**Anleitung SCALA Applikation ‚lwfportal‘ im  
Programm Langfristige Waldökosystem-Forschung LWF**

Entwickelt von Simpal Kumar 2017 bis 2020

WSL, Birmensdorf, 2020

# Einleitung

Im Programm ‚Langfristigen Waldökosystem-Forschung (LWF)‘ (www.wsl.ch/lwf) werden unter anderem auf LWF-Flächen und Sanasilva Stichproben-Flächen wiederkehrende Messungen durch. Die Messungen werden von automatischen Messstationen aufgezeichnet, im Feld erhoben und im Labor bestimmt. Organisatorisch sind diese Messungen in einzelne Erhebungen (www.wsl.ch/lwf > Daten > Metadaten oder envidat.ch) gegliedert, und die Aufgaben und Verantwortlichkeiten innerhalb der WSL entsprechend verteilt. Die Daten fliessen jedoch in zentrale LWF Datenbanken (Oracle-Datenbank) von wo aus sie für verschiedene Zwecke genutzt werden können, um beispielsweise die jährliche Data Submission an ICP Forests abzuwickeln.

Ein Teil der Abläufe/Prozesse der Datenverarbeitung sind teil- oder vollautomatisiert. Ein wichtiges Werkzeug ist dabei die hier beschriebene SCALA Applikation ‚lwfportal‘.

# Technologie

|  |  |  |
| --- | --- | --- |
| **Software** | **Version** | **Introduction** |
| Oracle-Datenbank | **12.1** | https://docs.oracle.com/en/database/oracle/oracle-database/index.html |
| Oracle JODBC Driver (ojdbc6-11.2.0.2.0.jar) | **6** | https://www.oracle.com/database/technologies/jdbcdriver-ucp-downloads.html |
| Java Runtime Environment (JRE) | **1.8 (8.1.2.1)** | java.com |
| Java Development Kit (JDK) | **1.8 (8.2.1.1)** | java.com |
| Scala | **2.12.6** | https://docs.scala-lang.org |
| Scala Built Tools (SBT) | **0.13.13.1** |  |
| IntellJ Idea | ideal C-2017.1 or idalU-2018.2 | www.jetbrains.com/idea |
| Maven Repository |  | https://maven.apache.org/ |
| Playframe |  |  |
| Git | z.B. 2.26.2 | https://git-scm.com/book |
| WinSCP |  |  |
| Putty 0.64 | z.B. 0.64 |  |
| FileZilla |  |  |
| Virtuelle Linux Machine | 3.10.0 x86 – 64bit | www.linux.org |

\*) fett=Versionen müssen miteinander kompatibel sein.

# Requirements and Privilegues

|  |  |
| --- | --- |
| **Requirement** | **Herausgeber** |
| GitHub Account | github.com  (basic accounts free of charge) |
| Invitation to the Repository simpalK/beo | Simpal Kumar |
| Passwort der Virtuellen Maschine lwfportal | Simpal Kumar/Markus Reinhard |
| Windows-, Unix-like oder Mac-Rechner,  Administrator-Passwort | WSL IT (Ticket) |

# Manual for Development and Maintenance

## Installation of Virtual Machine

Die virtuelle Linux Maschine wird von der WSL IT (Markus Reinhard) aufgesetzt.

## Installation on Personal Computer (e.g. Windows 10)

## Remove disturbing Software

Important: Uninstall older/newer of Java RunTime Environment and restart the computer.

## Install the following Software Versions

1. WinSCP (any version)
2. Putty (any version)
3. Git (any version)
4. Java Runtime Environment JRE (Version 1.8)
5. Java Development Kit JDK (Version 1.8)

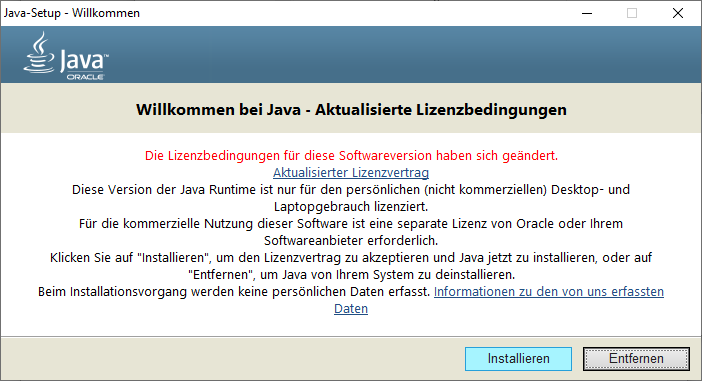
*... and only hereafter also ...*

1. IntelliJ Idea (Version 2017.1)

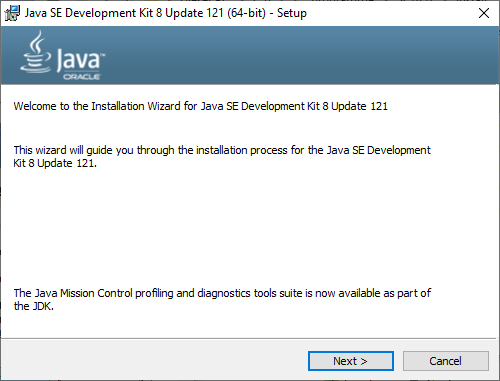
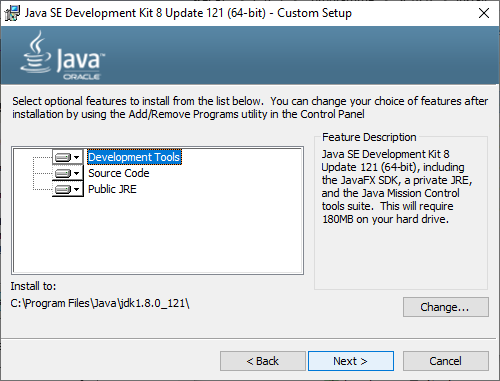
Start with the installation of ***WinSCP, Putty*** and ***Git***. The installation of this programmes is not shown here. When successfully installed you may continue with the steps shown in the following.

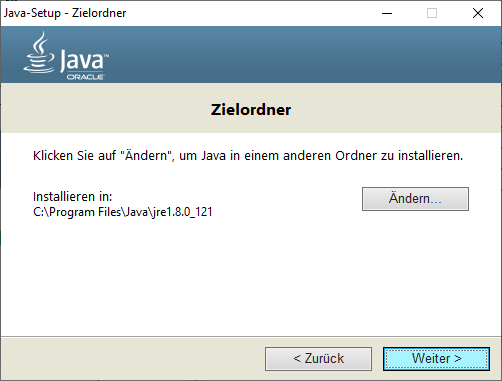
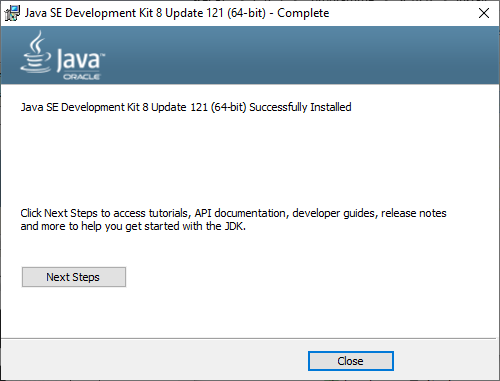
### Installation Java Runtime Environment 1.8 mit

jre-8u211-windows-au.exe



### Installation Java Development Kit 1.8 mit jre-8u211-windows-au.exe

## Repository : Git Bash und Github-Account mit SSH Key verbinden

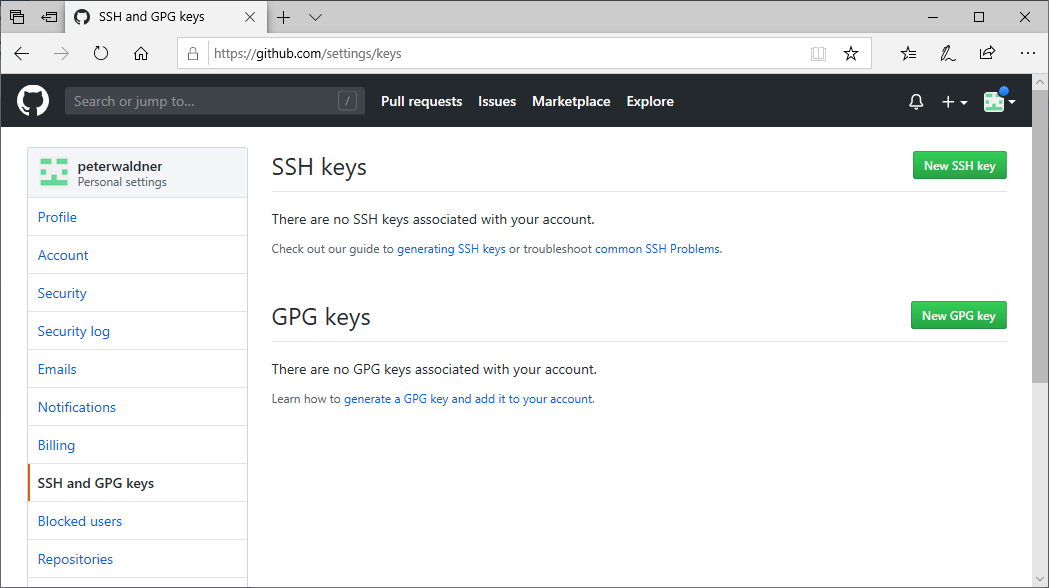
**SSH-Key anlegen in Git Bash mit**$ ssh-keygen   
(Passphrase leer lassen.)

