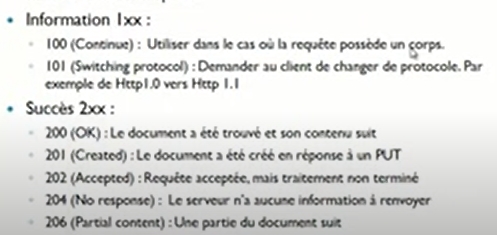
HTTP : HyperText Transfer Protocol : protocole que permet au client de faire des opérations sur des documents du serveur.

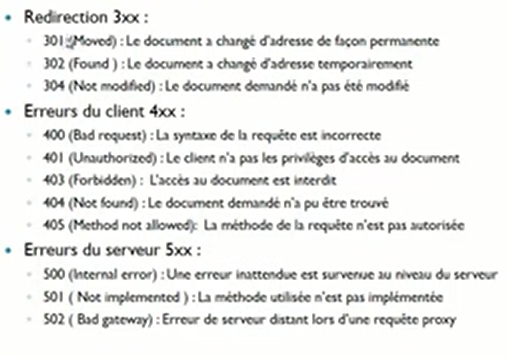
HEAD : méthode http pour consulter les métadonnées d’une ressource.

OPTIONS : permet au Client d’interroger le serveur pour fournir les options de communications a utiliser pour une ressource ou ensemble de ressources, par exemple les méthodes autorisées, entête autorisées, origines autorisées.

CODE STATUS :

1xx : Information | 2xx : Succès | 3xx : Redirection | 4xx : Erreur de Client | 5xx : Erreur de Serveur





**JEE**

1 – Client envoie une requête http

2 -- Serveur TOMCAT reçoit la requête et la revoie vers SERVLET

3 – SERVLET va envoyer la réponse via l’objet Response

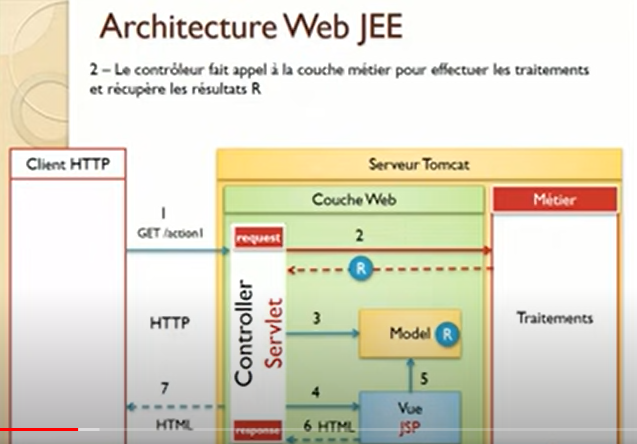
4 – pour chaque requête la servlet va effectuer un traitement, généralement **la servlet** fait appel à la **couche métier (couche service)**.

5 – la couche métier pour accéder à la base donnée va appeler la couche DAO

6 – quand la SERVLET va récupérer les données il va les stocker dans un **MODEL**

7 – la vue JSP va récupérer les données du MODEL et va générer un flux HTML

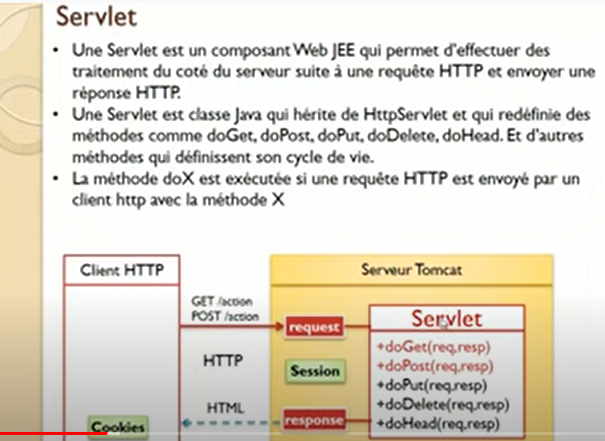
8 – SERVLET va renvoyer le flux générer vers le client.

