

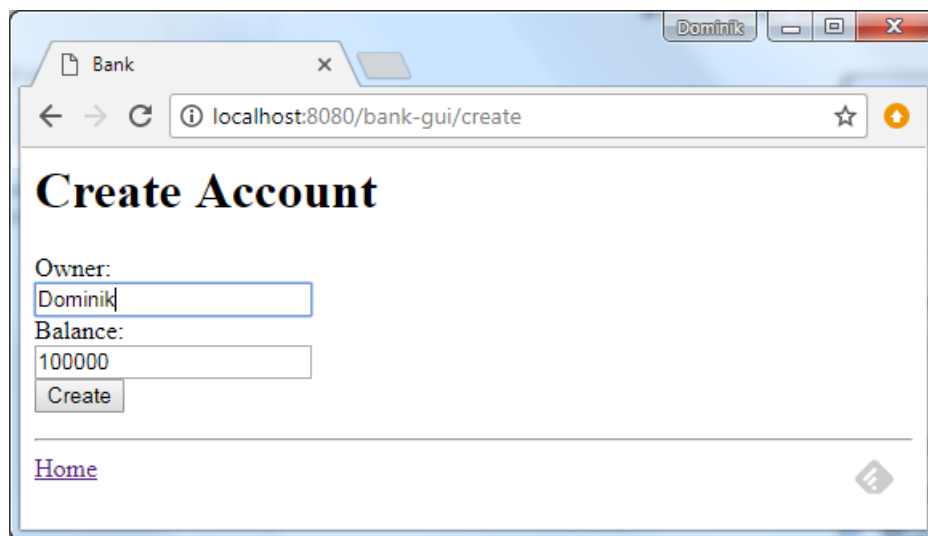
## Übung 2: HTTP Bank

In dieser Übung soll die Kommunikation über HTTP erfolgen, um auf die Bank auf dem Server zugreifen zu können. Zwei Varianten stehen zur Auswahl:

### 1) *HTML GUI*

Bei diesem Ansatz nimmt der Server, typischerweise ein Servlet, HTTP-Anfragen entgegen und bearbeitet diese, indem über die bekannten Schnittstellen `bank.Bank` und `bank.Account` auf die Bank zugegriffen wird. Damit kann die entsprechende Implementierung der Bankfunktionalität aus Übung 1 unverändert übernommen werden.

Auf den Web-Server kann über einen normalen Browser zugegriffen werden. Die Benutzerschnittstelle wird dabei durch HTML-Seiten realisiert, die entweder statisch auf dem Server bereitliegen oder dynamisch generiert werden.



### 2) *Java GUI*

Bei diesem Ansatz verwenden Sie das gegebene JavaFX GUI, um aus den Klienten-Klassen heraus HTTP-Anfragen abzusetzen, die dann vom HTTP-Server bearbeitet werden. Der HTTP-Server liefert als Antwort auf die HTTP-Anfragen natürlich keine HTML-Seiten zurück, sondern die Antworten auf die Anfragen.

Das von Ihnen in Übung 1 definierte Protokoll kann auch in Übung 2 verwendet werden. Wenn Sie mit serialisierten Command-Objekten kommunizieren, dann könnte das Anfrageobjekt im Body eines POST-Requests übertragen werden.

Verwenden Sie als Servlet-Server den Java-internen Webserver. Dokumentation zum Java-Webserver finden Sie unter <https://docs.oracle.com/en/java/javase/21/docs/api/jdk.httpserver/module-summary.html>.

Als Alternative zum `HttpServer` aus dem JDK kann auch (bei beiden Varianten) Tomcat verwendet werden. Tomcat implementiert den Servlet Standard und kann als stand-alone Servlet-Container gestartet werden oder embedded verwendet werden.

Tomcat 11 (aktuell: 11.0.4) finden Sie unter <http://tomcat.apache.org/>.

Abgabe: 17./19. März 2025 (Besprechung am 24./26. März 2025)