Anzeigen lassen des Public-Texts des Keys mit   
$ cat .ssh/rsa\_id.pub

kopieren des angezeigten Textes 'ssh-rsa .... ' mit   
rechter Maustaste > copy or Ctrl-Insert

**Insert the public part of the SSH Key into your Github Account**

Open Webbrowser, Sign in to github.com with your username and password  
Choose'Profile > Settings > SSH and GPG keys',

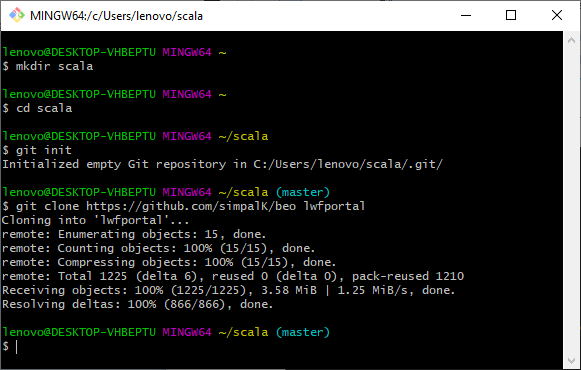


Add ‚New SSH key‘, type any name in the upper field and paste the copied text into the lower field

Download Source Code from the Repository

In the Terminology of the Version Control Systems, this action is called ‘checkout’ or ‘clone’. Using Git Bash it can be done as follows

Erstellen eines Verzeichnis c:/users/lenovo/scala auf dem lokalen Komputer, konfigurieren als Git-Haupt-Verzeichnis mit 'git init' und Inhalt des Repositories github.com/simpalK/beo herunterladen (sogenanntes 'checkout', bzw. 'clone') mit 'git clone [url] [name Unterordner]



**Security Remark:**

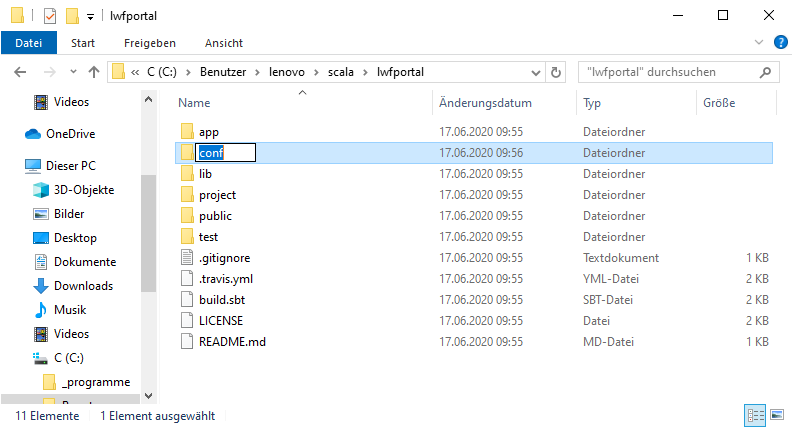
The subdirectory /conf is not contained in the github repository for security reasons. This directory is also prevented from beeing shared with github by listing it in the ‘.gitignor’ file.

.gitignor files not stored in the repository

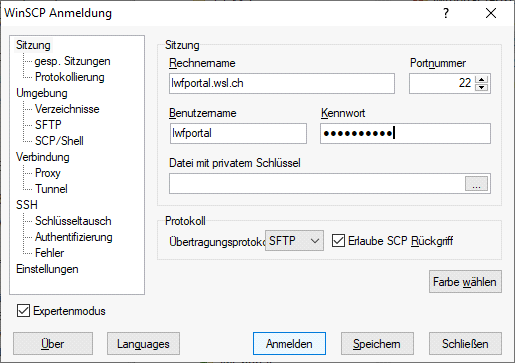
The directory /conf contains the files with the credentials (servernames, usernames, passwords, portnumbers etc.). The files application.conf contains main configuration (Akka Configuration), the file logback.html the places where the logs are written and routes the IP configuration. These files are stored within WSL only (e.g. on H:\ and on the virutal machine).

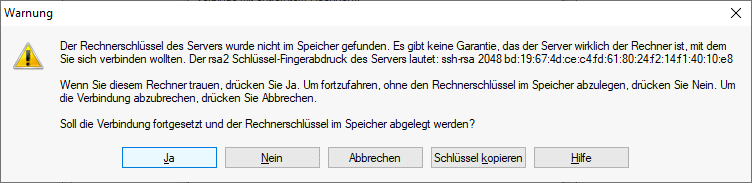
After the 'download' from the source code is completed, these file have to be added from a local source.

## Add Configuration Files to the project

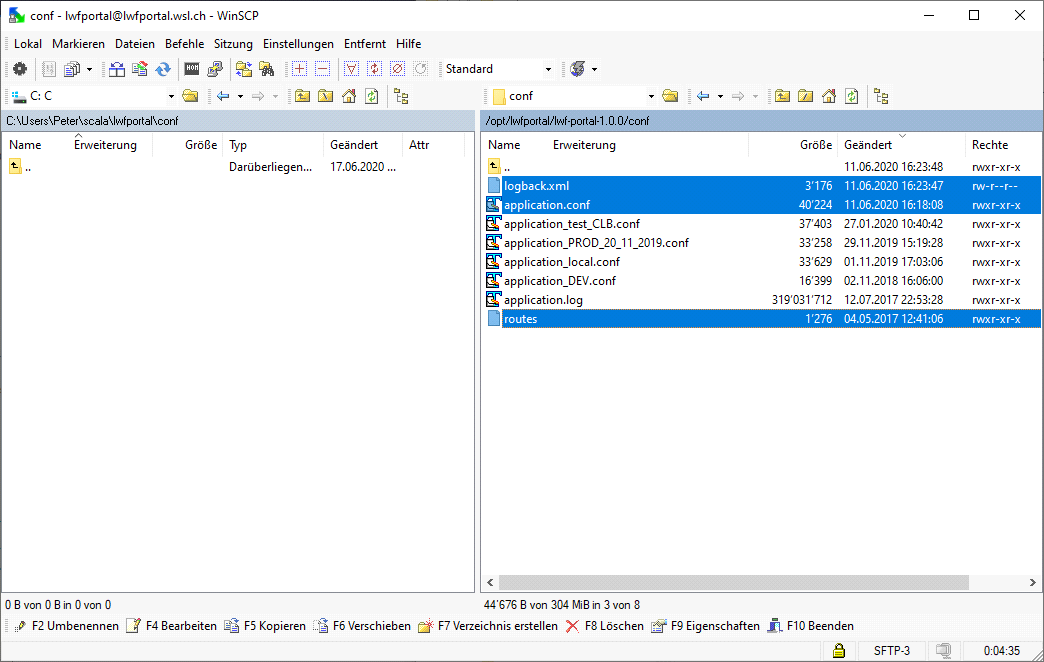


start WinSCP and open an sftp connection to lwfportal.wsl.ch with server: lwfportal, user: lwfportal and password: \*\*\*\*\*\*\*\*\*\*

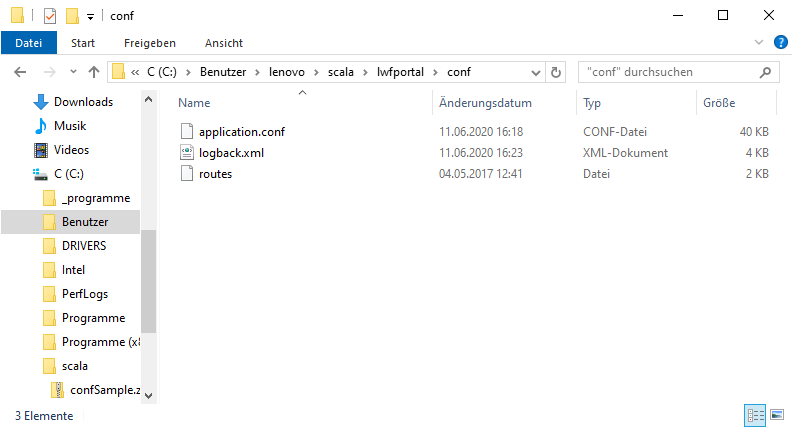




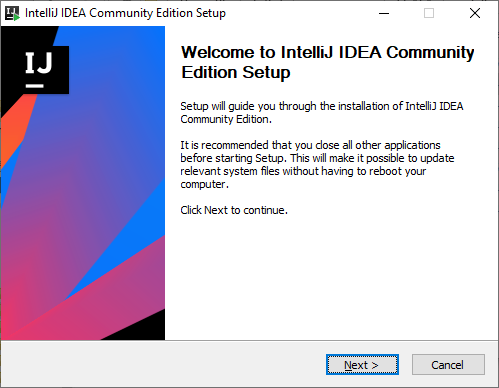
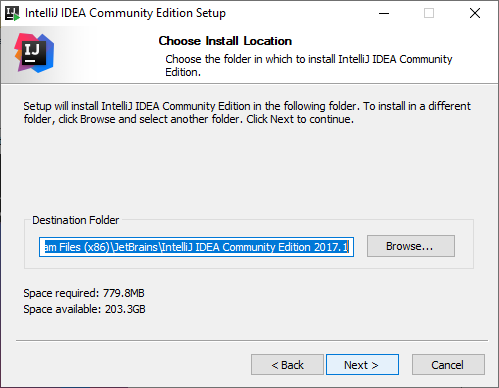
copy the files 'application.conf', logback.xml,routes from the lwfportal.wsl.ch

