

JEE

Dr Maurice D. FAYE

faye.maurice-djibril@ugb.edu.sn

Documentation (2017)

- Liens pour documentation : JEE7, GlassFish, Netbean
 - <https://docs.oracle.com/javaee/7/tutorial/preface.htm#GEXAF>
- connecter serveur Mysql à netbeans
 - <https://netbeans.org/kb/docs/ide/mysql.html>
- JEE7 tutorial
 - <https://docs.oracle.com/javaee/7/JEETT.pdf>
- <http://java.sun.com/JEE/>
- <http://jean-luc.massat.perso.luminy.univ-amu.fr/ens/jee/index.html>
- <https://openclassrooms.com/courses/creez-votre-application-web-avec-java-ee>
- <https://www.mistra.fr/tutoriels-java/tutoriel-jee.html>

Documentation (2017)

- Erreur serveur GlassFish
 - <https://javaee.github.io/glassfish/doc/4.0/release-notes.pdf>
 - https://docs.oracle.com/cd/E26576_01/index.htm

Introduction

- JEE ⇒ Java Platform, Enterprise Edition
 - Anciennement J2EE (jugé lourd, complexe)
 - Une spécification qui a été implémentée par divers entreprises
 - Définit des normes, des types de ressources, leurs interactions,...
 - Une architecture, un ensemble de technologies
 - Développement d'applications réparties robustes

Introduction

- Les applications JEE
 - adhèrent aux standards JEE
 - suivent les spécifications JEE
 - sont écrites en Java
 - sont déployables sur tous les serveurs implémentant JEE
- Versions actuelles :
 - JEE 8 .
 - Oracle ⇒ JEE Open source.
 - <http://www.oracle.com/technetwork/java/javaee/documentation/ee8-release-notes-3894362.html>
 - JDK 8

Architecture JEE

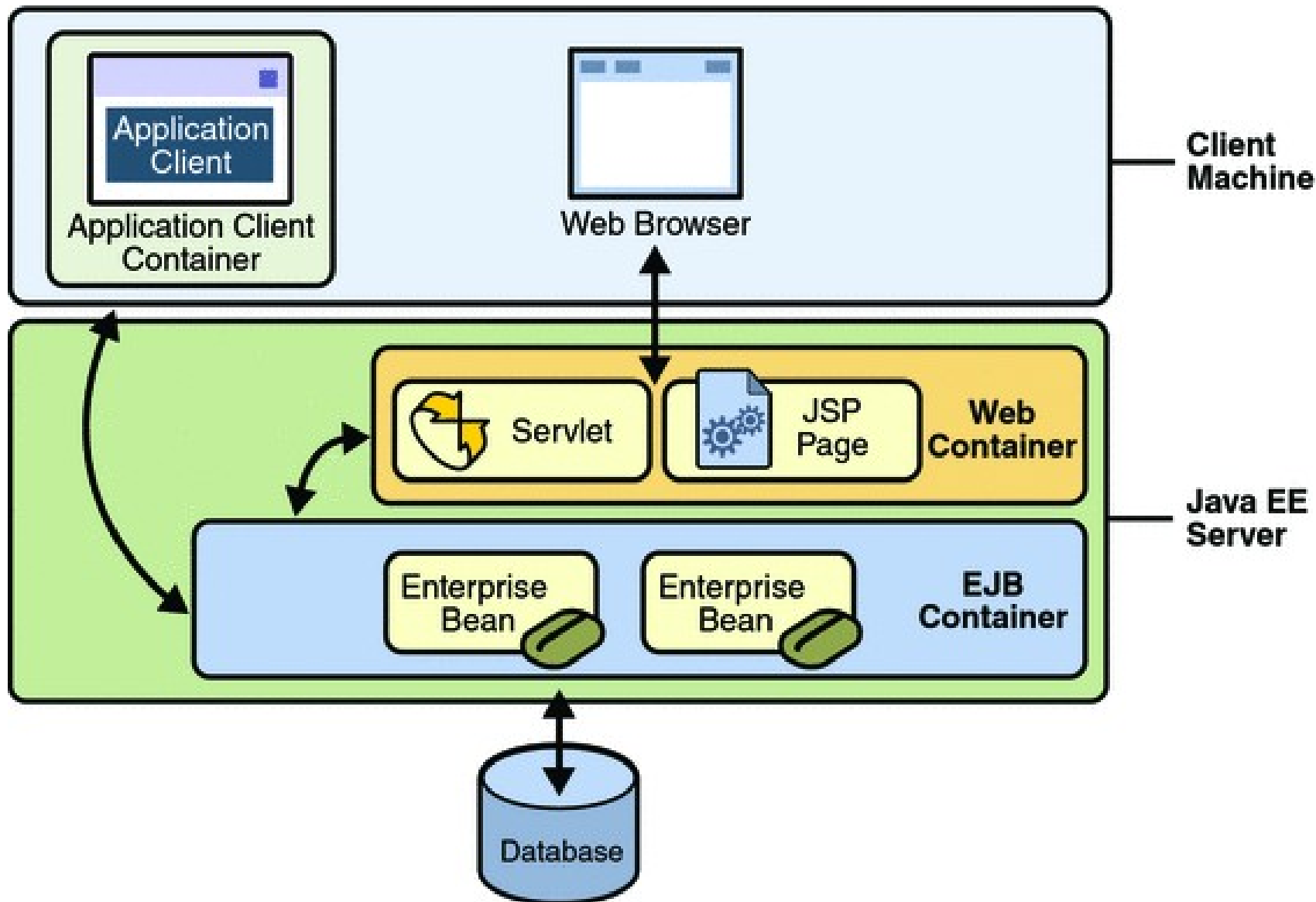
- Le modèle d'application JEE
 - Pour supporter les applications d'entreprise qui doivent inter-agir (fournir des services) avec divers partenaires, clients, fournisseurs
 - Ce type d'application est naturellement complexe, avec souvent le besoin d'utiliser des données provenant de diverses sources,
 - Nécessitent souvent une distribution (des données ou des acteurs)