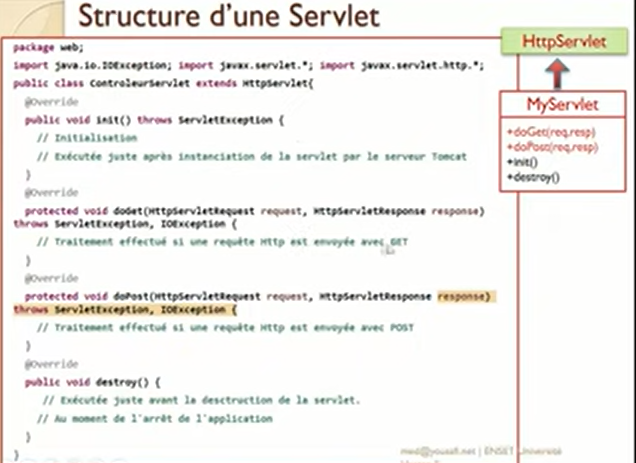


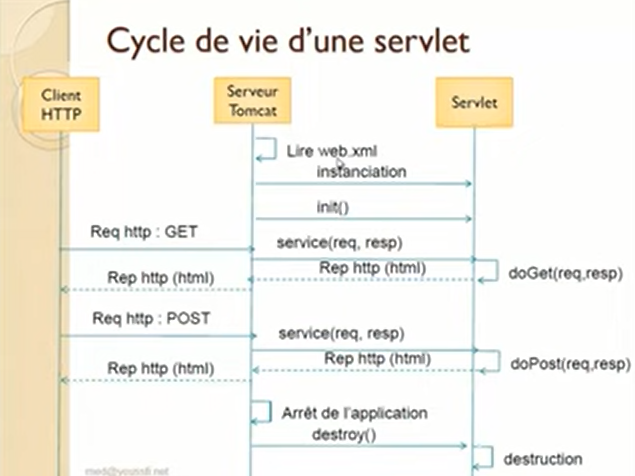
C’est le design pattern MVC

Web.xml

**SERVLET**







**Servlet** : class qui hérite du class **HttpServlet,** dans laquelle on redéfinie les méthodes init, doGet/doPost

**Web.xml** : descripteur de déploiement : le fichier dans lequel on déclare nos servlets

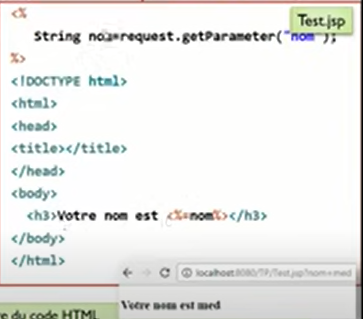


! On peut déployer une servlet directement avec l’annotation @webServlet, sans passer par web.xml