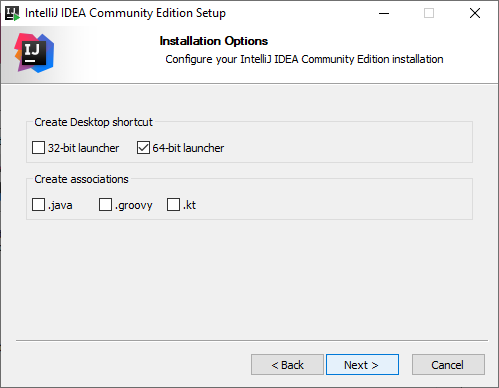
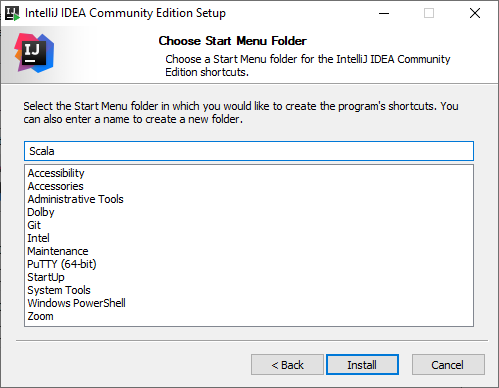


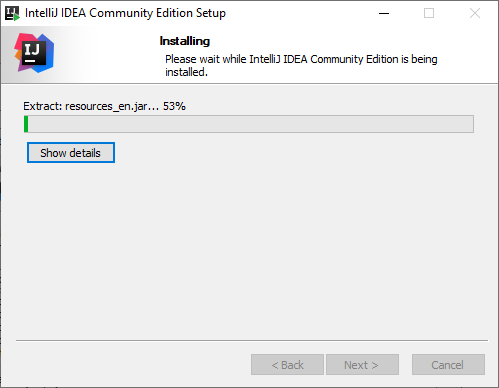
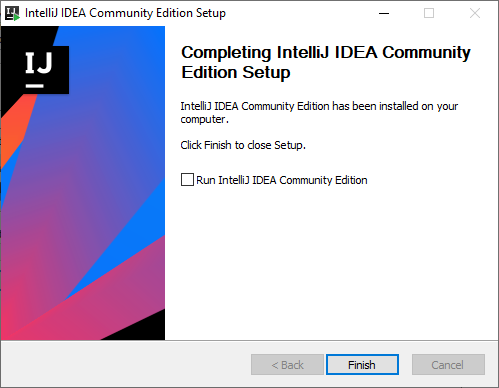
past them under .../lwfportal/conf

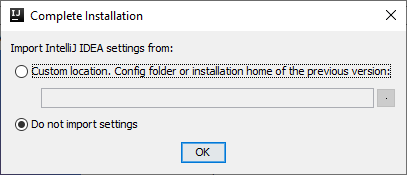


## Install Development Client 'IntellJ Idea 2017.1'

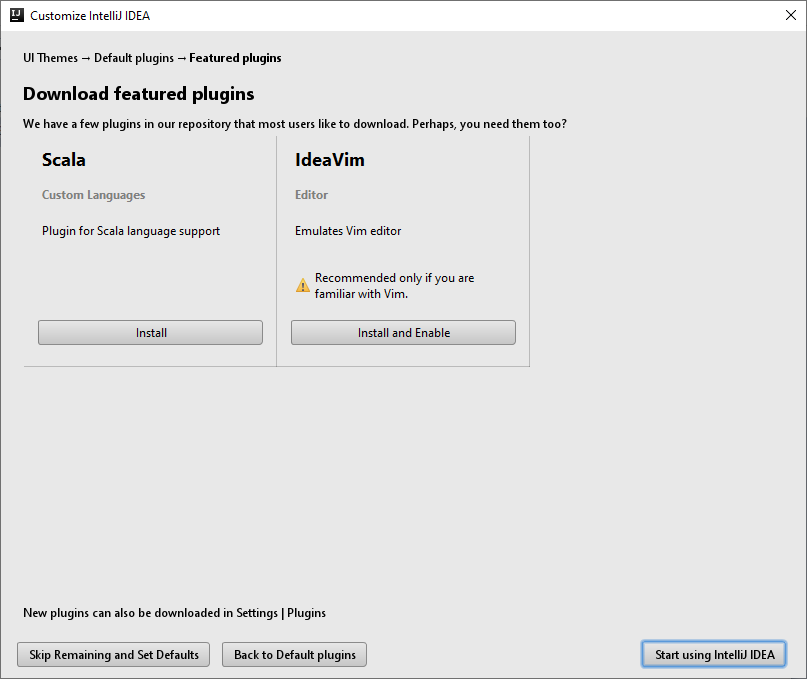
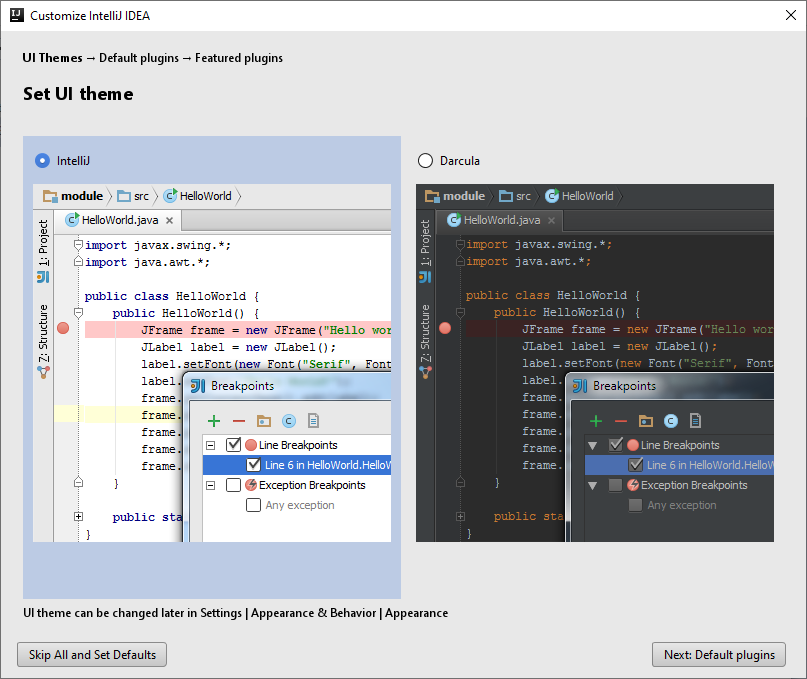
 

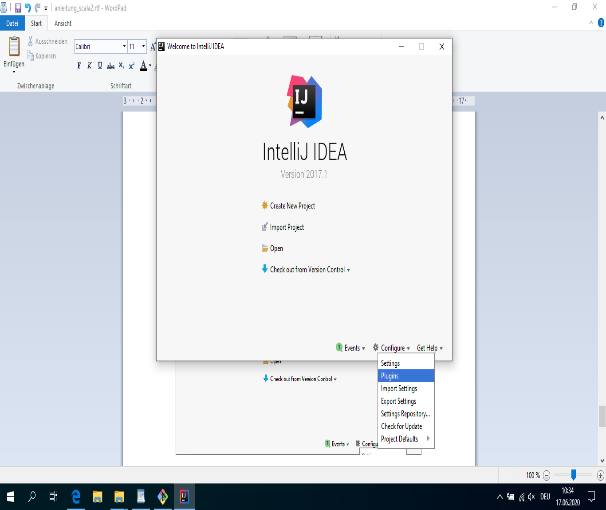
 



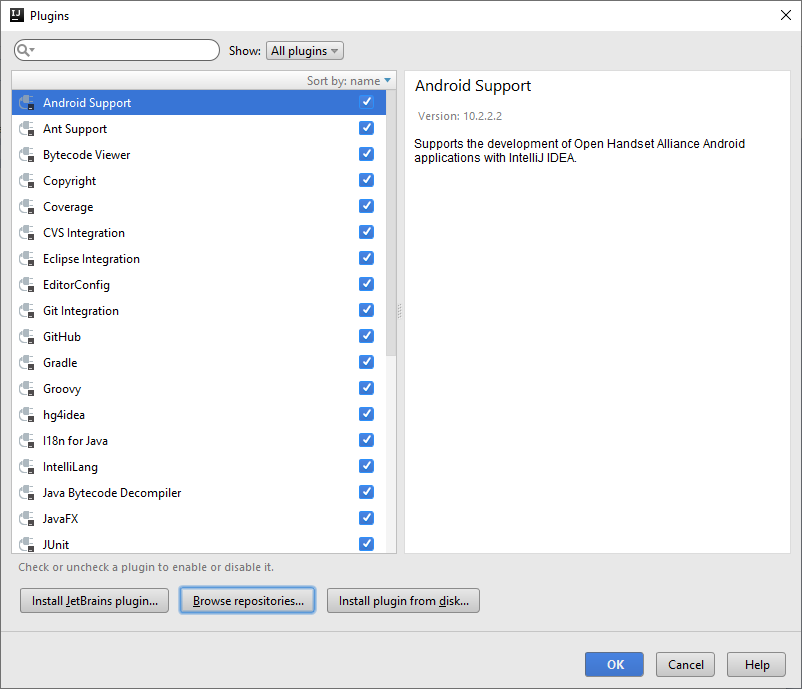
Choose Appearance Theme and Skip the remaining settings and set default settings



