

### Commencer à travailler sur un état :

Téléchargement d'un rapport et du XML de données :

- Rentrer en modification du paramètre état dans « Gérer les paramètres de publication »
- Cliquer sur l'icône en forme de zip pour récupérer les fichiers de l'état
- Mettre le format de sortie en XML, et valider
- Aller dans l'activité de publication, et publier un élément, le XML se télécharge
- Retourner dans le paramètre de publication, remettre le format en PDF et valider
- Dézipper le zip

Si c'est la première modification du rapport :

- Renommer le rapport dossier du rapport dézippé (exemple N9\_commandeVente en N9\_commandeVente\_NOMCLIENT)
- Renommer le rapport principal, jrxml et jasper (exemple N9\_commandeVente.jrxml en N9\_commandeVente\_NOMCLIENT.jrxml)

Paramétrage de IReport :

- Aller dans le menu « Options / chemin de classe »
- Supprimer le chemin de classe existant si il y en a un
- Renseigner le chemin de classe (exemple C:\Publis\N9\_commandeVente\_NOMCLIENT)
- Aller dans le menu « données / Connexions / source de données », cliquer sur « SYLOB »
- Renseigner le chemin du XML de données (exemple C:\Publis\CommandeVente.XML)
- Redémarrer IReport

### Syntaxes :

- $\$F\{\text{nomChamp}\}$  -> référence au champ nommé nomChamp
- $\$R\{\text{clemessage}\}$  -> référence à une clé dans le fichier specif-etat.properties
- $\$P\{\text{nomParametre}\}$  -> référence au paramètre nommé nomParametre
- $\$V\{\text{nomVariable}\}$  -> référence à la variable nommée nomVariable

### Fonctions utiles :

- IReportUtil.codeBarreC39( $\$F\{\text{valeurCodeBarreAAfficher}\}$ ) : Permet d'afficher un code barre
- IReportUtil.codeBarreEAN128( $\$F\{\text{valeurCodeBarreAAfficher}\}$ , 90) : Permet d'afficher un code barre avec une rotation de 90°
- sylob:logoSociete() : Affiche le logo de la société
- sylob:commentaire() : affiche le commentaire de l'élément
- ( condition ? expressionSiVrai : expressionSiFaux ) : Structure conditionnelle : Permet de conditionner un affichage
- && : Permet de faire un « ET » entre deux conditions
- || : Permet de faire un « OU » entre deux conditions

### Fonctions mathématiques :

- IReportUtil.add( $\$F\{\text{nomInteger}\}$ , 5) : Fait l'opération nomInteger + 5
- IReportUtil.subtract( $\$F\{\text{Integer1}\}$ ,  $\$F\{\text{Integer2}\}$ ) : Fait l'opération Integer1 – Integer2
- IReportUtil.multiply( $\$F\{\text{nomInteger}\}$ , 5) : Fait l'opération nomInteger X 5
- IReportUtil.divide( $\$F\{\text{nomInteger}\}$ , 5) : Fait l'opération nomInteger / 5

# MEMO IREPORT

### Terminer le travail sur un état :

Création du zip :

- Aller dans le répertoire racine du rapport (exemple C:\Publis\N9\_commandeVente\_NOMCLIENT)
- Zipper l'ensemble des fichiers (exemple N9\_commandeVente\_NOMCLIENT.zip), il faut que le zip ait forcément le même nom que le répertoire dans lequel on est.

Chargement dans Sylob

- Rentrer en modification du paramètre état dans « Gérer les paramètres de publication »
- Si c'est le premier chargement créer un nouveau groupe de feuille de style, que vous appellerez par exemple NOMCLIENT
- Sinon, entrer en modification du paramètre NOMCLIENT
- Entrer en modification de l'unique feuille de style
- Sélectionner le zip, et valider 2 fois
- Si c'est le premier chargement, sélectionner dans la liste des groupes de feuilles de styles le groupe NOMCLIENT
- Valider

### Formatages :

- IReportUtil.formatNumber( $\$F\{\text{Prix}\}$ ,  $\$F\{\text{nbDecimale}\}$ ,  $\$P\{\text{REPORT_LOCALE}\}$ ) : Formate un numérique avec une valeur variable de décimales
- IReportUtil.formatNumber( $\$F\{\text{Quantite}\}$ , 2,  $\$P\{\text{REPORT_LOCALE}\}$ ) : Formate un numérique avec 2 décimales
- IReportUtil.formatDate( $\$F\{\text{maDate}\}$ ,  $\$R\{\text{pattern.date8}\}$ ) : formate une date au format JJ/MM/AAAA

### Fonctions de tests :

- IReportUtil.isEmpty( $\$F\{\text{nomChamp}\}$ ) : Teste qu'un champ est renseigné
- IReportUtil.isEmpty( $\$F\{\text{nomChamp}\}$ ) : Teste qu'un champ n'est pas renseigné
- IReportUtil.equals( $\$F\{\text{nomChamp}\}$ , "MaChaine") : Teste que deux chaines sont égales
- IReportUtil.notEquals( $\$F\{\text{nomChamp}\}$ , "MaChaine") : Teste que deux chaines sont différentes
- IReportUtil.endsWith( $\$F\{\text{nomChamp}\}$ , "MaChaine") : Teste qu'une chaine se termine par « MaChaine »
- IReportUtil.startsWith( $\$F\{\text{nomChamp}\}$ , "MaChaine") : Teste qu'une chaine commence par « MaChaine »
- IReportUtil.length( $\$F\{\text{nomChamp}\}$ ) : renvoie le nombre de caractères d'une chaine
- IReportUtil.substring( $\$F\{\text{nomChamp}\}$ , 1, 6) : string coupe la chaine du caractère 2 au 6
- IReportUtil.contains( $\$F\{\text{nomChamp}\}$ , "MaChaine") : Teste si un champ contient « MaChaine »
- IReportUtil.notContains( $\$F\{\text{nomChamp}\}$ , "MaChaine") : Teste qu'un champ ne contient pas « MaChaine »