**Calendar**

**1. Lợi thế của Calendar so với Date:**

Do lớp Date có nhiều hành vi đã lỗi thời và không còn hỗ trợ được nhiều nữa nên lớp Calendar được ngày càng sử dụng rộng rãi. Với Calendar việc get và set cho từng thành phần như: ngày, tháng, năm. Giờ phút giây trở lên đơn giản vì Calendar hỗ trợ sẵn những hàm get và set nên việc thao tác với những thành phần này trở lên đơn giản hơn khi sử dụng Date.

Calendar là một lớp trừu tượng nên không có constructor để khởi tạo. Nhưng Calendar cung cấp 2 methor tĩnh để tạo đối tượng calendar.

- getInstance();// Lấy ra đối tượng Calendar mô tả thời điểm hiện tại với locale và timezone hiện tại.

- getInstance(TimeZone zone, Locale locale);// lấy đối tượng Calendar mô tả thời điểm ta xét cho calendar đó.

**2. Calendar có 3 lớp con**:

- Gregorian Calendar: dương lịch, lịch hiện đại.

- Buddhist Calendar: bộ lịch phật giáo, thường được sử dụng tại các nước Đông Nam Á, hiện tại không được sử dụng 1 cách chính thức.

- Japanese Imperial Calendar: lịch truyền thống nhật bản, hiện không sử dụng 1 cách chính thức.

**3. Những phương thức get và set cho từng thành phần trong Calendar**

|  |  |
| --- | --- |
| **Phương thức get(int)** | **Giá trị trả về** |
| get(Calendar.DAY\_OF\_WEEK) | 1 (Calendar.SUNDAY) tới 7 (Calendar.SATURDAY). |
| get(Calendar.YEAR) | Năm (year) |
| get(Calendar.MONTH) | 0 (Calendar.JANUARY) tới 11 (Calendar.DECEMBER). |
| get(Calendar.DAY\_OF\_MONTH) | 1 tới 31 |
| get(Calendar.DATE) | 1 tới 31 |
| get(Calendar.HOUR\_OF\_DAY) | 0 tới 23 |
| get(Calendar.MINUTE) | 0 tới 59 |
| get(Calendar.SECOND) | 0 tới 59 |
| get(Calendar.MILLISECOND) | 0 tới 999 |
| get(Calendar.HOUR) | 0 tới 11, được sử dụng cùng với Calendar.AM\_PM. |
| get(Calendar.AM\_PM) | 0 (Calendar.AM) hoặc 1 (Calendar.PM). |
| get(Calendar.DAY\_OF\_WEEK\_IN\_MONTH) | DAY\_OF\_MONTH 1 tới 7 luôn luôn tương ứng với DAY\_OF\_WEEK\_IN\_MONTH 1;  8 tới 14 tương ứng với DAY\_OF\_WEEK\_IN\_MONTH 2, ... |
| get(Calendar.DAY\_OF\_YEAR) | 1 tới 366 |
| get(Calendar.ZONE\_OFFSET) | Giá trị GMT của múi giờ. |
| get(Calendar.ERA) | Biểu thị AD (GregorianCalendar.AD), BC (GregorianCalendar.BC). |

Chuyển đổi từ Date sang Calendar thông qua phương thức setTime(date)

Date date = new Date();

Calendar c = Calendar.getInstance();

c.setTime(date);

Chuyển đổi từ Calendar qua Date thông qua phương thức getTime()

4. Gregorian Calendar: Lịch hiện đại thường được sử dụng nhất hiện nay

4.1 Constructor:

- Gregorian cung cấp 2 constructor:

+ GregorianCalendar(“trong đây có thể truyền tham số ngày tháng năm”): sẽ thiết lập ngày tháng năm theo như ta xét, nếu không xét sẽ lấy ngày tháng năm hiện tại và sẽ dùng zone mặc định và locale mặc định.

+ GregorianCalendar(Timezone zone, Locale alocale): xây dụng Gregorian với thời gian hiện tại và có zone cung cấp và locale cung cấp.

4.2 Những phương thúc của GregorianCalendar:

|  |  |
| --- | --- |
| **STT** | **Phương thức và miêu tả** |
| 1 | **void add(int field, int amount)**  Thêm số time đã xác định tới trường time đã cung cấp, dựa trên các qui tắc của calendar đó. |
| 2 | **protected void computeFields()**  Biến đổi UTC dạng mili giây thành các giá trị trường time. |
| 3 | **protected void computeTime()**  Override các giá trị biến đổi trường time thành UTC dạng mili giây. |
| 4 | **boolean equals(Object obj)**  So sánh GregorianCalendar với một tham chiếu đối tượng |
| 5 | **int get(int field)**  Nhận giá trị cho một trường time đã cung cấp |
| 6 | **int getActualMaximum(int field)**  Trả về giá trị lớn nhất mà trường này có thể có, đã cung cấp date hiện tại |
| 7 | **int getActualMinimum(int field)**  Trả về giá trị nhỏ nhất mà trường này có thể có, đã cung cấp date hiện tại |
| 8 | **int getGreatestMinimum(int field)**  Trả về giá trị tối thiểu cao nhất cho trường đã cung cấp nếu nó biến đổi |
| 9 | **Date getGregorianChange()**  Nhận date thay đổi theo Gregorian Calendar |
| 10 | **int getLeastMaximum(int field)**  Trả về giá trị tối đa thấp nhất cho trường đã cung cấp nếu nó biến đổi |
| 11 | **int getMaximum(int field)**  Trả về giá trị tối đa cho trường đã cung cấp |
| 12 | **Date getTime()**  Nhận time hiện tại của Calendar này |
| 13 | **long getTimeInMillis()**  Nhận time hiện tại của Calendar này dạng long |
| 14 | **TimeZone getTimeZone()**  Nhận time zone. |
| 15 | **int getMinimum(int field)**  Trả về giá trị tối thiểu cho trường đã cung cấp |
| 16 | **int hashCode()**  Override mã hóa băm. |
| 17 | **boolean isLeapYear(int year)**  Xác định nếu năm đã cho là một leap year |
| 18 | **void roll(int field, boolean up)**  Cộng hoặc trừ một đơn vị đơn của time trên trường time đã cho mà không thay đổi các trường lớn hơn |
| 19 | **void set(int field, int value)**  Thiết lập trường time với giá trị đã cho |
| 20 | **void set(int year, int month, int date)**  Thiết lập các giá trị cho các trường year, month, và date. |
| 21 | **void set(int year, int month, int date, int hour, int minute)**  Thiết lập các giá trị cho các trường year, month, date, hour, và minute. |
| 22 | **void set(int year, int month, int date, int hour, int minute, int second)**  Thiết lập các giá trị cho các trường year, month, date, hour, minute và second. |
| 23 | **void setGregorianChange(Date date)**  Thiết lập ngày thay đổi theo GregorianCalendar |
| 24 | **void setTime(Date date)**  Thiết lập time hiện tại của Calendar này với Date đã cho |
| 25 | **void setTimeInMillis(long millis)**  Thiết lập time hiện tại của Calendar này từ giá trị long đã cho |
| 26 | **void setTimeZone(TimeZone value)**  Thiết lập time zone với giá trị time zone đã cho |
| 27 | **String toString()**  Trả về một biểu diễn chuỗi của calendar này |

\* lưu ý month lấy trả về hoặc nhận từ 0->11