Labo 2

Sérialisation XML avec Java jdom

Sérialisation Json avec Java &Google/Gson

Vincent Guidoux et David Jaquet

Avril 2018

Table des matières

[1 Petite introduction 2](#_Toc511915006)

[2 Les modifications éventuelles apportées à vos structures XML et JSON élaborées dans le 1er laboratoire. Décrire alors l’objectif de ces modifications. 2](#_Toc511915007)

[3 Présenter le code des contrôleurs ControleurXMLCreation et ControleurMedia 2](#_Toc511915008)

[4 Commentez votre solution (méthodes rajoutées, complétées, leur objectif, points remarquables). 2](#_Toc511915009)

[5 Présenter un extrait représentatif des deux fichiers que vous aurez générés. 2](#_Toc511915010)

[6 Bilan - Conclusion 2](#_Toc511915011)

# Petite introduction

Nous avons pour objectif de générer des d

# Les modifications éventuelles apportées à vos structures XML et JSON élaborées dans le 1er laboratoire. Décrire alors l’objectif de ces modifications.

Nous avions oublié de mettre l’heure et les minutes pour les projections. Et nous avions aussi une autre idée que celle de la base de donnée concernant les langafges

# Présenter le code des contrôleurs ControleurXMLCreation et ControleurMedia

## ControleurXMLCreation

**public void** createXML() {  
 **new** Thread() {  
 **public void** run() {  
 *mainGUI*.setAcknoledgeMessage(**"Creation XML... WAIT"**);  
 **long** currentTime = System.*currentTimeMillis*();  
 **try** {  
 **globalData** = **ormAccess**.GET\_GLOBAL\_DATA();  
 **try** {  
 Document doc = **new** Document();  
 XMLOutputter outp = **new** XMLOutputter(Format.*getPrettyFormat*());  
  
 Element projections = **new** Element(**"projections"**);  
  
 **for** (Projection p : **globalData**.getProjections()) {  
 Element projection = **new** Element(**"projection"**);  
  
 Element date = **new** Element(**"date"**);  
  
 Calendar dateheure = p.getDateHeure();  
 Element jour = **new** Element(**"jour"**);  
 jour.addContent(Integer.*toString*(dateheure.get(Calendar.***DAY\_OF\_MONTH***)));  
 Element mois = **new** Element(**"mois"**);  
 mois.addContent(Integer.*toString*(dateheure.get(Calendar.***MONTH***)));  
 Element annee = **new** Element(**"annee"**);  
 annee.addContent(Integer.*toString*(dateheure.get(Calendar.***YEAR***)));  
 Element heure = **new** Element(**"heure"**);  
 heure.addContent(Integer.*toString*(dateheure.get(Calendar.***HOUR\_OF\_DAY***)));  
 Element minute = **new** Element(**"minute"**);  
 minute.addContent(Integer.*toString*(dateheure.get(Calendar.***MINUTE***)));  
  
 date.addContent(jour);  
 date.addContent(mois);  
 date.addContent(annee);  
 date.addContent(heure);  
 date.addContent(minute);  
  
  
 Element salle = **new** Element(**"salle"**);  
 salle.addContent(p.getSalle().getNo());  
  
 Element film = **new** Element(**"film"**);  
  
 Film f = p.getFilm();  
  
 Element titre = **new** Element(**"titre"**);  
 titre.addContent(f.getTitre());  
 Element synopsis = **new** Element(**"synopsis"**);  
 synopsis.addContent(f.getSynopsis());  
 Element duree = **new** Element(**"duree"**);  
 duree.addContent(Integer.*toString*(f.getDuree()));  
  
 Element critiques = **new** Element(**"critiques"**);  
 **for** (Critique c : f.getCritiques()) {  
 Element critique = **new** Element(**"critique"**);  
  
