Atitit classload util 功能总结

## 建立自己对classloader

List li\_libdir=Lists.*newArrayList*();

li\_libdir.add("D:\\libs\_junbo");

String classpath = "D:\\apiJubo\\api\\common\\target\\classes";

List li\_classDir=Lists.*newArrayList*();

li\_classDir.add(classpath);

URLClassLoader myClassLoader = *getURLClassLoader*(li\_libdir,li\_classDir);

private static URLClassLoader getURLClassLoader(List<String> **li\_libdir**, List<String> **li\_classDir**) {

List<URL> li\_urls=Lists.*newArrayList*();

for (String libpath : **li\_libdir**) {

File file\_directory = new File(libpath);

File[] fileNames =file\_directory.listFiles();

// URL urls[] = new URL[fileNames.length];

for (int i = 0; i < fileNames.length; i++) {

try {

li\_urls.add(fileNames[i].~~toURL~~());

} catch (MalformedURLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

for (String clsD : **li\_classDir**) {

try {

li\_urls.add(new File(clsD).~~toURL~~());

} catch (MalformedURLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

URL[] urls=li\_urls.toArray(new URL[li\_urls.size()]) ;

URLClassLoader loader = new URLClassLoader(urls,null);

return loader;

// return null;

}

## 动态新增jar或者classdir 加载

New urlclassloader,父亲loader指向上一级的。。。

## 从jar或者class目录 load class

public static Class<?> loadClass(String **name**, ClassLoader **classLoader**, boolean **isInitialized**)

cn.hutool.core.util.ClassLoaderUtil.loadClass(jarOrDir, name)