Choose Menü 'IntelliJ IDEA Version 2017.1 > Configure > Plugins

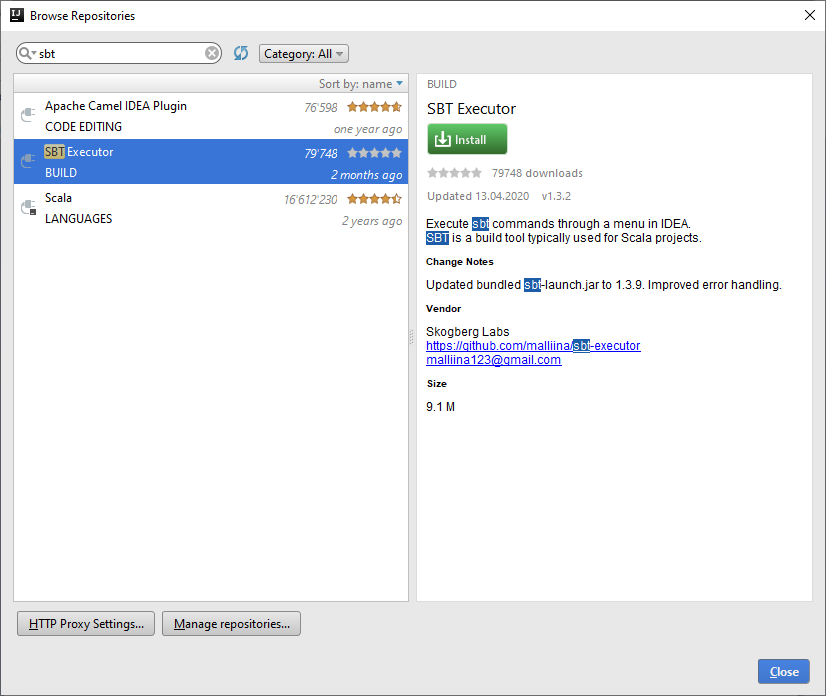
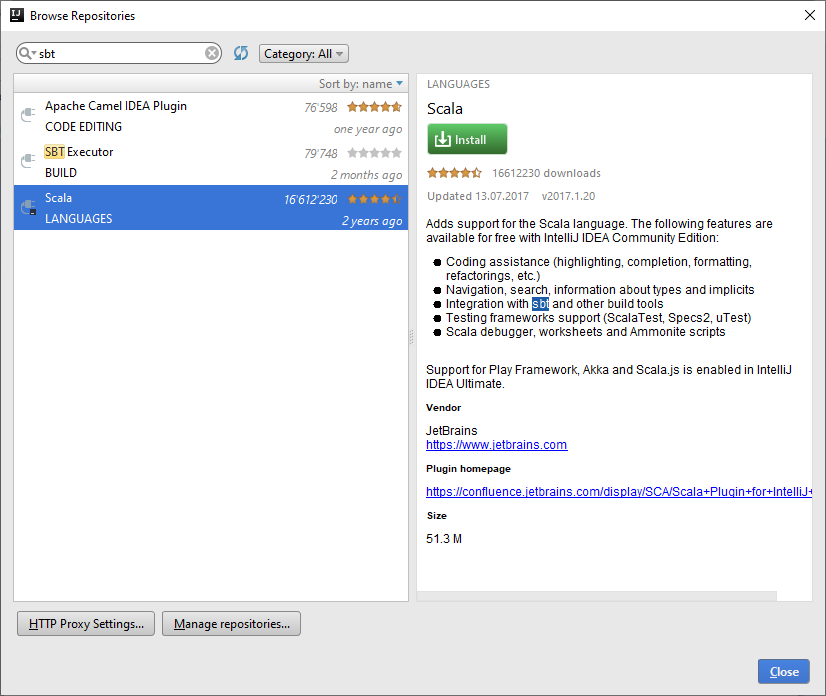


Select 'browse repositories ...'



search for 'sbt' in the search field on the top left part of the window

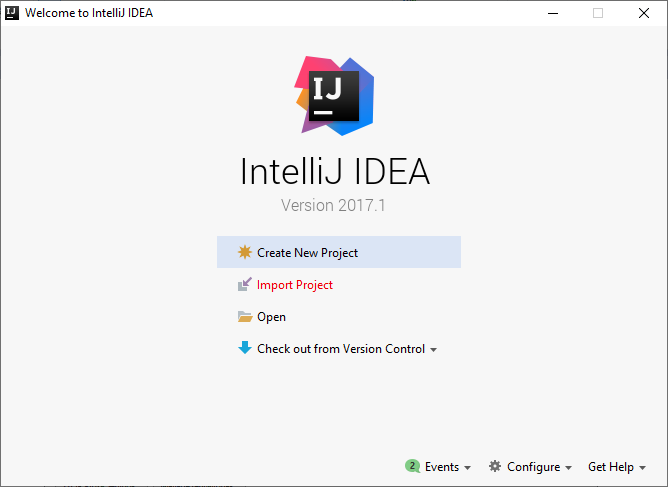
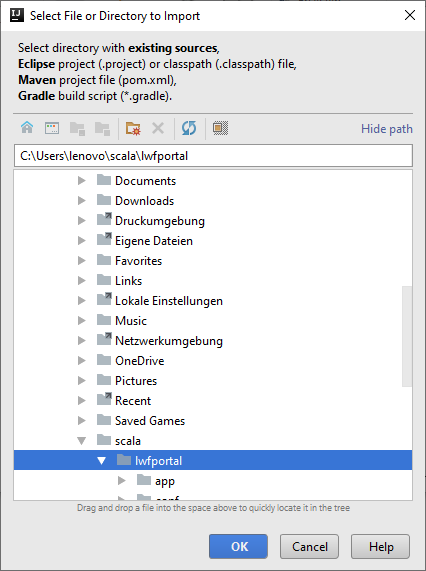
Select and install 'scala' and 'sbt executor'