Architecture JEE

- Le modèle d'application JEE
 - Implémente les services sous forme d'applications multi-tiers distribuées. Chaque tiers/couche/niveau gérant logiquement des fonctionnalités comparables :
 - Données, couche métier, présentation/client,...
 - Permet de concevoir des applications qui passent à l'échelle, qui sont disponibles et maintenables
 - Qualités essentielles pour des applications d'entreprise

Architecture JEE

<https://fernandofranzini.files.wordpress.com/2017/10/overview-serverandcontainers.gif?w=1400&h=9999>



Architecture JEE

Applications multi-tiers (généralement 3-tiers)

- Tiers Client
 - Applications clientes (navigateur, application java, autre serveur,...), pas forcément écrites en Java, qui envoient des requêtes au serveur JEE , qui les traite et leur renvoie les réponses. Clients et serveur ne sont généralement sur la même machine (applications distribuées).

Architecture JEE

Applications multi-tiers (généralement 3-tiers)

- Tiers web (web tiers)

- Composants pour la gestion des inter-actions entre clients et tiers métier
- Fonctions de base
 - Génération dynamique de contenu, en différents formats, pour les clients
 - Collecter les données d'entrée des utilisateurs de l'interface cliente, les transfère aux composants du tiers métier, et retourne aux clients les résultats appropriés en provenance du tiers métier
 - Contrôle le flux des écrans [application console] ou des pages[navigateur] pour les clients .
 - Maintient l'état des données pour une session utilisateur
 - Etc...
- Technologies principales du tiers web : Servlet, JavaServer Faces, Contexts and Dependency Injection for Java EE, Expression Language,...

Architecture JEE

Applications multi-tiers (généralement 3-tiers)

- Tiers métier (business tiers)
 - Composants qui fournissent la logique métier d'une application [logique métier ⇒ code qui fournit les fonctionnalités concrètes et dépend de l'application : banque, aéronautique, médecine,...]
 - L'implémentation des fonctionnalités de base [le métier] d'une application d'entreprise doit se faire à ce niveau.
 - Parmi les technologies principales du tiers métier :
 - Composant Enterprise JavaBeans (enterprise bean) / EJB
 - Services web de type JAX-RS (RESTful)
 - Les entités de l'API Java Persistence API (JPA)

Architecture JEE

Applications multi-tiers (généralement 3-tiers)

