[**http://www.cnblogs.com/feika/p/4448573.html**](http://www.cnblogs.com/feika/p/4448573.html)

# 1.使用经验

## 1.解释，下文中先将得到的字符串转为Date ，再将得到的Date转为LocalDateTime

@PutMapping(**"/largeOrder/updateAudit"**)  
**public** String updateAudit(String id, String auditState, Date auditTime, String auditPerson ) {  
 **int** updateStatus = **largeOrderService**.updateLargeOrderAudit(id,auditState,**this**.DateToLocalDateTime(auditTime), auditPerson);  
 **if**(updateStatus!=0){  
 **return "ok"**;  
 }  
 **return null**;  
}  
*//将url字符串日期转化为Date 类型*@InitBinder  
**public void** initBinder(ServletRequestDataBinder binder){  
 binder.registerCustomEditor(Date.**class**, **new** CustomDateEditor(**new** SimpleDateFormat(**"yyyy-MM-dd"**), **true**));  
}  
*//将Date转化为LocalDateTime***public** LocalDateTime DateToLocalDateTime(Date auditTime){  
 Instant instant = Instant.*ofEpochMilli*(auditTime.getTime());  
 LocalDateTime res = LocalDateTime.*ofInstant*(instant, ZoneId.*systemDefault*());  
 **return** res;  
}

## [Convert Date between LocalDateTime](http://www.cnblogs.com/feika/p/4448573.html)

http://blog.progs.be/542/date-to-java-time

Java8 has new date and time classes to “replace” the old not-so-beloved java.util.Date class.

Unfortunately though, converting between the two is somewhat less obvious than you might expect.

## Convert java.util.Date to java.time.LocalDateTime

|  |
| --- |
| Date ts = ...;  Instant instant = Instant.ofEpochMilli(ts.getTime());  LocalDateTime res = LocalDateTime.ofInstant(instant, ZoneId.systemDefault()); |

The big trick (for all these conversions) is to convert to Instant. This can be converted to LocalDateTime by telling the system which timezone to use. This needs to be the system default locale, otherwise the time will change.

## Convert java.util.Date to java.time.LocalDate

|  |
| --- |
| Date date = ...;  Instant instant = Instant.ofEpochMilli(date.getTime());  LocalDate res = LocalDateTime.ofInstant(instant, ZoneId.systemDefault()).toLocalDate(); |

## Convert java.util.Date to java.time.LocalTime

|  |
| --- |
| Date time = ...;  Instant instant = Instant.ofEpochMilli(time.getTime());  LocalTime res = LocalDateTime.ofInstant(instant, ZoneId.systemDefault()).toLocalTime(); |

## Convert java.time.LocalDateTime to java.util.Date

|  |
| --- |
| LocalDateTime ldt = ...;  Instant instant = ldt.atZone(ZoneId.systemDefault()).toInstant();  Date res = Date.from(instant); |

## Convert java.time.LocalDate to java.util.Date

|  |
| --- |
| LocalDate ld = ...;Instant instant = ld.atStartOfDay().atZone(ZoneId.systemDefault()).toInstant();  Date res = Date.from(instant); |

## Convert java.time.LocalTime to java.util.Date

|  |
| --- |
| LocalTime lt = ...;  Instant instant = lt.atDate(LocalDate.of(A\_YEAR, A\_MONTH, A\_DAY)).  atZone(ZoneId.systemDefault()).toInstant();  Date time = Date.from(instant); |

This one is a little, funny, you need to inject a date to convert the time… Gives you the option of start of epoch or something else.

