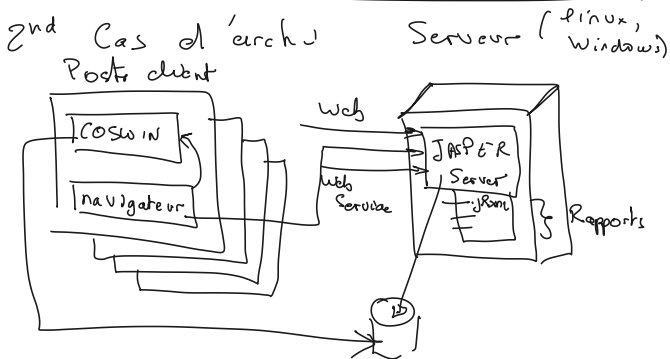
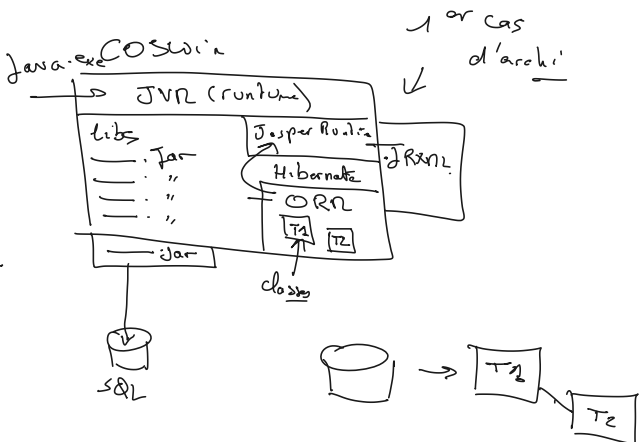
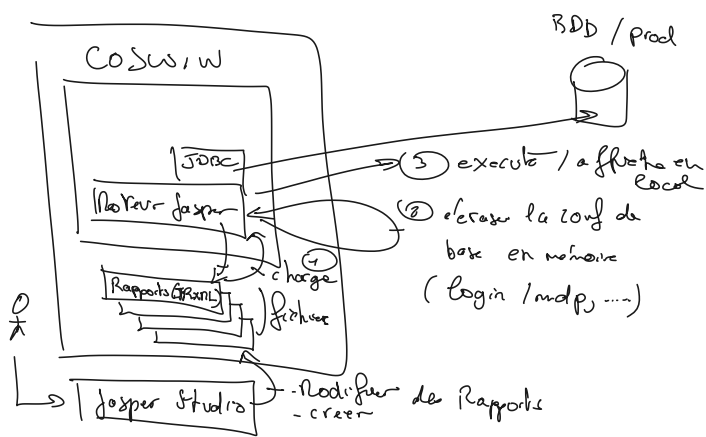
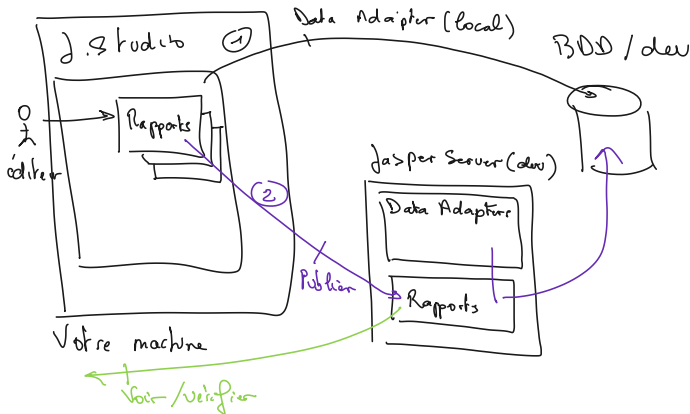
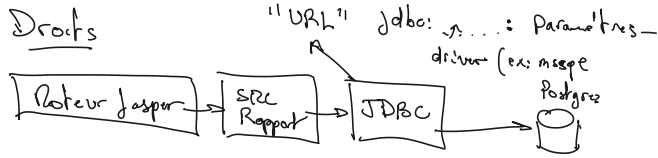