- Tiers données (data tiers / EIS=Enterprise Information System)
 - Composants pour la gestion des données : serveurs de base de données, autre sources de données de l'entreprise
 - Ces ressources sont généralement sauvegardées sur une autre machine que celle qui héberge le serveur JEE, et l'accès à ces ressources se fait à travers des composants du tiers métier.
 - Les technologies suivantes sont utilisées pour accéder au tiers données dans les applications JEE:
 - The Java Database Connectivity API (JDBC)
 - The Java Persistence API (JPA)
 - The Java EE Connector Architecture
 - The Java Transaction API (JTA)

Architecture JEE

Applications multi-tiers (généralement 3-tiers)

- Plate-forme JEE \Rightarrow tiers web et métier

Technologies JEE

- Enterprise JavaBeans Technology (EJB)
 - code java implémentant une logique métier. Brique pouvant fonctionner seul ou être utilisée avec d'autres beans. Bloc de base pour les applications JEE
- JPA (Java Persistence API) :
 - est une couche d'abstraction de la couche JDBC. Elle permet notamment de faire du Mapping Relationnel-Objet (ORM, Object-Relationnal Mapping en anglais)
- SERVLET
 - Classe java généralement utilisé pour étendre les applications hébergées par les serveurs web
- ==>

Technologies JEE

- JSF : JavaServer Faces Technology :
 - Framework pour construire des applications web
- JSP : JavaServer Pages Technology
 - Permet de construire des pages web dynamique
 - cf. JSTL
- JAX-RS : Java API for RESTful Web Services
 - services web selon l'architecture/modèle REST (Representational State Transfer)
- JMS : Java Message Service API
 - Standard permettant aux applications JEE de créer, envoyer des messages
- Etc ...

Technologies JEE

inclut dans la JSE

- **JDBC : Java Database Connectivity (JDBC) API**
 - Inclut dans la JSE
 - connexion avec les bases de données
- **JNDI : Java Naming and Directory Interface API**
 - Services de nommage et d'annuaires
- **JAXP : Java API for XML Processing**
 - Traitement document XML
- **JAX-WS : Java API for XML Web Services**
 - Spécification pour le support des services web
- **Etc...**

Outils à installer

- Installation (version combinées ou séparées).
 - Jdk 7 / openJdk 7 ou java_8_oracle : requis pour JSE ou JEE
 - <http://www.oracle.com/technetwork/java/javase/downloads/jdk-netbeans-jsp-142931.html>
 - <http://www.oracle.com/technetwork/java/javaee/downloads/index.html>
 - Netbeans 8.2 EE avec serveur Glassfish /tomcat
 - NB : les serveurs ne sont pas installés par défaut, il faut les choisir (cocher) au moment de l'installation , dans la fenêtre personnaliser/customize
 - Page télécharger : <https://netbeans.org/downloads/index.html>
 - Instructions installation : <https://netbeans.org/community/releases/82/install.html>
 - Mysql server
 - Installer mysql sous linux :
<https://www.digitalocean.com/community/tutorials/how-to-install-mysql-on-ubuntu-14-04>
 - Installer sous windows
- Les exemples du cours testés avec Netbeans 8.2 EE et java_8_oracle, mysql Server 5.5 ; java DB

Outils à installer

NetBeans IDE Download - Google Chrome

NetBeans IDE Download

https://netbeans.org/downloads/index.html

Applications Paramètres Guide pratique UQAM | Faculté Accueil - allAfrica Ministère de l'en 326 écoles public Afrique - BBC Af

Choose page language

NetBeans NetBeans IDE NetBeans Platform Plugins Docs & Support Community Partners

HOME / Download

NetBeans IDE 8.2 Download

8.1 | 8.2 | Development | Archive

Email address (optional):

Subscribe to newsletters: ☒ Monthly ☐ Weekly

☒ NetBeans can contact me at this address

IDE Language: **English** Platform: **Linux (x86/x64)**

Note: Greyed out technologies are not supported for this platform.

NetBeans IDE Download Bundles

Supported technologies *	Java SE	Java EE	HTML5/JavaScript	PHP	C/C++	All
NetBeans Platform SDK	•	•				•
Java SE	•	•				•
Java FX	•	•				•
Java EE		•				•
Java ME						•
HTML5/JavaScript		•	•	•		•
PHP			•	•		•
C/C++					•	•
Groovy						•
Java Card™ 3 Connected						—
Bundled servers						
GlassFish Server Open Source Edition 4.1.1		•				•
Apache Tomcat 8.0.27		•				•

Download Download Download x86 Download x86 Download x86 Download x64 Download x64 Download x64 Download

Serveurs GlassFish et Tomcat

- GlassFish
 - Implémentation complète de Java EE
 -
- Tomcat
 - Pas un serveur d'application complet [implémente une partie ==> web tier]
 - Implémentation open source des spécifications : Java Servlet, JavaServer Pages, Java Expression Language et Java WebSocket .