# 2.String 转化为LocalDateTime

*/\*\*  
 \* 用于完成string和localdatetime的格式转化  
 \*  
 \* Created by HealerJean on 2017/4/10.  
 \*/*@Service  
**public class** DateFormatUtil {  
 */\*\*  
 \*  
 \** ***@param dateString*** *string格式的日期  
 \** ***@param datetype*** *日期格式 比如 yyyy-MM-dd  
 \** ***@return*** *\*/* **public** LocalDateTime getLocalDateTime(String dateString,String datetype) {  
  
 SimpleDateFormat formatter = **new** SimpleDateFormat(datetype.trim());  
 **try** {  
 Date date = formatter.parse(dateString.trim());  
 Instant instant = Instant.*ofEpochMilli*(date.getTime());  
 LocalDateTime res = LocalDateTime.*ofInstant*(instant, ZoneId.*systemDefault*());  
 **return** res;  
  
 }**catch** (ParseException e){  
 e.printStackTrace();  
 }  
 **return null**;  
 }  
  
 **public static void** main(String[] args) {  
 DateFormatUtil dateFormatUtil = **new** DateFormatUtil();  
 System.***out***.println(dateFormatUtil.getLocalDateTime(**"2017-04-04"**,**"yyyy-MM-dd"**));  
  
 }  
}

# 3.String 转化为LocalDateTime

LocalDateTime fairStartTime = LocalDateTime.*parse*(recCareerEventDTO.getFairStartTime());

# 4.String 转化为LocalDate

LocalDate.*parse*(studentInformationDTO.getBirthdate()

# 5 在ajax 接收date对象时候的转化

## 1.如果不转化



## 2.转化方法

<**script th:inline="JavaScript"**>  
 **var *auditTime*** = [[${reassignApplication.auditTime}]];  
 **if** (***auditTime***!=**null**) {  
 **var *auditTimeString*** = ***auditTime***.**year** + **'-'** + ***auditTime***.monthValue + **'-'** + ***auditTime***.dayOfMonth + **' '** + ***auditTime***.hour + **':'** + ***auditTime***.minute;  
 **$**(**'#auditTime'**).val(***auditTimeString***);  
 }  
</**script**>

# 6. 根据日期生成编号

*//生成编号*DateTimeFormatter fmt = DateTimeFormatter.*ofPattern*(**"yyyyMMddHHmmssSSS"**);  
String newsNo = **"xw"**+LocalDateTime.*now*().format(fmt);

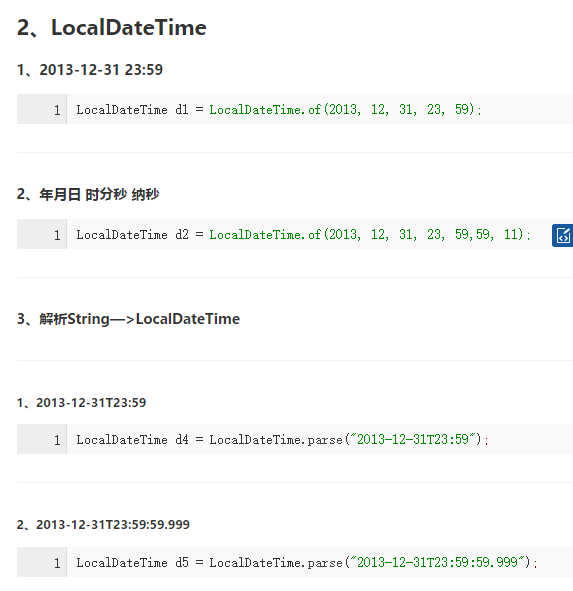
SimpleDateFormat df = **new** SimpleDateFormat(**"yyyyMMddHHmmss"**);*//设置日期格式*

String date = df.format(**new** Date());*// new Date()为获取当前系统时间，也可使用当前时间戳*

# 7.LocalDate



# 8.LocalDateTime



# 9. ****其他格式的String Date 转化为LocalDateTime****

|  |
| --- |
| /\*\*  \* @Description 目前传来的日期的格式是 yyyy-MM-dd HH:mm:ss ，此时不能转化LocalDateTime，所以需要转化  \* @Author HealerJean  \* @CreateDate 2017/5/25 15:13  \*/  private LocalDateTime getLocalDateTime(String strDate) {  try {  SimpleDateFormat formatter = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");  Date date = formatter.parse(strDate);  Instant instant = Instant.ofEpochMilli(date.getTime());  LocalDateTime localDateTime = LocalDateTime.ofInstant(instant, ZoneId.systemDefault());  return localDateTime;  } catch (ParseException e) {  e.printStackTrace();  }  return null;  } |