


E_Uebung_I

Aufgabe 1)

Der Glassfish-Server wurde erfolgreich heruntergeladen, entpackt und installiert. 

Aufgabe 2)

Der Server wurde gestartet und gestoppt. 

```
[2022-12-14 12:56:16,139] Artifact UE_1:war exploded: Waiting for server connection to start artifact deployment...
Detected server http port: 8080
Waiting for domain1 to start .....
Successfully started the domain : domain1
domain Location: /opt/glassfish6/glassfish/domains/domain1
Log File: /opt/glassfish6/glassfish/domains/domain1/logs/server.log
Admin Port: 4848
Command start-domain executed successfully.
Connected to server
[2022-12-14 12:56:24,945] Artifact UE_1:war exploded: Artifact is being deployed, please wait...
[2022-12-14 12:56:28,138] Artifact UE_1:war exploded: Artifact is deployed successfully
[2022-12-14 12:56:28,138] Artifact UE_1:war exploded: Deploy took 3,193 milliseconds
/opt/glassfish6/glassfish/bin/asadmin stop-domain domain1
[2022-12-14 01:19:18,279] Artifact UE_1:war exploded: Artifact is not deployed. Press 'Deploy' to start deployment

Process finished with exit code 0
Waiting for the domain to stop .
Command stop-domain executed successfully.
Disconnected from server
```

Abbildung 1; Server start/stopp

Aufgabe 3)

Der Server wurde gestartet und ein Web-Projekt wurde deployed. ✓

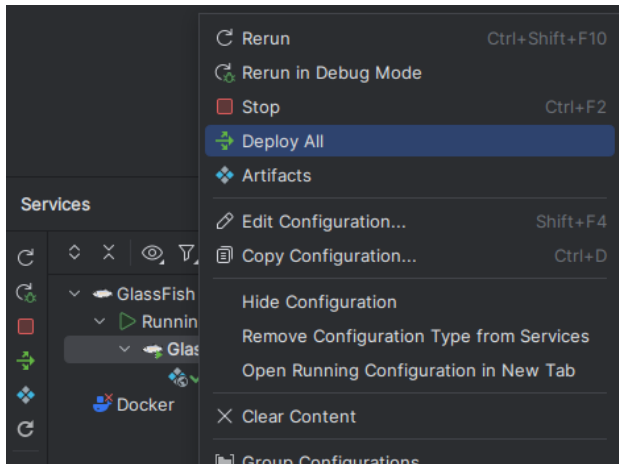


Abbildung 2; Deploy



Abbildung 3; Webserver gestartet

Begriffserklärungen

Was passiert bei Clean and Build?

Bei "Clean and Build" werden im Allgemeinen alle bisher erstellten Dateien der Anwendung gelöscht und die Anwendung neu erstellt.

Dabei werden alle relevanten Dateien der Anwendung kompiliert, gebündelt und in einem deploybaren Format bereitgestellt.

In einer Webapplikation mit Jakarta EE könnte "Clean and Build" beispielsweise dazu führen, dass alle Java-Klassen neu kompiliert und in ein Archiv-Format (z.B. JAR oder WAR) gepackt werden.

Zusätzlich könnten auch alle HTML-, CSS- und JavaScript-Dateien zu einem einzigen deploybaren Paket zusammengeführt werden.

Durch das "Clean and Build"-Verfahren wird sichergestellt, dass die Anwendung in einem konsistenten und vollständigen Zustand bereitgestellt wird und dass alle Änderungen, die während der Entwicklung vorgenommen wurden, in das finale Deployment-Paket übernommen werden.

Was passiert beim Deployment?

Beim Deployment werden alle benötigten Ressourcen wie z.B. Java-Klassen, HTML-Dateien und Datenbankverbindungen konfiguriert und eingerichtet, sodass die Anwendung betriebsbereit ist. Nach erfolgreichem Deployment kann die Anwendung dann von Nutzern über einen Webbrowser oder andere Clients aufgerufen und genutzt werden.

Was passiert beim Erstellen einer neuen Web-Applikation?

Beim Erstellen einer neuen Webapplikation geht es darum, die grundlegende Struktur und Funktionalität der Anwendung zu entwickeln.

Dazu gehört unter anderem die Festlegung der Anforderungen an die Anwendung, die Auswahl der geeigneten Technologien und Programmiersprachen, die Entwicklung des Backend- und Frontend-Codes sowie die Integration von Datenbanken und Frameworks.

Was passiert beim Starten einer neuen Web-Applikation?

Wenn eine Web-Anwendung gestartet wird, wird in der Regel ein Server gestartet, der die Anwendung hostet. Dieser Server kann entweder auf demselben Computer wie die Anwendung installiert sein oder auf einem Remote-Server. Sobald der Server gestartet ist, wird die Anwendung auf dem Server ausgeführt und kann von Benutzern über das Internet aufgerufen werden. Dies geschieht normalerweise, indem der Benutzer eine URL in seinem Webbrowser eingibt, die auf die Anwendung verweist. Der Server empfängt dann die Anfrage des Benutzers und sendet die Antwort in Form der Web-Anwendung zurück an den Benutzer. Die genauen Schritte, die beim Starten einer Web-Anwendung ausgeführt werden, können je nach verwendeter Anwendung und Server Software variieren. Im Allgemeinen ist es jedoch der Prozess, der beschrieben wurde, um die Anwendung für Benutzer verfügbar zu machen.

Properties einer Web-Applikation

Eine Webapplikation kann verschiedene Eigenschaften haben, je nachdem, welche Anforderungen sie erfüllen soll. Einige mögliche Eigenschaften einer Webapplikation sind:

- Benutzerfreundlichkeit: Eine gute Webapplikation sollte für die Benutzer einfach zu verstehen und zu nutzen sein.
- Zugänglichkeit: Eine gute Webapplikation sollte von möglichst vielen Menschen genutzt werden können, auch von Menschen mit Behinderungen oder Einschränkungen.
- Sicherheit: Eine gute Webapplikation sollte die Daten und Informationen ihrer Nutzer schützen und vor unbefugtem Zugriff schützen.
- Skalierbarkeit: Eine gute Webapplikation sollte in der Lage sein, ihre Leistung und Kapazität an die steigenden Anforderungen anzupassen, ohne dass die Qualität der Anwendung beeinträchtigt wird.
- Flexibilität: Eine gute Webapplikation sollte sich leicht an veränderte Anforderungen anpassen lassen und neue Funktionen und Features integrieren können.
- Performance: Eine gute Webapplikation sollte schnell und reaktionsschnell sein und eine hohe Verfügbarkeit bieten.