

# UE SIH

## *Sélection de cours, documentations et tutoriels*

Emmanuel Promayon  
Emmanuel.Promayon@imag.fr

# Introduction

- Objectif : compléter vos connaissances techniques
- Ce document rassemble de nombreuses références
  - Orientées auto-formation
  - Majoritairement sous forme de cours en lignes et tutoriels
- Les liens sont cliquables



# Introduction

- Les références portent sur
  - Passage Modélisation UML → BD relationnelle (SQL)
  - Connexion à une BD en Java par JDBC
  - Programmation d'interface utilisateur avec Java Swing
  - Paramétrage / Configuration / Personnalisation d'applications Java à l'aide de "properties"



# Plan

- Introduction
- UML → SQL
- JDBC
- Swing et MVC
- Properties



# Cours D. Donsez

## ■ Cours de Didier Donsez

- Explique le passage d'un diagramme de classe à un schéma relationnel pour créer une BD équivalente
- Ce cours utilise aussi le formalisme du modèle entité-association (différent d'UML), très utilisé pour la description de bases de données
- <http://membres-liglab.imag.fr/donsez/cours/eaumlrel.pdf>



# Tutorial Sparx Systems

- Comment passer d'UML à une BD relationnelle
- Cette entreprise propose un produit (sous licence propriétaire et payant) de modélisation UML dont un module permet de passer d'UML à une BD relationnelle, le tutorial ci-dessous explique comment faire avec un exemple
- [http://www.sparxsystems.com/resources/uml\\_datamodel.html](http://www.sparxsystems.com/resources/uml_datamodel.html)



# Cours T. Jewett

## ■ Cours de Tom Jewett

- Cours complet permettant de bien comprendre l'articulation et le passage de UML vers une base de données relationnelle
- Très pédagogique et progressif, ce tutoriel est conçu pour être lu et compris par auto-apprentissage
- <http://www.tomjewett.com/dbdesign>



# Design pattern

- Design pattern DAO (Data Access Object)
  - Patron de conception permettant de construire des classes Java dont les instances sont le reflet des informations de la BD
  - Inconvénient : très technique
  - Avantages : propre, réutilisable, savoir-faire (CV)
  - <http://www.oracle.com/technetwork/java/dataaccessobject-138824.html>
  - <http://jingdao.sourceforge.net/related.html>
  - <http://cyrille-herby.developpez.com/tutoriels/java/mapper-sa-base-donnees-avec-pattern-dao/>  
(en français)



# Modélisation BD en UML

- Présentation d'un profil UML pour la modélisation de BD
  - Explique comment utiliser le formalisme UML de façon à représenter une base de données
  - Ce n'est qu'une façon de faire, pas forcément nécessaire dans le cadre de ce projet
  - <http://www.agiledata.org/essays/umlDataModelingProfile.html>



# Plan

- Introduction
- UML → SQL
- JDBC
- Swing et MVC
- Properties



# Tutoriels Oracle

- Connexion entre une base de données est SQL
  - Les tutoriels sont fortement conseillés, car il s'agit d'acquérir des connaissances techniques
- JDBC = Java DataBase Connectivity



# Tutoriels Oracle

- Oracle proposent de nombreux cours, notamment
- Tutoriel
  - <http://download.oracle.com/javase/tutorial/jdbc/basics/index.html>



# Tutoriels Netbeans

- Tutoriels Netbeans (mysql)
  - <http://www.netbeans.org/kb/docs/ide/mysql.html>
- Tutoriels Netbeans (derby javadb)
  - <http://www.netbeans.org/kb/docs/ide/java-db.html>
- Autre Netbeans/mysql (non officiel)
  - <http://www.linglom.com/2007/12/05/accessing-mysql-on-netbeans-using-jdbc-part-i-create-a-connection/>



# Pour Netbeans 6.1

- Pour netbeans 6.1 (salles 048/050)
  - Téléchargez l'archive des documentations  
<http://netbeans.org/projects/usersguide/downloads/download/NB61-docs.zip>
  - Le tutoriel pour mysql est [kb/61/ide/mysql.html](http://netbeans.org/projects/usersguide/downloads/download/NB61-docs.zip)

(pour derby, il faut regarder dans l'archive des documentations 6.0)



# Plan

- Introduction
- UML → SQL
- JDBC
- Swing et MVC
- Properties



# Swing

- Java Swing = plateforme pour le développement d'interface utilisateurs
- Utilise le design pattern Model-View-Controller
- Revoir
  - TIS3 Architecture Logicielle (cours et TP)
- Tutoriels Oracle
  - <http://download.oracle.com/javase/tutorial/uiswing/>
- Autre (en français, non officiel)

<http://baptiste-wicht.developpez.com/tutoriel/java/swing/debutant/>



# MVC

- Un petit tutoriel expliquant l'utilisation concrète du design pattern MVC avec Swing
  - <http://www.cs.wcupa.edu/~rkline/Java/mvc-example.html>
- Deux autres URL concernant le sujet
- <http://www.oracle.com/technetwork/articles/javase/mvc-136693.html>
- <http://www.oracle.com/technetwork/java/architecture-142923.html>



# Plan

- Introduction
- UML → SQL
- JDBC
- Swing et MVC
- Properties



# Configuration / Personnalisation

- Pour effectuer la personnalisation, on peut par exemple utiliser des fichiers de configuration « .properties »
- Documentation Oracle
  - <http://download.oracle.com/javase/1.5.0/docs/api/java/util/Properties.html#loadFromXML%28java.io.InputStream%29>
- On peut aussi envisager une solution utilisant un langage XML propre à l'application
  - Utilisation de JAXP ou JAXB (voir cours TIS4 XML)