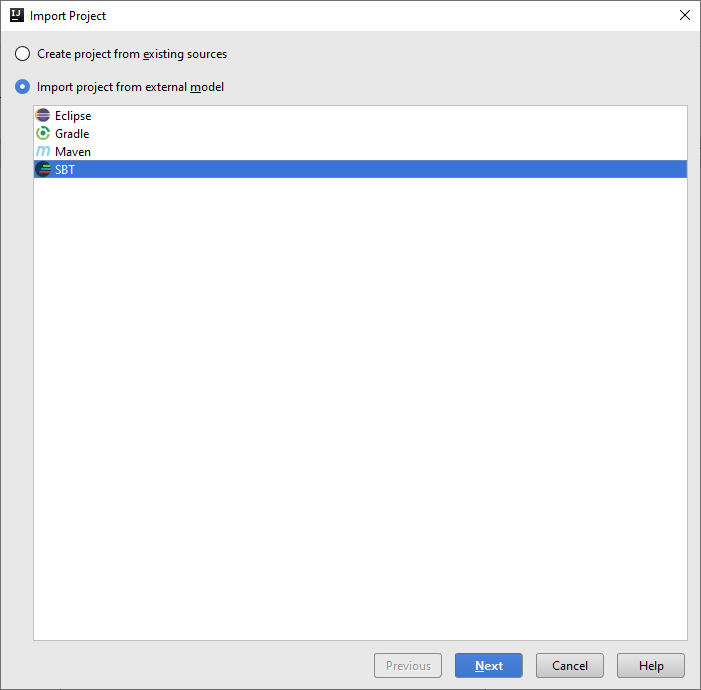
 

Restart IntelliJ Idea

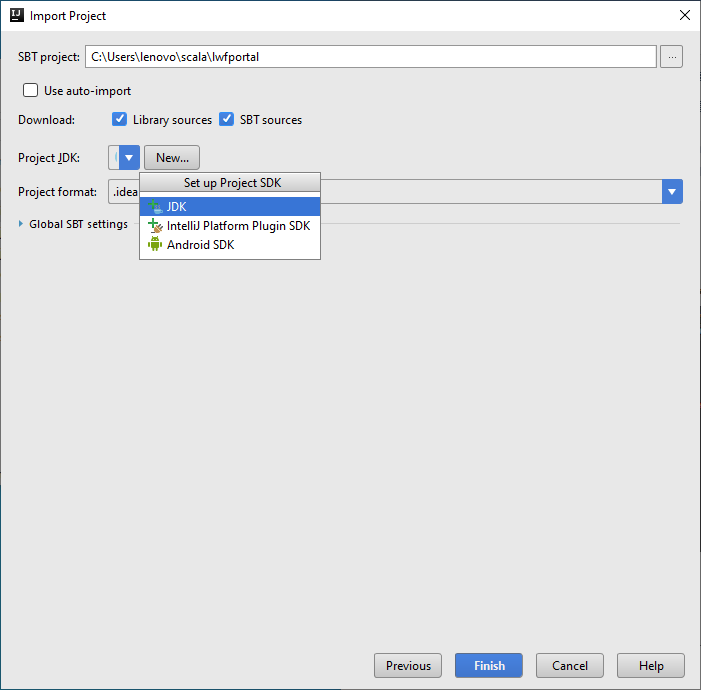
## Import SCALA Project

Press ‚Import Project‘

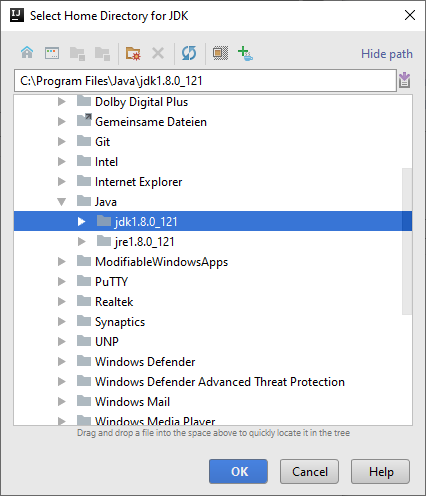
 



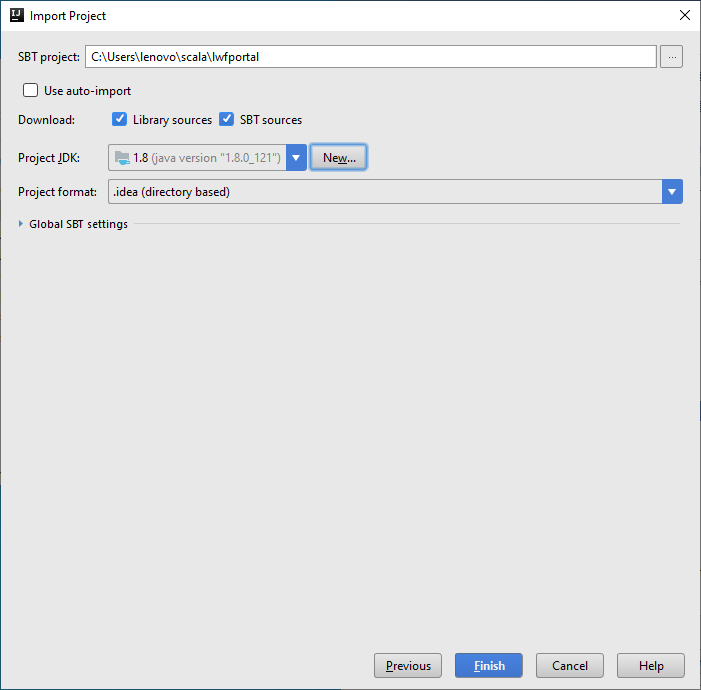
Check ‚SBT sourcees‘ and choose or add the JDK 1.8

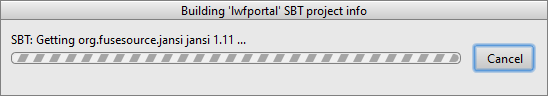


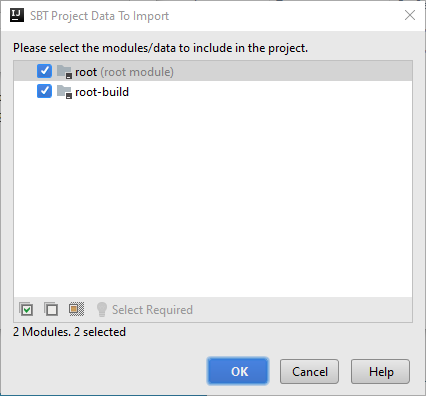
select the directory with your Java Development Kit Installation (for installation see above).



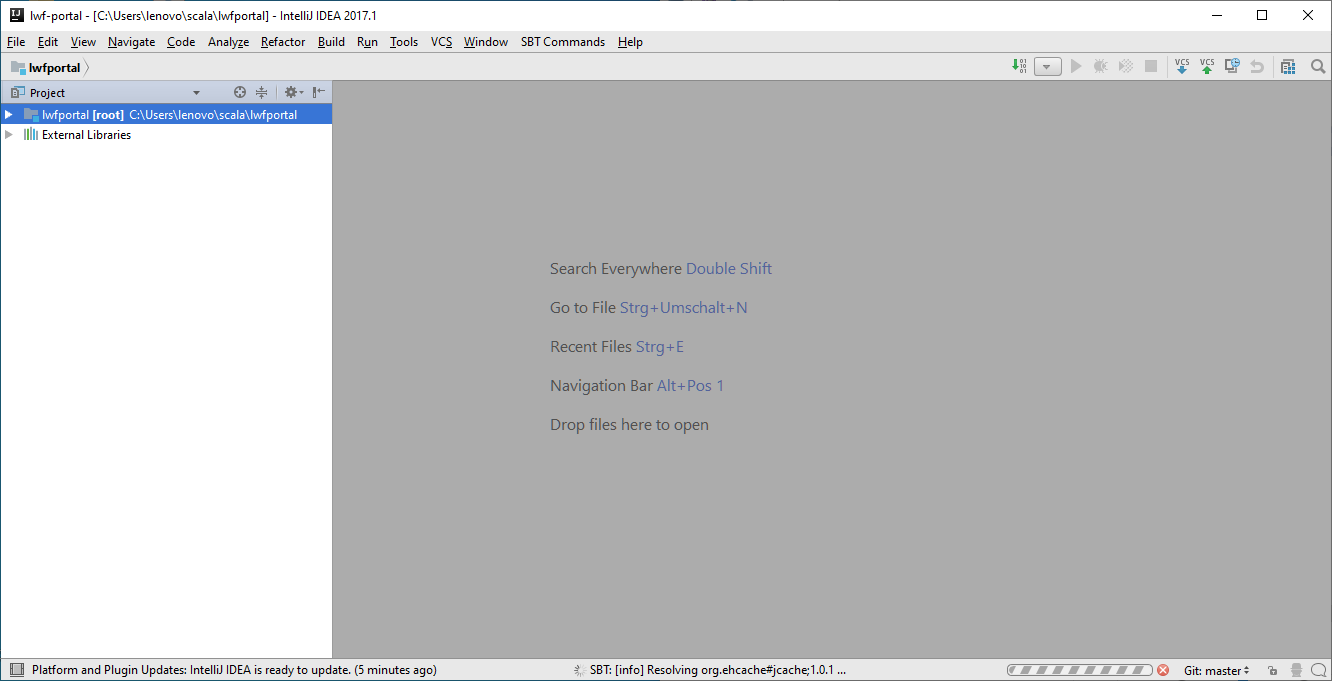
Check 'SBT sources'!