Installation serveur GlassFish

- Inclus dans Netbeans EE et all mais pas installé par défaut. Il faut le choisir pendant la phase d'installation de netbeans EE ou ALL ⇒ **solution préférable**
- Installation de glassFish (windows)
 - Pour le chemin d'accès, il est conseillé un nom sans espace du style c:\glassfish ou bien c:\rep_parent_install_netbeans\glassFish
- Si pas installé avec Netbeans, télécharger, extraire dans un répertoire et ajouter un nouveau serveur à partir de netbeans en donnant le chemin du répertoire où les fichiers ont été extraits.
- Site téléchargement : <https://javaee.github.io/glassfish/>

Serveur GlassFish

- Dispose d'un outils d'administration
 - **Graphique** ⇒ `http://localhost:4848/` [server:port]
 - Ou les ports 4848,8080,8181
 - **Ligne de commande** ⇒ `Rep installation/bin/asadmin`
 - Pour : Démarrer, arrêter, Changer mot de passe, déployer des applications, voir liste des applications déployées,
- Le fichier de configuration pour changer les numéros de port en cas de conflit [on peut aussi créer une autre instance du serveur==> voir Ajouter un serveur GlassFish]
 - Rep d'installation `/glassfish-4.1.1/glassfish/domains/domain1/config/domain.xml`
 - `<network-listener protocol="http-listener-1" port="8080" name="http-listener-1" thread-pool="http-thread-pool" transport="tcp"></network-listener>`
 - on peut changer le numero de port

Ajouter un serveur GlassFish

- Ajouter/créer une autre instance de serveur GlassFish dans netbeans
 - clic droit sur serveurs ==> ajouter un serveur ==> choisir le type => renseignez les champs nom ==> choisir le répertoire d'installation du serveur (suppose que déjà installé) => plusieurs cas
 - Si il existe déjà d'autres instances avec leur nom de domaine, il faudra choisir un autre nom de **domain** dans la fenêtre suivante ==> **changer les numéro de port** (il y a souvent problème de port déjà utilisé) et éventuellement mettre un user+mot de passe (ou laisser vide). Après , il faut modifier les propriétés de l'application qu'on veut déployer, et dans la partie run, choisir le serveur qu'on veut.
- Menu outils => serveurs ==> ajouter un serveur (ou clic droit serveur)

Installation serveur Tomcat

- Inclus dans Netbeans EE et all mais pas installé par défaut. Il faut le choisir pendant la phase d'installation de netbeans EE ou ALL ⇒ **solution préférable**
- Installation de tomcat (windows)
 - Pour le chemin d'accès, il est conseillé un nom sans espace du style c:\tomcat ou bien c:\rep_parent_install_netbeans\tomcat
- Si pas installé avec Netbeans, télécharger, extraire dans un répertoire et ajouter un nouveau serveur à partir de netbeans en donnant le chemin du répertoire où les fichiers ont été extraits.
- Site téléchargement : <http://tomcat.apache.org/>

Serveur Apache Tomcat

- Dispose d'un outils d'administration
 - Graphique ⇒ <http://localhost:8084/> [serveur:port]
 - Une partie des outils est accessible [à l'utilisateur connecté] : etat serveur, liste des applications, ...
 - Pour savoir l'utilisateur connecté ⇒ clic sur le serveur==> propriété
 - On peut ajouter des utilisateurs et leur attribuer des rôles pour que certains outils soient accessible ou non.

