```
package com.sxt.test;

import java.io.ByteArrayOutputStream;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.InputStream;

/**
 * 自定义文件系统类加载器
 * @author 江
 *
 */

public class FileSystemClassLoader extends ClassLoader{
        private String rootDir;

        public FileSystemClassLoader(String rootDir) {
                this.rootDir=rootDir;
        }

        @Override
        protected Class<?> findClass(String name) throws ClassNotFoundException
{
           Class c=findLoadedClass(name);

           //应该要先查询有没有加载过这个类。如果已经加载，则直接返回加载好的
类。如果没有，则加载新的类。
           if(c!=null) {
                   return c;
           }else {
                   ClassLoader parent=this.getParent();
                   try {
                       c=parent.loadClass(name);        //委派给父类加载器
                   }catch(Exception e) {
```

```
                }
                if(c!=null) {
                        return c;
                }else {
                        byte[] classData=getClassData(name);
                        if(classData==null) {
                                throw new ClassNotFoundException();
                        }else {
                                c=defineClass(name, classData,
0,classData.length);
                        }
                }
            }
            return c;
        }

        private byte[] getClassData(String classname) {
                //com.sxt.test.User  -->  d:/myjava/com/sxt/test/User.class
                String path=rootDir+"/"+classname.replace('.','/')+".class";

                //IOUtils,可以使用它将流中的数据转成字节数组
                InputStream is = null;
                ByteArrayOutputStream baos=new ByteArrayOutputStream();
                try {
                        is=new FileInputStream(path);
                        byte[] buffer=new byte[1024];
                        int temp;
                        while((temp=is.read(buffer))!=-1) {
                                baos.write(buffer,0,temp);
                        }
                        return baos.toByteArray();
                } catch (Exception e) {
                        e.printStackTrace();
                        return null;
```

```
            }finally {
                        try {
                                if(is!=null) {
                                is.close();
                                }
                        } catch (IOException e) {
                                e.printStackTrace();
                        }

            }

        }

}
```