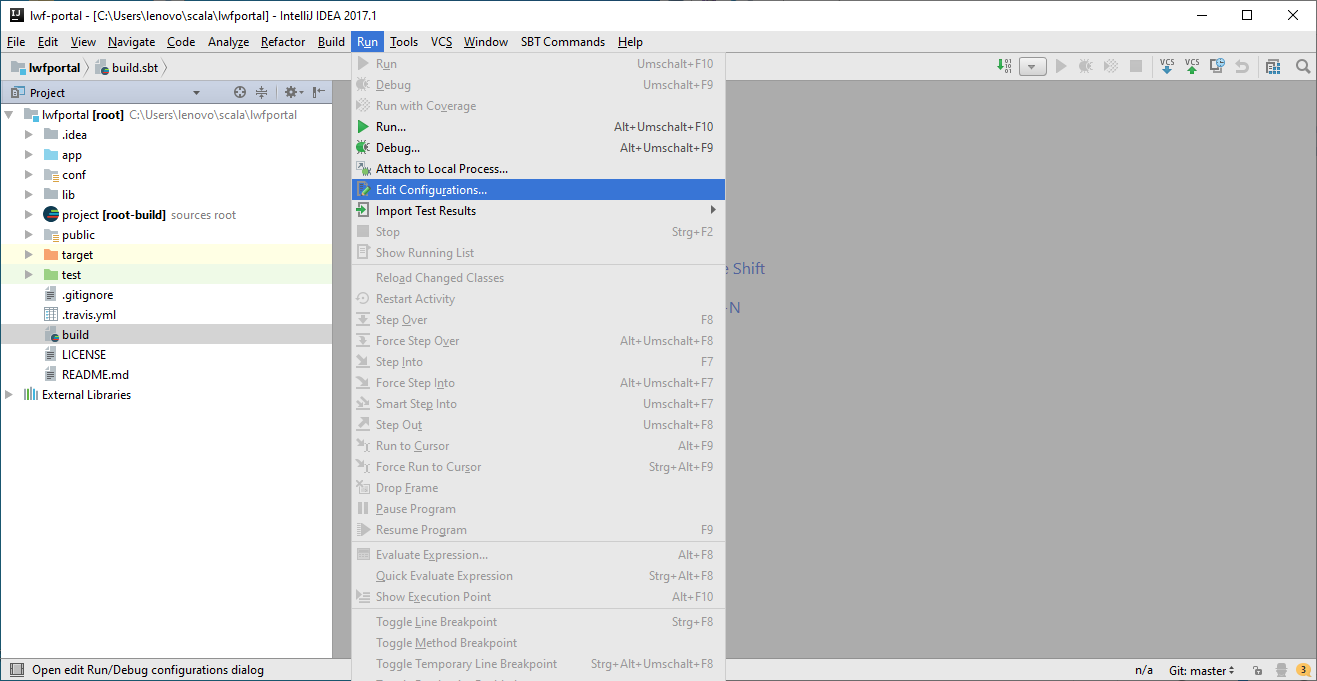




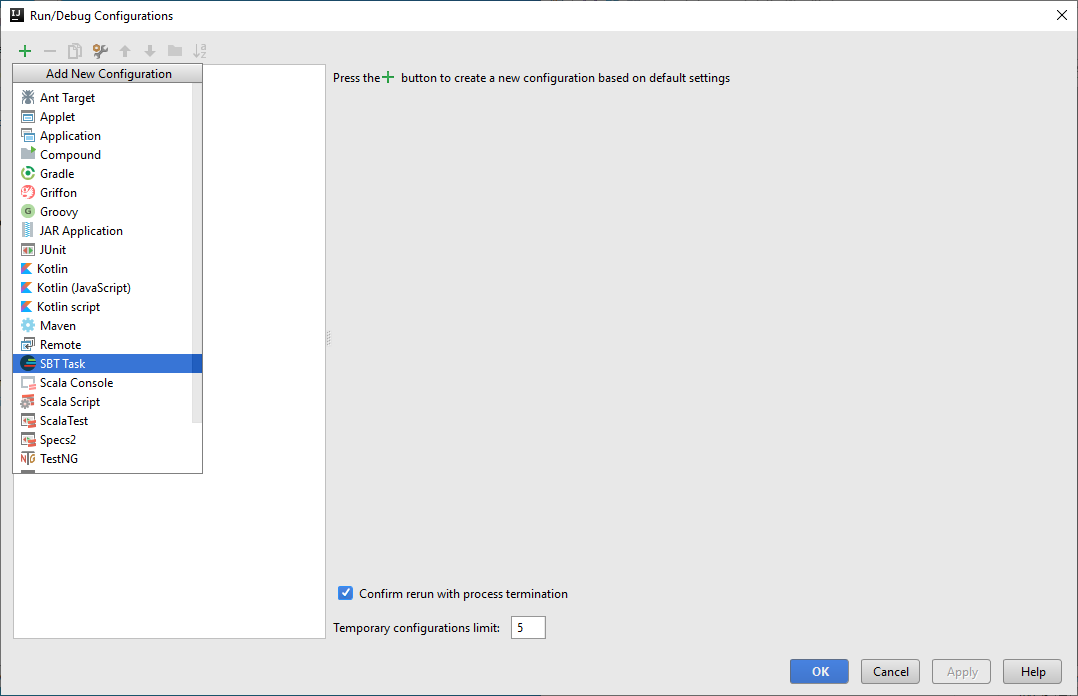
Select the project 'lwfportal' in the Explorer subwindow



select menu 'Run > Edit Configuration'



Press the green '+' top left and add 'SBT Task'

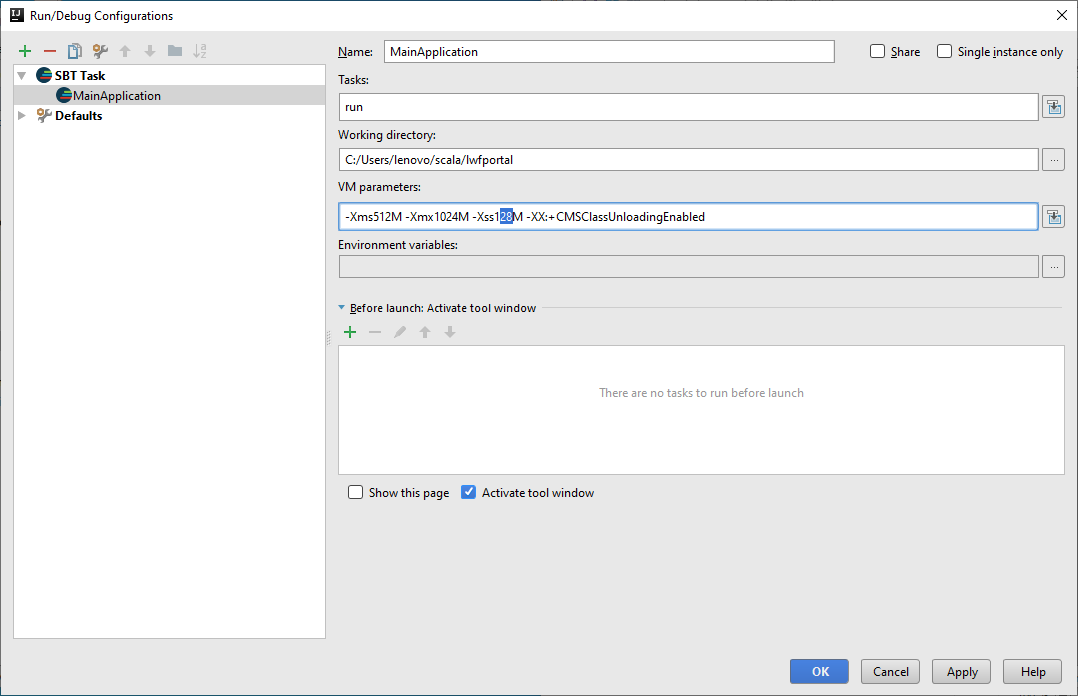


Rename 'Unnamed' to 'MainApplication', add the command 'run' of the task and increase memory of Virtual Machine (VM) by changing the settings

Name: 'MainApplication'

Task: 'run'

VM paremters: '.... -Xss**12**8M ....'



Bar on the right side appears including a button 'SBT'.

## Develop or Maintain the Source Code (make the necessary changes)

/conf/application.conf Configuration of Application and Modules

/conf/logback.xlm Configure Logging activities

/app/Module.scala Switch on/off whole Modules

/app/models/repositories/\*.scala  
 DB-Connection and collected SQL Statments

/project/plugins.sbt Versions of used SBT Plugins

/built.sbt Repositories of included external source code libraries

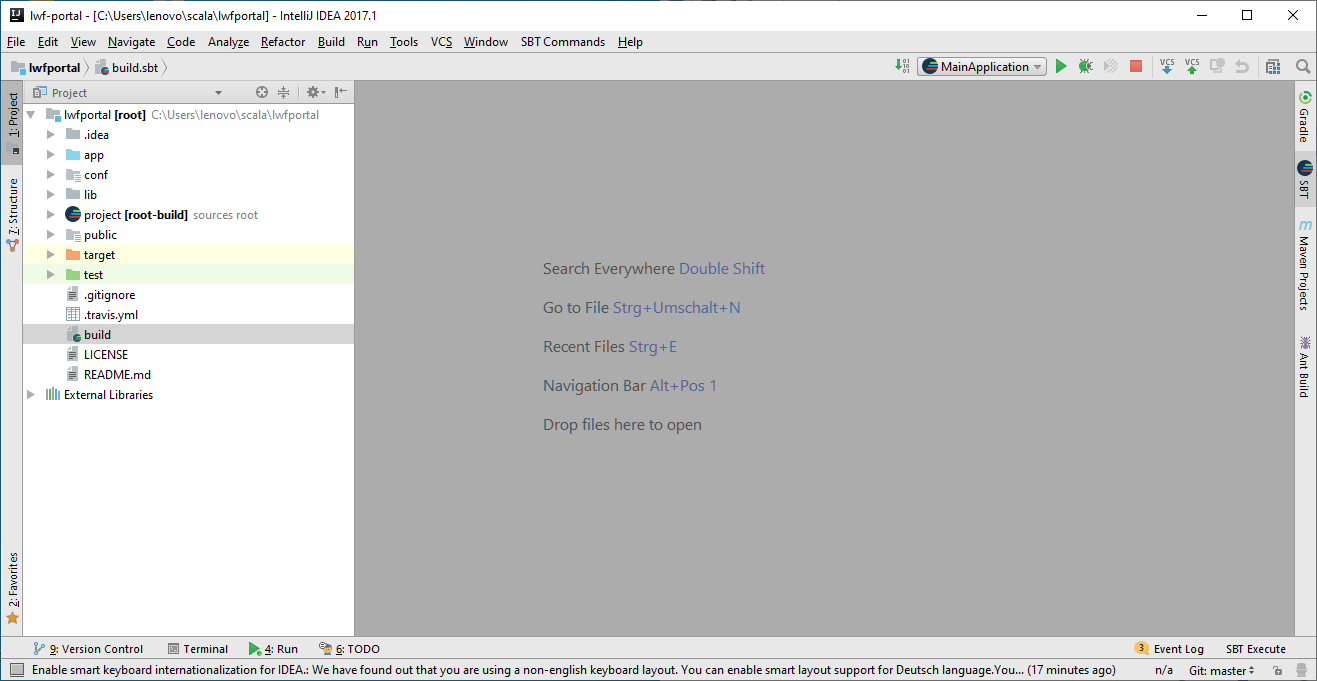
see also the SCALA conepts

changes are immediately in the saved in the local folder (only)

