

21-设置Fix Version/s后自动把结束时间写在Due Date上

```
import com.atlassian.jira.component.ComponentAccessor
import com.atlassian.jira.datetime.DateTimeFormatter
import com.atlassian.jira.datetime.DateTimeStyle
import com.atlassian.jira.project.version.Version

if (underlyingIssue) {
    return;
}

def fixVersionsCf = getFieldById("fixVersions");
def dueDateCf = getFieldById("duedate");
if (fixVersionsCf == null || dueDateCf == null) {
    return;
}

def dueDateValue = dueDateCf.getFormValue();

def fixVersionsValue = fixVersionsCf.getFormValue()

log.info("dueDateValue: [" + dueDateValue + "]")
log.info("fixVersionsValue: [" + fixVersionsValue + "]")

if (fixVersionsValue) {
    def latestDate = getLatestDate(fixVersionsValue);
    log.info("latestDate: " + latestDate)
    //没有设置版本时间把duedate置空
    if (!latestDate) {
        // dueDate.setFormValue(null)
        return;
    }
    //获得日期格式转化 类
    DateTimeFormatter dateFormatter = ComponentAccessor.getComponent(
        DateTimeFormatter.class);
    //根据当前操作人获得它的日期格式
    dateFormatter = dateFormatter.forLoggedInUser();
    dueDateValue = dateFormatter.withStyle(DateTimeStyle.DATE_PICKER).format(
        latestDate as Date);
    log.info("dueDateValue: " + dueDateValue)
    dueDateCf.setFormValue(dueDateValue)
} else {
    // dueDate.setFormValue(null)
}
```

//获取多个版本中最迟的完成时间

```
def getLatestDate(versions) {  
    if(!(versions instanceof List)) {  
        return ComponentAccessor.getVersionManager().getVersion(versions as Long).  
getReleaseDate();  
    }  
    def sortArray = [];  
    for(x in versions) {  
        log.debug("xx:"+x)  
        def version = ComponentAccessor.getVersionManager().getVersion(x as Long);  
        if(version == null) {  
            continue;  
        }  
        if(version.getReleaseDate()) {  
            sortArray.add(version.getReleaseDate());  
        }  
    }  
    return sortArray.max();  
}
```