Annexe 11 - Notes sur les classes Calendar, GregorianCalendar et SimpleDateFormat

## Classes Calendar, GregorianCalendar

* encapsule en un objet les différents champs ( jour, mois, année, heure… ) relatifs à une date
* font partie du package java.util

## Classe Calendar

* il s’agit d’une classe abstraite, mettant en œuvre deux interfaces
* il n’est donc pas possible de créer d’objet Calendar à l’aide de new
* une sous-classe GregorianCalendar existe : on peut donc créer des objets de cette sous-classe :

**GregorianCalendar gc = new GregorianCalendar();**

* D'une autre façon, une méthode statique getInstance permet de créer un objet GregorianCalendar à partir de la classe abstraite Calendar, comme ceci :

**Calendar c = Calendar.getInstance();**

méthode statique créant un objet GregorianCalendar contenant la date actuelle du système

Polymorphisme :

Un GregorianCalendar est un Calendar

Nom de référence de l’objet comprenant les différentes variables relative à la date

## Classe GregorianCalendar

* Créer un objet contenant la date actuelle du système ( similaire à getInstance ) :

GregorianCalendar present = new GregorianCalendar();

* Créer un objet contenant une date quelconque :

GregorianCalendar pass = new GregorianCalendar (1990,1,6);

* représente le 6 février 1990 **\*\*\* valeur du mois commence à 0 !!!**
* Pour aller chercher un attribut / variable de l’objet : méthode get avec paramètre représentant la variable

pass.get(Calendar.DAY\_OF\_MONTH) == 6

pass.get( Calendar.YEAR ) == 1990

pass.get ( Calendar.DAY\_OF\_YEAR ) == 37 ( 37 ème jour de l’année )

* Pour modifier un attribut de l’objet : methode set

pass.set( Calendar.MONTH, 3 ) 🡪 change le mois de février à avril

pass.set( 2000, 2, 13 ) 🡪 modifie année, mois, jour

* Pour augmenter ou diminuer la valeur d’un attribut, méthode add

present.add( Calendar.DAY\_OF\_MONTH, 2 ) 🡪 on est rendus 2 jours plus tard

* Pour savoir si un objet Calendar est chronologiquement avant un autre : méthode before

present.before(pass) 🡪 retourne false

## Classe SimpleDateFormat

* utile lorsqu’on veut afficher des dates contenues dans un objet GregorianCalendar
* fonctionne en créant un modèle qui sera passé dans l’appel du constructeur du SimpleDateFormat ( comme avec le DecimalFormat )
* Les caractères principaux utilisés dans le modèle sont :

|  |  |  |
| --- | --- | --- |
| Caractères | Correspondant à… | Exemple |
| yyyy | Année | 1989 |
| MMM ( 3 ou plus ) | Mois dans l’année ( en lettres ) | Juillet |
| MM ( 2 caractères ) | Mois dans l’année ( en chiffres ) | 07 |
| dd | Jour dans le mois | 22 |
| HH | Heure dans le jour ( 00 – 23 ) | 15 |
| mm | Minutes | 56 |
| ss | Secondes | 43 |

* il existe d’autres variantes : voir l’API

Exemple :

GregorianCalendar gc = new GregorianCalendar();

SimpleDateFormat sdf = new SimpleDateFormat("dd/MMMM/yy");

System.out.println ( sdf.format( gc.getTime()));

* 7/octobre/21 ( aujourd’hui )