Serveur Apache Tomcat

- Ajouter rôles et utilisateurs
 - Si on veut ajouter des rôles et des utilisateurs==> modifier le fichier repertoire installation du serveur tomcat / conf/ tomcat_users.xml
 - Définir les rôles parmi
 - manager-gui ==> autorise l'accès à l'interface Graphique HTML et au status des pages
 - manager-script ==> autorise l'accès à l'interface texte et au status des pages
 - manager-jmx ==> autorise l'accès au proxy JMX et au status des pages
 - manager-status ==> autorise l'accès au status des pages seulement
 - Définir un utilisateur (nom+mot de passe) et lui attribuer un rôle
 - Le fichier server.xml contient description paramètres serveur

Ajouter un serveur Apache Tomcat

- Ajouter/créer une autre instance de serveur Tomcat dans netbeans
 - clic droit sur serveurs ==> ajouter un serveur ==> choisir le type
⇒ renseignez les champs nom ==> choisir le répertoire
d'installation du serveur (suppose que déjà installé) ⇒
plusieurs cas
 - Si existe d'autres instances avec les mêmes noms de
répertoire **catalina_base** ⇒ cocher la case à cocher « use
private configuration (catalina_base) ==> créer un
répertoire et donner le chemin ==> renseignez **utilisateur**
et **mot de passe**
- Menu outils ⇒ serveurs ==> ajouter un serveur (ou clic droit
serveur)

Tester installation serveurs

- Pour tester si les serveurs sont bien configurés, on va créer une application web simple et la déployer.
- Nouveau projet ⇒ catégories Java web ⇒ web
 - Donner nom, ...
 - Pour le choix du serveur d'application, choisissez un , on va tester l'autre plus tard.
 - Ne pas choisir de framework
- Ouvrir le fichier **web pages / index.html** de l'application créée et modifier le titre et le corps en mettant « test » par exemple
- Onglets services ⇒ démarrer serveur choisi : clic droit ⇒ start
- Clic droit intérieur fichier index.html ⇒ exécuter le fichier
 - Vérifier résultat sur le navigateur

Connecter un serveur Mysql à netbeans

- On peut connecter un serveur Mysql à netbeans et créer des BD à travers l'interface Netbeans : s'assurer d'abord que le serveur Mysql a été démarré
 - Onglets services ⇒ clic droit Bases de données ⇒ connecter un serveur Mysql ⇒ renseigner les champs BD, user, password
 - Ou bien : section Driver ⇒ clic droit sur Mysql ⇒ connect using ⇒ renseigner les champs
 - Voir ⇒ <https://netbeans.org/kb/docs/ide/mysql.html>
- Il peut être nécessaire (windows?) de renseigner les champs de l'onglet avance (ou propriétés de l'admin) lorsqu'on clic sur connect using ou connecter un serveur Mysql
 - Path/URL to admin tool ⇒ copier le lien (naviguer) de l'outils d'administration comme PhpMyAdmin, MySQL Admin Tool, ou autre outils utilisé.
 - S'il y a des arguments à passer à l'outils , les saisir dans la zone arguments
 - Path to start command ⇒ chemin de la commande pour demarrer , dans le repertoire d'installation ⇒ rep instal/ bin/mysqld ou bien bin/mysql_safe pour les Unix ⇒ le nom peut varier (voir diapo suivant) ou ../bin/mysqld-nt.exe
 - Path to stop command ⇒ généralement rep installation Mysql /bin/mysqladmin ou ../bin/mysql.exe
 - Sur la ligne des arguments pour la commande ⇒ u-root stop
- Ou bien créer d'abord la BD et connecter le serveur à netbeans. On verra la liste des BD déjà créés , et on peut aussi créer de nouvelles à partir de l'interface de Netbeans

Connecter un serveur Mysql à netbeans ⇒ Ubuntu (config automatique)

Propriétés d'administration

URL de l'outil d'administration:

Arguments:

de la commande de démarrage:

Arguments:

de la commande d'arrêt:

Arguments:

Connecter un serveur Mysql à netbeans

- Si la connexion a été déjà réalisée, il suffit de démarrer le serveur
 - Services ⇒ BD ⇒ Clic droit sur le nom du Serveur ⇒ démarrer
- java DB / Derby est un SGBD intégré à netbeans

Schéma de la BD à créer

- Les tables
 - Etudiants (**num_ins**, nom, prenom);
 - Modules (**code_mod**, intitule);
 - Inscriptions (**num_ins**,**code_mod**);
- En gras les clés primaires.
- Type des champs : chaînes , sauf num_ins et code_mod qui sont des entiers .