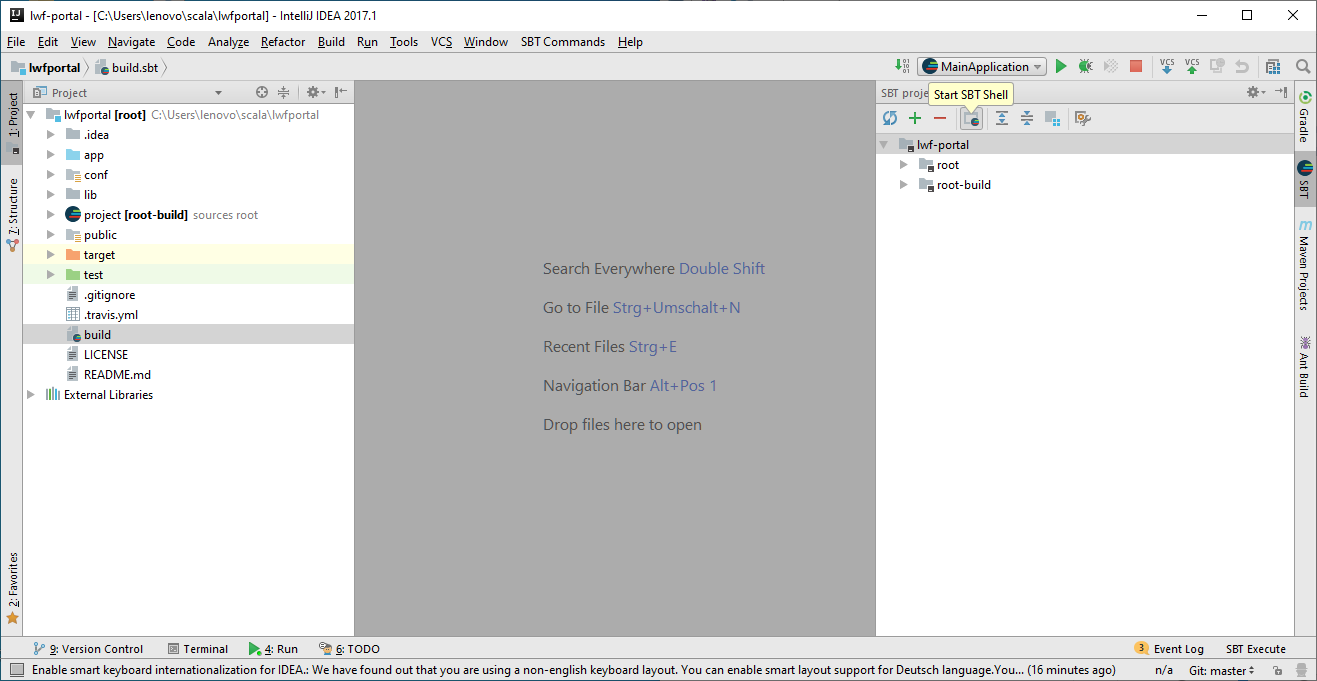
**Write Changes back to the Repository**

Start 'Git Bash', schreibe die Änderungen ins Repository zurück, versehen mit dem Kommentar '......' (in VCS Terminology ‚commit‘ oder ‚push request‘, hier wird commit verwendet)  
$ cd script/lwfportal  
$ git commit -m 'commit message'

## Compilation of the Application

Press 'SBT' on the right side ..   


An subwindow 'SBT projects' appears on the right side. with press the small SBT terminal button in its top bar.



To compile the project and build a \*.jar and a \*.zip package file type the following commands into the Terminal

> **reload**

> **clean**

> **compile**

Messages Explanation Action

[warn] ... depreaced ... commands used that still work but are announced to be replaced in future versions no immediate action needed

[info] resolving version and repository of external java code is stored in .../lwfportal/built.sbt if it is not avalable, consult that website, usually there are recommandation which newer version can be taken.

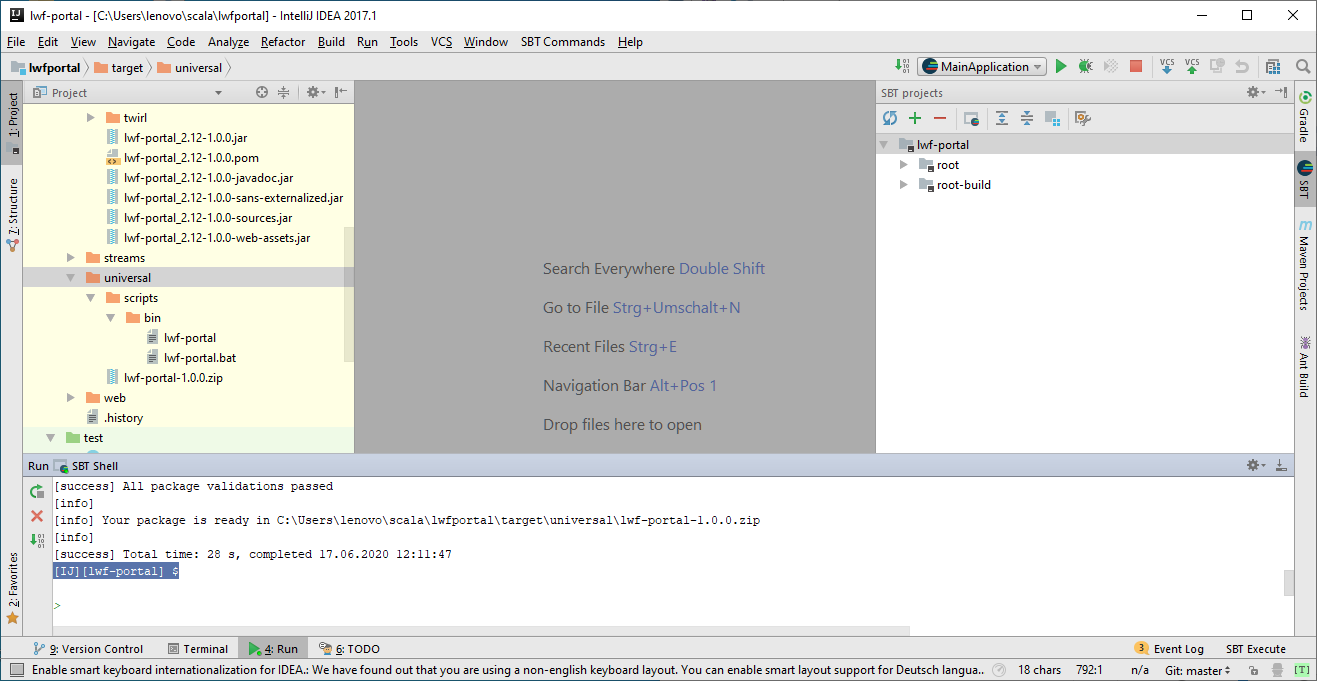
[error] ... something went wrong identify and solve the probleme

> **package**

a lwf-portal\_\*.jar package files are created in .../lwfportal/target/scala-2.12/

> **dist**

a lwf-portal-\*.zip package file is created in .../lwfportal/universal/



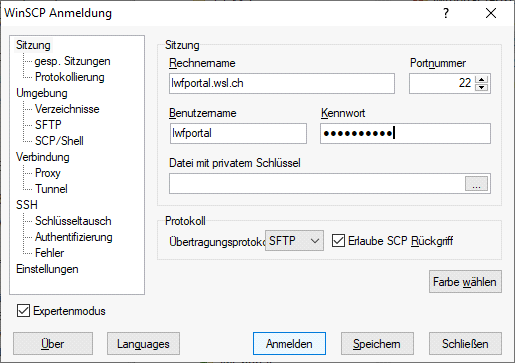
Furthermore, a Unix-scipt 'lwfportal' and a Windows-Shell-Script 'lwf-portal.bat' is created to start the application with the \*.jar or the \*.zip file on a machine with a Unix or a Windows Operating System, respectivaly.

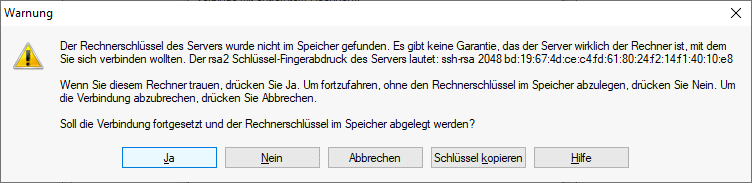
## Deployment

Änderungen auf das operationel laufende System übertragen.

