参考

https://www.zu-min.com/archives/1255

ChatGPT

1. vscode で servlet の簡単なプログラムを作って tomcat でリモートデバッグする流れを順番に教えて
2. この手順のデプロイを簡単にする方法を教えて
3. tomcat7-maven-plugin を使って簡単にするデプロイできませんか？

方法 1

概要

tomcat を別に立てて、tomcat の manger に package を送り込んでデプロイする方法。

環境構築手順

1. maven を使った servlet のサンプルを作成する
2. tomcat の manager を設定する
    1. cp -r /usr/local/tomcat/webapps.dist/manager /usr/local/tomcat/webapps/manager
    2. tomcat-users.xml 編集
        1. <role rolename="manager-gui"/>
        2. <role rolename="manager-script"/>
        3. <user username="admin" password="password" roles="manager-gui,manager-script" />
    3. webapps/manager/META-INF/context.xml 編集
        1. IP 制限を外す (docker を使ってホストから接続するなどの場合)
        2. <Valve className="org.apache.catalina.valves.RemoteAddrValve" allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" /> をコメントアウト
3. pom.xml 編集
    1. tomcat7-maven-plugin の設定
        1. <url>http://tomcat:8080/manager/text</url>
        2. <username>admin</username>
        3. <password>password</password>
        4. <path>/app-path</path>
        5. <path> は tomcat のコンテキスト (アプリの URL パス) を指定できる
4. tomcat の JPDA を有効にする
    1. tomcat の起動オプションを修正
        1. catalina.sh jpda run
    2. JPDA の待受ホスト、ポートの設定 (0.0.0.0 にして他コンテナからのアクセスを可能にする)
        1. 環境変数に以下を設定
        2. JPDA_ADDRESS=0.0.0.0:8000
5. VSCode のデバッグ実行で tomcat の manager へのデプロイとリモートデバッグへの接続
    1. launch.json

```
{
    // IntelliSense を使用して利用可能な属性を学べます。
    // 既存の属性の説明をホバーして表示します。
    // 詳細情報は次を確認してください : https://go.microsoft.com/fwlink/?linkid=830387
    "version": "0.2.0",
    "configurations": [
        {
            "type": "java",
            "name": "Debug",
            "request": "attach",
            "hostName": "tomcat",
            "port": 8000,
```

```
            "preLaunchTask": "deploy"
        }
    ]
}
```

1. tasks.json

```
{
    // See https://go.microsoft.com/fwlink/?LinkId=733558
    // for the documentation about the tasks.json format
    "version": "2.0.0",
    "tasks": [
        {
            "label": "deploy",
            "type": "shell",
            "command": "mvn -B tomcat7:redeploy -f \"${workspaceFolder}/pom.xml\"",
            "group": "build"
        }
    ]
}
```

一式

上記の環境を devcontainer と docker で作成した環境

```
make build
make up
```

で http://localhost:8080/app-path/hello にアプリがデプロイされ、ブレークポイント等が使えるようになる。

方法 2

概要

上記の方法 1 とほぼ同じだが、VSCode の保存時の自動コンパイルの出力結果 (target/classes) などを直接指定することで自動デプロイを実現する。

設定

■ docker-compose.yml

上記の docker-compose.yml の tomcat の volumes に以下を追加

```
      - ./servlet/src/main/webapp:/usr/local/tomcat/webapps/app-path
      - ./servlet/target/classes:/usr/local/tomcat/webapps/app-path/WEB-INF/classes
      - ./servlet/target/hello-servlet/WEB-INF/lib:/usr/local/tomcat/webapps/app-path/WEB-INF/lib
```

tomcat のアプリに target/classes などを直接マウントする。

一式

方法 3

概要

上記の方法 2 とほぼ同じだが、VSCode の保存時の自動コンパイルの出力結果 (target/classes) など

を、VSCode でデバック実行をしたときに tomcat にデプロイする。
自動デプロイしないので、保存するたびにデプロイが実施されない。

設定

■ docker-compose.yml

上記の docker-compose.yml の tomcat の volumes に以下を追加

```
        - ./servlet/src/main/webapp:/usr/local/tomcat/webapps/app-path
                                                                                -
    ./servlet/target/hello-servlet/WEB-INF/classes:/usr/local/tomcat/webapps/app-path/WEB-INF/classes
        - ./servlet/target/hello-servlet/WEB-INF/lib:/usr/local/tomcat/webapps/app-path/WEB-INF/lib
```

■ .vscode/launch.json

```
{
    // IntelliSense を使用して利用可能な属性を学べます。
    // 既存の属性の説明をホバーして表示します。
    // 詳細情報は次を確認してください : https://go.microsoft.com/fwlink/?linkid=830387
    "version": "0.2.0",
    "configurations": [
        {
            "type": "java",
            "name": "Debug Reload",
            "request": "attach",
            "hostName": "tomcat",
            "port": 8000,
            "preLaunchTask": "reload"
        }
    ]
}
```

■ .vscode/tasks.json

```
{
    // See https://go.microsoft.com/fwlink/?LinkId=733558
    // for the documentation about the tasks.json format
    "version": "2.0.0",
    "tasks": [
        {
            "label": "reload",
            "type": "shell",
                                "command":    "\\cp    -rT    \"${workspaceFolder}/target/classes\"
\"${workspaceFolder}/target/hello-servlet/WEB-INF/classes\"    &&    curl    --user    admin:password
http://tomcat:8080/manager/text/reload?path=/app-path",
            "group": "build"
        }
    ]
}
```

一式