Créer deux bases de données

- Une base `testbd` sous `mysql` (`userName` : `testuser`, mot de passe : `testpw`)
- Une base `testbd_derby` sous `Netbeans` avec `java DB / Derby`
 - Tutorial ⇒ voir diapo suivantes
 - <https://netbeans.org/kb/docs/ide/java-db.html>

Créer une BD derby avec netbeans

Créer la BD

- Ouvrir netbeans ⇒ onglet Services ⇒ Base de données/Databases ==> Java DB
 - Si l'onglet Services n'apparaît pas ⇒ cliquer sur fenêtre/window==> services
 - Il y a une BD par défaut app/sample
- Avant de créer une BD, vous pouvez préciser le répertoire dans lequel les BD créés seront sauvegardés
 - Clic droit sur Java DB ⇒ propriétés ⇒ renseignez le deuxième champ (emplacement de la BD)
- Pour créer la BD : clic droit sur Java DB ⇒ créer une BD/create DB
 - Dans le fenêtre de création de la bd ⇒ renseignez les informations
 - Nom BD ⇒ testbd_derby
 - Nom utilisateur ⇒ testuser (à retenir sera utilisé pour identifier la BD)
 - Mot de passe ⇒ testpw

Créer une BD derby avec netbeans

Créer la BD

- Après validation création BD
 - Sous services ==> Bases de données ==> java DB
 - Listes des BD derby
 - Si on utilise un autre serveur de BD (mysql) connecté à Netbeans =>
 - aura aussi sa ligne sous Bases de données et si le serveur est en mode connecté => liste des BD créés sous mysql
 - Si Mysql n'est pas connecté à netbeans => clic droit sur icône base de données => connecter un serveur Mysql => saisir mot de passe root
 - Sous services ==> Bases de données ==> pilotes/drivers
 - Pilotes des SGBD qu'on peut utiliser (certains intégrés, d'autres à télécharger)
 - Juste après pilotes => une ligne commençant par jdbc: ...
 - Une ligne pour chaque BD (créée ou par défaut) connectée ==> repérer les lignes des deux BD qu'on a créés : décrit une connexion
 - Pour connecter serveur de BD ou une BD ==> clic droit => connecter

Créer une BD derby avec netbeans

Créer les tables de la BD

- Après avoir connecté la BD testbd_derby ⇒ création tables
 - En affichant la ligne commençant par jdbc :... de cette BD,
 - Nom bd en gras ⇒ afficher ⇒ tables/tableaux , vues, procédures
 - Pour créer des tables pour la BD
 - Mode Graphique : clic droit sur tableaux ⇒ créer une table ⇒ nom Table + bouton ajouter colonne pour chaque colonne.
 - Mode sql : clic droit sur tableaux/tables ⇒ exécuter la commande ==> écrire commande sql création table et l'exécuter. Syntaxe sql présente quelques différences.

Créer une BD derby avec netbeans

Créer les tables de la BD

- Cf. liens documentation Java DB (2018)
 - <http://www.oracle.com/technetwork/java/javadb/documentation/index.html>
 - <https://docs.oracle.com/javadb/>
- **Mode sql** : exemple création table avec une clé primaire un champ

```
CREATE TABLE "enseignants_Test_creation_sql" (  
  "matricule" INTEGER not null primary key,  
  "prenom" VARCHAR(30),  
  "nom" VARCHAR(10),  
  "grade" VARCHAR(15),  
  "section" VARCHAR(10),  
  "adresse" VARCHAR(50) );
```

Créer une BD derby avec netbeans

Créer les tables de la BD

- **Mode sql** : exemple création table avec une clé primaire deux champs

```
CREATE TABLE "enseignants2_Test_creation_sql" (  
  "matricule" INTEGER not null ,  
  "prenom" VARCHAR(30) not null,  
  "nom" VARCHAR(10),  
  "grade" VARCHAR(15),  
  "section" VARCHAR(10),  
  "adresse" VARCHAR(50),  
  constraint pk_double primary key("matricule","prenom")  
);
```

Créer une BD derby avec netbeans

Créer les tables de la BD

- Pour créer une clé primaire constituée de plusieurs champs
 - Après avoir créé la table avec les champs, ajouter une contrainte

- Exemple avec la table Inscriptions

alter table Inscriptions

ADD CONSTRAINT pk PRIMARY KEY (num_ins,code_mod)