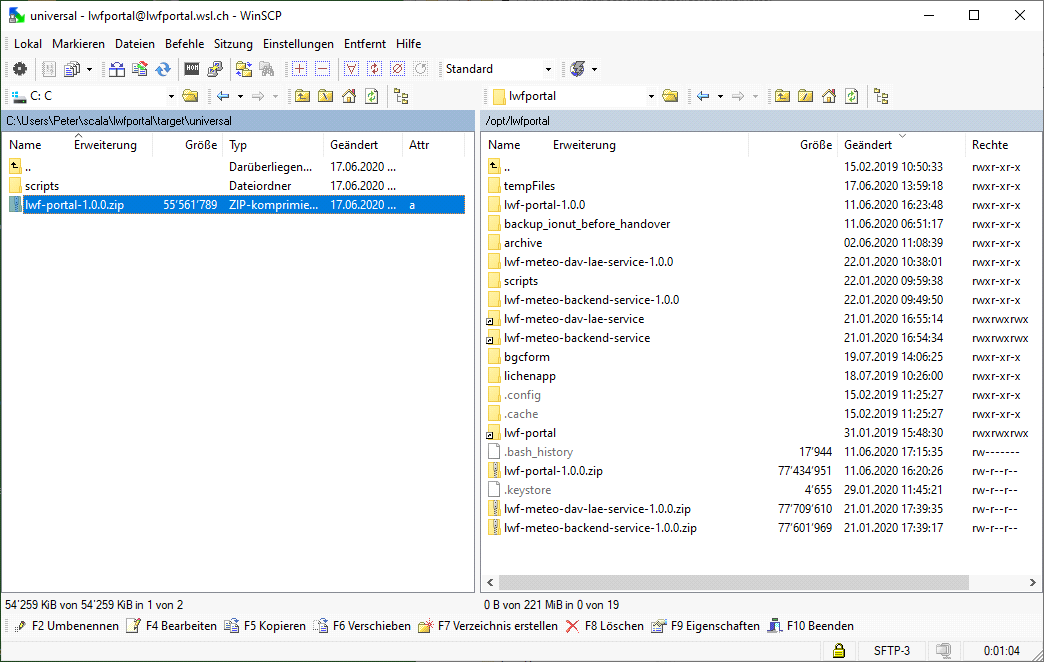
**sftp transfer of \*.zip**

Start WinSCP and make an sftp connection (Port 22) from local project folder, e.g. C:\...\lwfportal to the virtual machine with server:'lwfportal.wsl.ch' with username: 'lwfportal' and password: '\*\*\*\*\*\*\*'





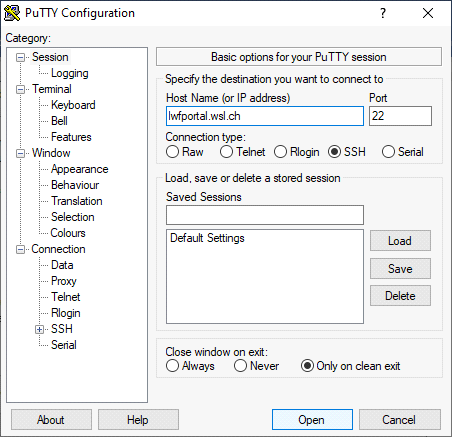
Copy the file C:\...\lwfportal\target\universal\lwf-portal\*.zip to lwfportal.wsl.ch/opt/lwfportal



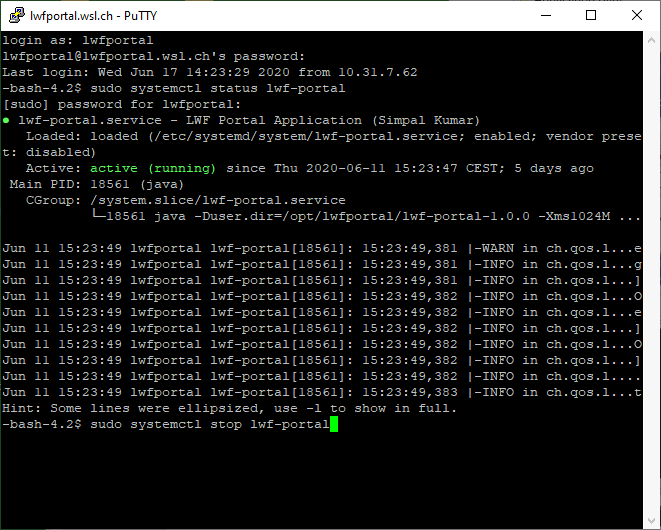
**SSH connection to start/stop the service and launch the deployment scripts**

The virtual machine lwfportal.wsl.ch has been set up by Markus Reinhard. He also adapted the deployment scripts to the settings of the machine.

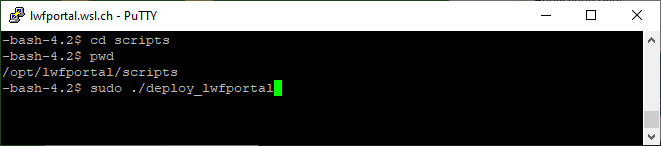
Start Putty and open an ssh connection (Port 22) to lwfportal using server: lwfportal.wsl.ch, username: lwfportal and password: '\*\*\*\*\*\*\*'



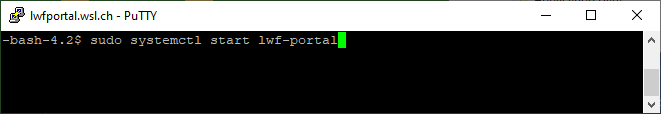
check the status of the service with 'sudo systemctl status lwf-portal' and stop it



Deploy the changes to the virtual machine by launching the deployment script /opt/lwfportal/scripts/deploy\_lwfportal with superuser privileges

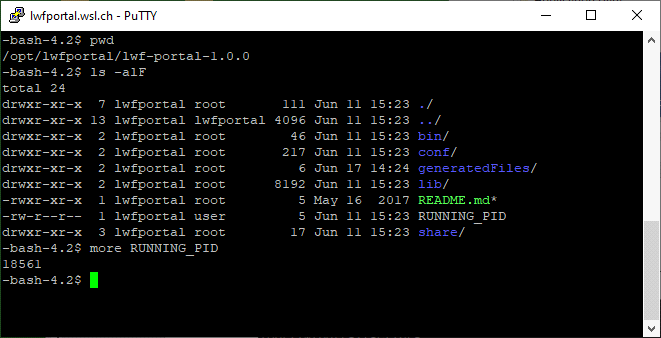


Start the service



## Check the running Application

(1) Check that the date of the file /opt/lwfportal/lwf-portal\*/RUNNING\_PID. If this file has been updated, this indicates that the start of the application/service was successful.



(2) Check the status with 'sudo systemctl status lwf-portal

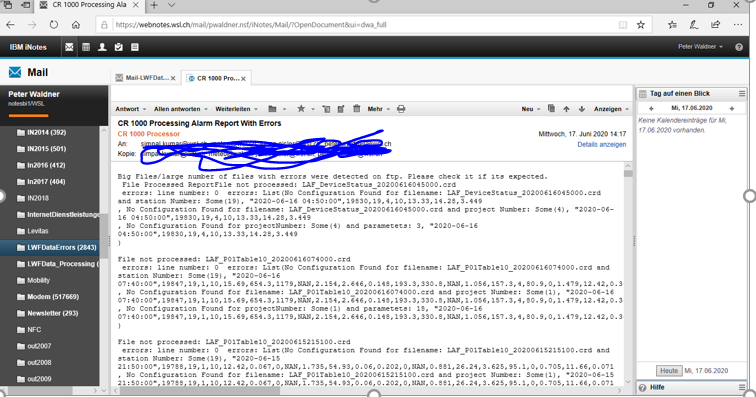
see above.

(3) Check the Weblogs

cd /weblog/

tail -50 application\_lwfportal.log

(4) Check the emails from 'cr1000'



(5) if errors occur, open an ftp connection to <ftp://lwfcr1000@ftp.wsl.ch> e.g. with FileZilla

Check if the structure of the files is in accordance with the

- Akka station configuration in /conf/application.conf

- DB station configuration in the Oracle-Tables METEO.STATKONF

(6) to solve problems, the logfiles can be consulted under ... (see logback.xml)

### Security remarks

The application password(secret) contained in the source file /application.conf is asked when the application is beeing started. The password is then supplied by the starting script on the virtual machine. Hence, make sure that the application password in the starting script on the virtual machine, i.e. lwfportal.wsl.ch/opt/lwfportal/scripts/deploy\_lwfportal is the same as the application password in the scala/java source code.

/etc/system/system …

### Compatibility remarks

The version of Java Runtime Environment (Version 1.8), Java Development Kit (Version 1.8), Scala (Version 2.12.1), the Scala Built Tools (Version 0.13.13.1), the Oracle JODBC Driver (Version 6) and the Oracle Database (Version 10.2) have to be compatible.

# Annex: Scala

„Prof. Martin Odersky from EPFL(https://lampwww.epfl.ch/~odersky/) is the author of scala and he might have contact with other authors e.g. Paul Chiusano, Runar Bjarnasan who wrote the book "Functional programming in scala", William E. Wheeler writer of "The definitive guide to functional programming for scala and java 8 developer" to answer for scala and java versions. In case if one would like to get in direct touch with Paul Chiusano, his github account is https://github.com/runarorama For sbt tool related questions you may write: https://github.com/sbt

Once you are satisfied and find scala interesting enough then you may think of scala-days (https://scaladays.org/) or scala-enthusiasts meetup https://www.meetup.com/de-DE/Zurich-Scala/ …“ (Simpal Kumar, June 2020)