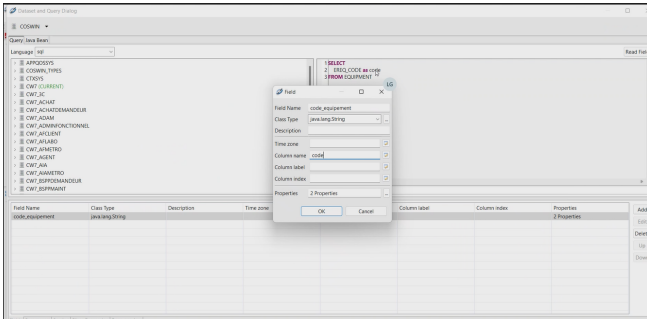
Jasper Reports



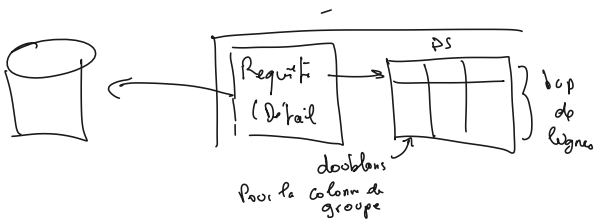
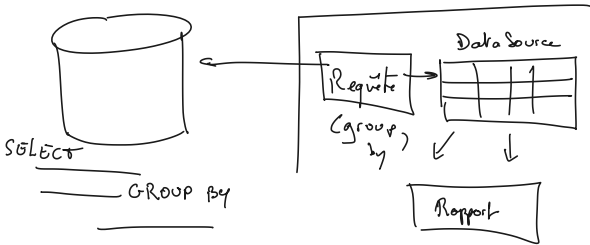


Droits





groupes:



Transformations de Types - Nom de Col

Requeste SQL →

c1	c2	c3	c4
int	varchar(50)	double	decimal(2,2)

- Type SQL

⇨ - Mapping

Types SQL -

int

(n)(var)char

boolean

(small)(date)(time)

decimal (nb chiffres entiers,
nb décimales)

double → virgule flottante,
arrondi

nb. $10^{(n)}$

$$\underbrace{1.3333} + \underbrace{1.66666\dots} = 2 \quad \text{Compta} \Rightarrow \boxed{1,99}$$

Java

col1	col2	col3	col4
java.lang. Integer (int)	String	Double	BigDecimal

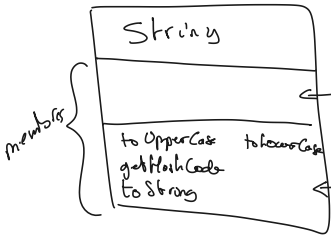
← Fields
&F

→ Comptabilité, pas
d'arrondi

Java:

Object → Type générique qui fait
rien mais accepte tout.

Type Java (sont des classes)



1 classes →

← données 0..n instances

1 Instance = 1 valeur

← fonctions

fonctions members = methods
même terme.

Surcharge d'opérateurs

$$1 + 1 = 2$$

$$"1" + "1" = \text{?}$$

Concatéateur

$$1 + 1 \rightarrow 2$$

(int) + (int)

$$(\text{Integer}) + (\text{Integer}) \rightarrow 2$$

$$\text{BigDecimal} + \text{BigDecimal} \rightarrow \text{Pos}$$

Possible

$$\text{Double} + \text{Double} \rightarrow \text{ok}$$

<https://docs.oracle.com/javase/8/docs/api/java/math/BigDecimal.html>

~~BigDecimal * int~~

⇒ (2)

↙ (1) ↓

int * int

BigDecimal.multiply(.ToBig...)

~~int~~

Integer → BigDecimal ?

BigDecimal . doubleValue() * Integer . doubleValue()

↳ petite perte de précision

BigDecimal . add
subtract
multiply
divide

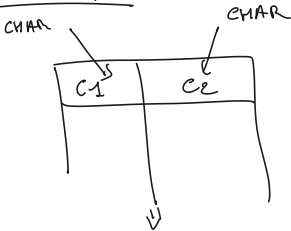
(new BigDecimal (int))

- . double Value()
- . floatValue()
- . intValue()

Autre solution à l'évaluation

→ créer une variable pour chaque champ qui convertisse en double

Autre risque:

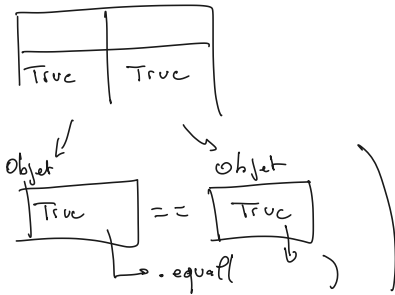


1 = 1 ? → True

Expression

→ $\$F\{c1\} == \$F\{c2\}$

OK → quand $c1 = c2$



• Equals
 Pour la
 Comparaison
 de
 chaîne

Opérateur "ternaire" pour les
 tests IF en expression

Valeur booléenne ? val si vraie : val si fausse

Développer vos propres méthodes / fonctions si non disponible sous Jasper

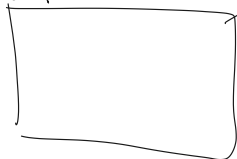
① Créer un jar - bibliothèque de méthodes

Projet Java-lib-jar (Eclipse, VS Code, ... ^{ant, maven})



②

Jasper



window / preference / Jasper /

build path / User
libraries / Add

↓
chemin vers la lib

③

Ajouter une scriptlet:

Nom de Scriptlet → Nom de classe -

ex: Scriptlet - 1

④

Utilisez les méthodes de la classe

dans des expressions

en faisant:

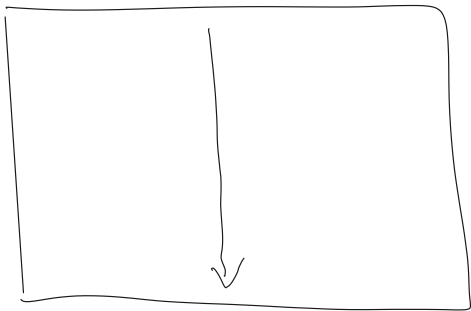
<Nom de la Scriptlet> . <nom de méthode>

exempli.