Créer une BD derby avec netbeans

Créer les tables de la BD

- Pour créer une contrainte de clé étrangère : une fois que la table est créée avec les champs ==> clic sur table ==> exécuter commande et écrire le code sql et l'exécuter, ensuite clic droit nom table => rafraîchir: exemple pour la table inscriptions
 - alter table Inscriptions
ADD CONSTRAINT fk1 FOREIGN KEY (num_ins) REFERENCES Etudiants(num_ins)
 - alter table Inscriptions
ADD CONSTRAINT fk2 FOREIGN KEY (code_mod) REFERENCES Modules(code_mod)

Créer une BD derby avec netbeans

Insérer des enregistrements

- En mode graphique : Clic droit sur le nom table ⇒ afficher les données ⇒
 - En bas fenêtre résultat ==> Onglet select * from... ⇒ bouton insérer enregistrement [ajouter une ligne pour ajouter un nouvel enregistrement].
 - Pour voir le code sql correspondant ⇒ clic bouton **montrer sql**
- **Mode sql** ==> clic droit nom table ==> exécuter commande ==> écrire le code sql (insert into.....)

insert into Modules(code_mod,intitule) values (11,'Maths');

Connecter une BD

- S'assurer que le serveur (Mysql ou Java DB) est démarré, sinon : afficher services ==> bases de données ==> clic droit sur le serveur ⇒ démarrer
- Une fois que le serveur est en marche, la liste des BD qu'il gère s'affiche. Pour connecter une des BD, clic droit sur le nom de la BD ==> se connecter
 - Si tout se passe bien, une nouvelle ligne de connexion (jdbc :.....) s'affiche avec le nom de la BD
- Pour se connecter à un serveur de BD ⇒ afficher le contenu de la ligne pilotes/drivers ==> choisir le pilote correspondant au type de la BD ==> clic droit ⇒ se connecter à l'aide ⇒ renseigner les champs

Tester les BD avec un programme Java

- **Exo** : Rappel JDBC
- Créer une application java standard pour tester la connexion
 - Pour chacune des deux BD créées, Afficher la liste des étudiants, ensuite insérer un étudiant (30, 'big','fap')
- ⇒

Tester les BD avec un programme Java

- Ajouter les pilotes (pour les deux BD) au projet [plusieurs manières]
 - Clic droit sur la partie bibliothèques [library/librairies en anglais] du projet ⇒ ajouter une bibliothèque :
 - Si Java DB Driver et Pilotes JDBC mysql ne sont pas dans la liste ⇒ bouton importer et ajouter ses pilotes
 - Si on a déjà téléchargés les fichiers du pilote, on peut les ajouter en cliquant [droit] sur la partie bibliothèques/librairies du projet ⇒ ajouter des fichiers jar ⇒ et ajouter les fichiers
 - Propriétés du projet ⇒ librairies ⇒ onglet compiler ⇒ boutons ajouter bibliothèque et ajouter fichiers jar

Tester les BD avec un programme Java

- Pour modifier le serveur sur lequel on a déployé une application
 - Onglets projet ⇒ Clic droit nom projet ==> propriétés ⇒ run ⇒
 - Choisir le serveur, navigateur par défaut pour les projets web,
- Pour créer une nouvelle instance du serveur [lorsqu'une instance s'exécute sur un port]
 - ==> clic sur serveurs ==> ajouter un serveur ⇒ choisir le type de serveur ⇒ choisir le répertoire où le serveur est installé ⇒ changer le numéro de port

Tester les BD avec un programme Java

Informations pour se connecter aux BD

- PILOTES

// pilote BD mysql si on veut se connecter à une BD mysql

String driver="com.mysql.jdbc.Driver";

// pilote BD Java DB (derby) si on veut se connecter à une BD Derby

//String driver="org.apache.derby.jdbc.ClientDriver";

- URL

//url BD mysql, port par default 3306

String url="jdbc:mysql://localhost:3306/testdb";/"jdbc:mysql://localhost/testdb"

//url BD DERBY, port par default 1527

//String url="jdbc:derby://localhost:1527/testbd_derby";

- ⇒

Tester les BD avec un programme Java

- Utilisateur et mot de passe

// nom user + password selon la BD à laquelle on veut se connecter

String login="testuser"; //root par défaut

String pswd="testpw"; //mabd ou chaine vide