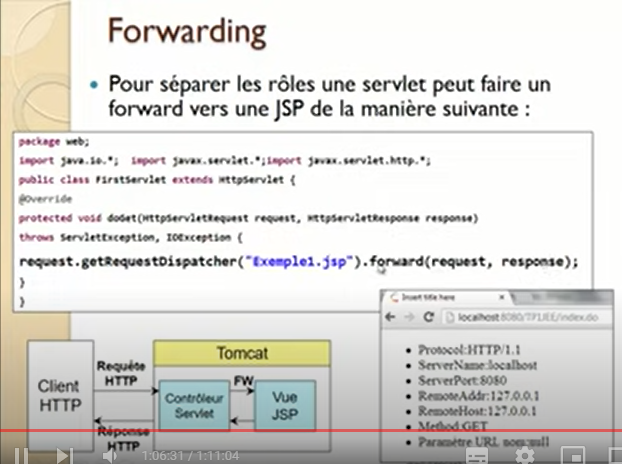
Imperatife

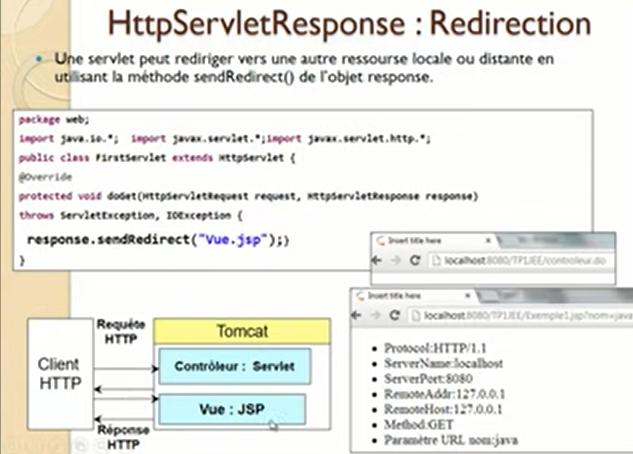


**JSP :** forme d’une page HTML dans laquelle on écrit du code java



**Passer d’une SERVLET a une page JSP :**

****

****

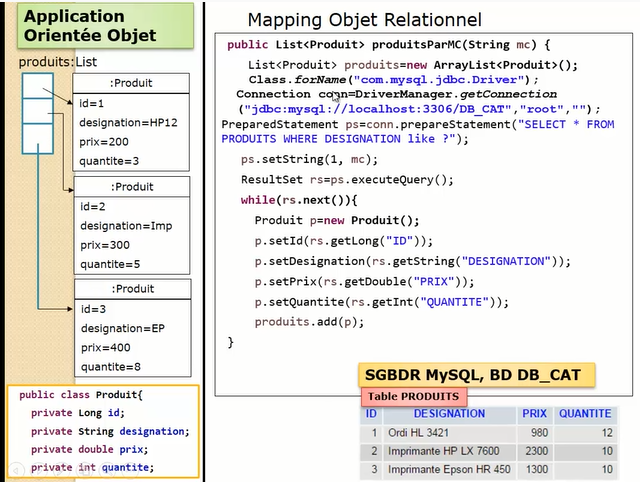
Dans JSP on trouve :