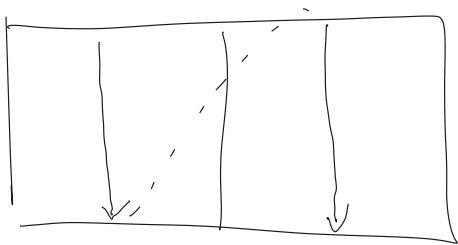
Scriptlet - 1 . colorToRGB(,)

retour de la fonction -

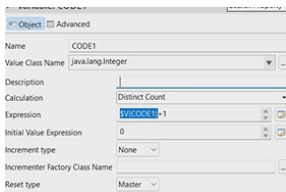
1 colonne



bas de
Page



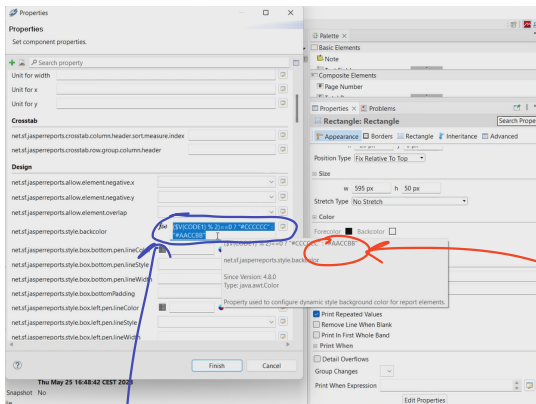
Création de la variable cumulée :



← la fonction de cumul
← d'expression de calcul

↑ le moment où la variable repasse à 0

Définition d'un propriété conditionnelle utilisant la variable :



retirez la code
"Transparent"
pour appliquer une
couleur de fond

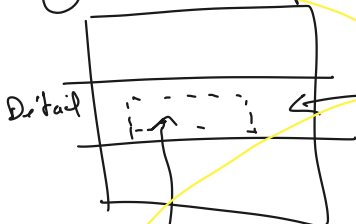
l'expression conditionnelle qui
change la couleur de fond
selon la variable, une ligne sur 2
grâce au module

Vendredi :

1. Paramètres
2. sous-rapports
3. Sous-rapports paramétriques
4. Graphiques
5. Tableaux croisés
6. Q&R et exemples de rapports

Paramètres & sous Rapports:

① créer le rapport principal + la Requête

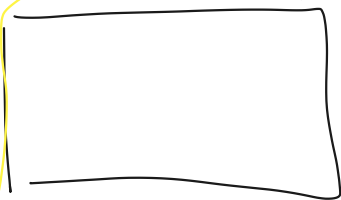


② Ajouter un subReport dans le détail

④ Éditer paramètres du Contrôle subReport, ajouter un

SubReport:

③ créer un Paramètre (de bon type)



Parameter
match
PARAM →
valeur du
Rapport
Principal
Tester

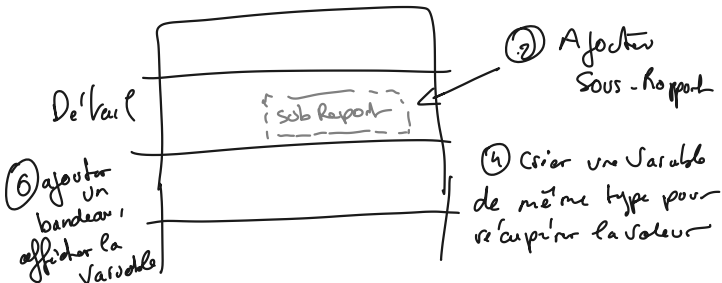
④ Définir une Requête dans le subReport avec une clause WHERE

where = \$P { Param-du-subReport }

⑤ Compiler le subReport (si le subReport est un `-.jasper`)
à chaque fois

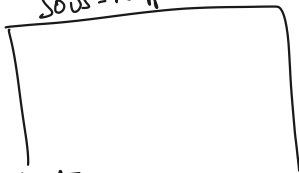
Récupérer une variable de sous-Rapport

(1) Créer Rapport + Requêtes



(4) Créer une variable de même type pour récupérer la valeur

(3) Créer une variable ayant une valeur cohérente en bas de sous Rapport



(5) Propriétés du Sous-Rapport /

SubReport /

Edit Return Value,

Add / ajouter,

From Variable: non exact de la variable dans le sous Rapport

To Variable: Variable de récupération dans le principal