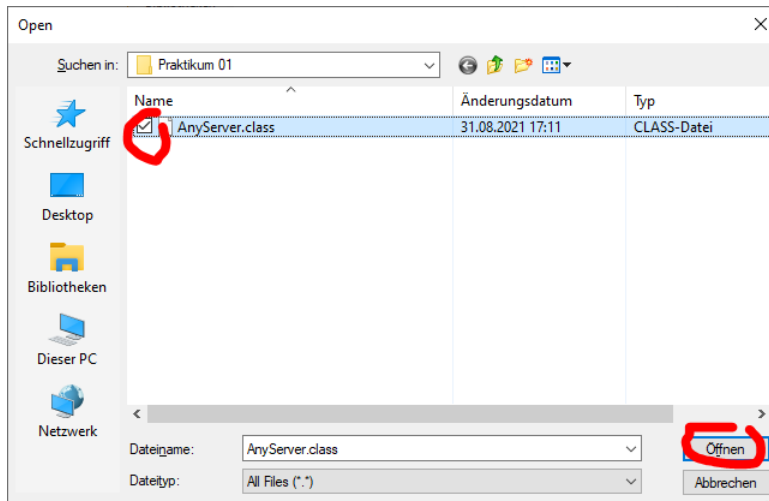
Beschreibung der Funktionalität

Das Hauptfenster des GUI-Clients enthält zwei Bereiche: 1) eine Textzeile im unteren Bereich für die Eingabe der Befehle und Argumente und 2) ein Textfenster, welches die Resultate der ausgeführten Befehle anzeigt. Ferner enthält der Menubalken mehreren Menugruppen:

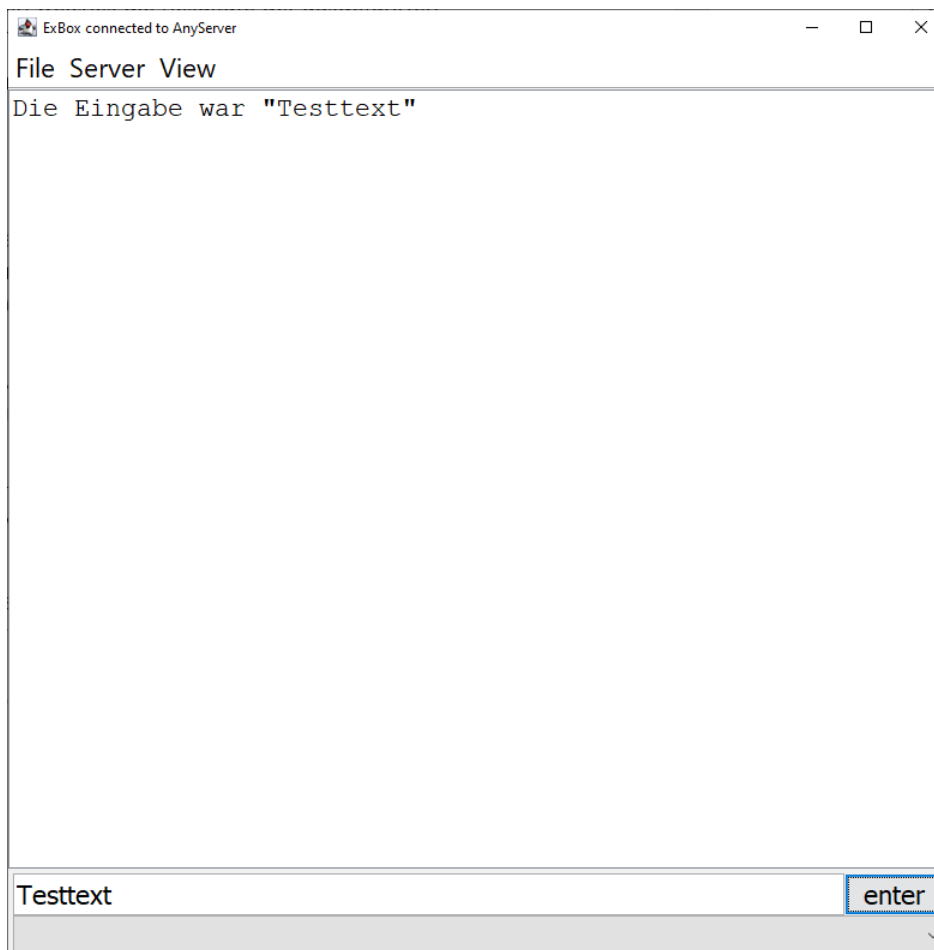
1. In der Menugruppe "File" ermöglicht "Exit" das Fenster zu schliessen (wird automatisch generiert).
2. "File" / "Open..." öffnet ein File und übergibt den Inhalt der execute-Methode (das werden wir erst in späteren Praktika anwenden).
3. In der Menugruppe "Server" ermöglicht "Connect" die Verbindungsaufnahme zu einem Server (den Sie jeweils im Praktikum erstellen werden), der das Interface CommandExecutor implementiert. Sie müssen hierfür jeweils das entsprechende, während der Compilation erzeugte Class-File in Ihrem Projekt auswählen. In der Klasse ServerFactory wird dann ein Objekt der Klasse, deren Namen im Argument der Methode createServer übergeben wird, instanziiert.
4. In der Menugruppe "View" können Sie zwischen Textdarstellung und grafischer Darstellung wechseln. Wir werden später in Praktika «Bilder» zeichnen.
5. Sobald der Enter-Knopf gedrückt wird oder "Enter" eingegeben wird, ruft der Client die Methode String execute (String command) des Servers auf. Die von dieser Methode zurückgegebenen Daten übernimmt der Client und stellt sie im Textfenster dar.

Verwenden Sie nun den im ersten Praktikum zur Verfügung gestellten Java-Code und stellen Sie eine Verbindung zum Server AnyServer her:





Geben Sie in der Textzeile im unteren Bereich einen Text ein und drücken Sie den "enter"-Button oder drücken Sie Enter. Der eingegebene Text erscheint jetzt auch im Textfenster.



Jetzt sind Sie bereit die Praktika zu lösen... viel Spass!