**Scriptlet** : code java dans la page JSP : <% int a = 0 ; %>

**Directive**: <%@page import= ‘web.MyClass’ %>

**Expression**: <%= a %>

**ORM**: mapping objet relationnel

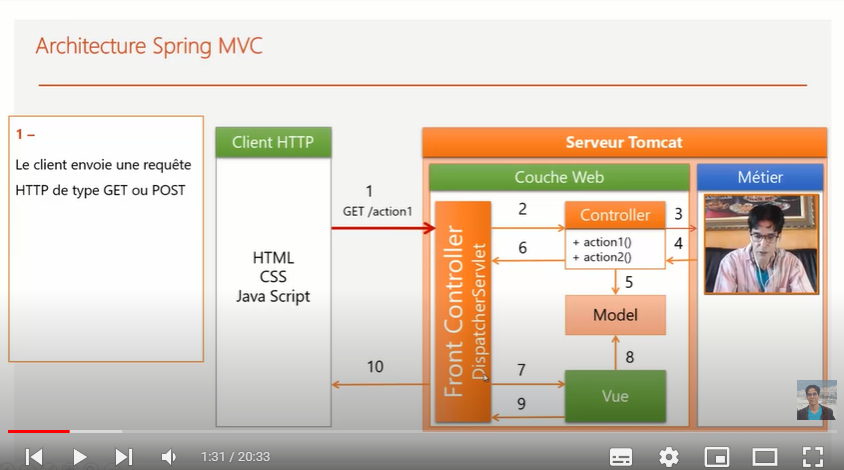


**SPRING MVC**

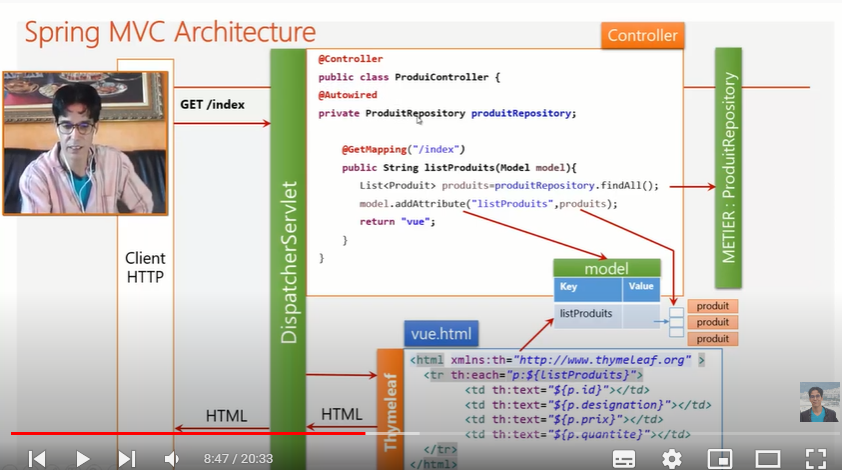
1-spring MVC vous donne une SERVLET prédéfinie s’appelle ‘Dispatcher Servlet’

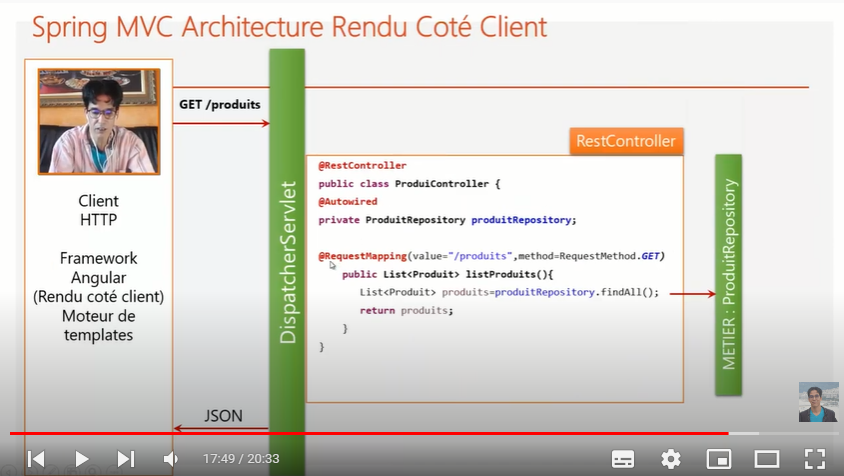
Il va intercepter (recevoir et traiter) tout les requêtes envoyer par le client.

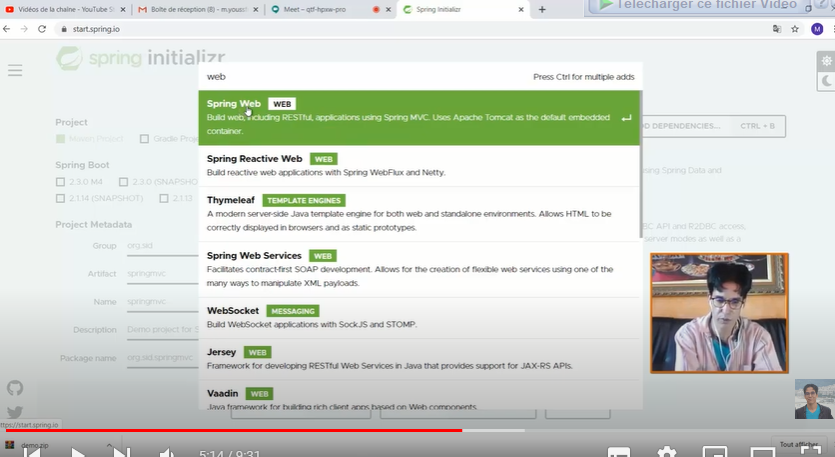
2- Spring MVC vous propose un model(objet de type Map) prédéfinie pour stocker les donnes.



**@Autowired** : pour injecter une implémentation de l’interface JpaRepository fournie par spring data





spring web : use TOMCAT server ( multithread avec I/O bloquant )

spring reactive web : use NETTY server ( single thread avec I/O non bloquant)