 Element texte = **new** Element(**"texte"**);  
 texte.addContent(c.getTexte());  
 Element note = **new** Element(**"note"**);  
 note.addContent(Integer.*toString*(c.getNote()));  
  
 critique.addContent(texte);  
 critique.addContent(note);  
  
 critiques.addContent(critique);  
 }  
  
 Element genres = **new** Element(**"genres"**);  
 **for** (Genre g : f.getGenres()) {  
 Element genre = **new** Element(**"genre"**);  
 genre.addContent(g.getLabel());  
  
 genres.addContent(genre);  
 }  
  
 Element mots\_cles = **new** Element(**"mots\_cles"**);  
 **for** (Motcle mc : f.getMotcles()) {  
 Element mot\_cle = **new** Element(**"mot\_cle"**);  
 mot\_cle.addContent(mc.getLabel());  
  
 mots\_cles.addContent(mot\_cle);  
 }  
  
 Element langages = **new** Element(**"langages"**);  
 **for** (Langage l : f.getLangages()) {  
 Element langage = **new** Element(**"langage"**);  
 langage.addContent(l.getLabel());  
  
 langages.addContent(langage);  
 }  
  
 Element photo = **new** Element(**"photo"**);  
 photo.setAttribute(**"url"**, **""** + f.getPhoto());  
  
  
 Set<Acteur> acteurSet = **new** HashSet<>();  
  
 **for** (RoleActeur ra : f.getRoles()) {  
 acteurSet.add(ra.getActeur());  
 }  
  
  
 Element acteurs = **new** Element(**"acteurs"**);  
 **for** (Acteur a : acteurSet) {  
 Element acteur = **new** Element(**"acteur"**);  
  
  
 acteur.setAttribute(**"sexe"**, a.getSexe().name());  
 Element nom = **new** Element(**"nom"**);  
 nom.addContent(a.getNom());  
 Element nom\_naissance = **new** Element(**"nom\_naissance"**);  
 nom\_naissance.addContent(a.getNomNaissance());  
 Element biographie = **new** Element(**"biographie"**);  
 biographie.addContent(a.getBiographie());  
  
 Element roles = **new** Element(**"roles"**);  
  
 **for** (RoleActeur ra : f.getRoles()) {  
 **if** (ra.getActeur().getId() == a.getId()) {  
 Element role = **new** Element(**"role"**);  
  
 Element personnage = **new** Element(**"nom"**);  
 personnage.addContent(ra.getPersonnage());  
 Element place = **new** Element(**"place"**);  
 place.addContent(Long.*toString*(ra.getPlace()));  
  
 role.addContent(personnage);  
 role.addContent(place);  
  
 roles.addContent(role);  
 }  
 }  
  
  
 Element date\_naissance = **new** Element(**"date\_naissance"**);  
  
 Calendar dateHeure1 = a.getDateNaissance();  
 Element jour1 = **new** Element(**"jour"**);  
 Element mois1 = **new** Element(**"mois"**);  
 Element annee1 = **new** Element(**"annee"**);  
 **if** (dateHeure1 != **null**) {  
 jour1.addContent(Integer.*toString*(dateHeure1.get(Calendar.***DAY\_OF\_MONTH***)));  
 mois1.addContent(Integer.*toString*(dateHeure1.get(Calendar.***MONTH***)));  
 annee1.addContent(Integer.*toString*(dateHeure1.get(Calendar.***YEAR***)));  
  
 }  
 date\_naissance.addContent(jour1);  
 date\_naissance.addContent(mois1);  
 date\_naissance.addContent(annee1);  
  
  
 Element date\_deces = **new** Element(**"date\_deces"**);  
  
 Calendar dateHeure2 = a.getDateDeces();  
 Element jour2 = **new** Element(**"jour"**);  
 Element mois2 = **new** Element(**"mois"**);  
 Element annee2 = **new** Element(**"annee"**);  
 **if** (dateHeure2 != **null**) {  
 jour2.addContent(**""** + Integer.*toString*(dateHeure2.get(Calendar.***DAY\_OF\_MONTH***)));  
 mois2.addContent(**""** + Integer.*toString*(dateHeure2.get(Calendar.***MONTH***)));  
 annee2.addContent(**""** + Integer.*toString*(dateHeure2.get(Calendar.***YEAR***)));  
 }  
 date\_deces.addContent(jour2);  
 date\_deces.addContent(mois2);  
 date\_deces.addContent(annee2);  
  
  
 acteur.addContent(nom);  
 acteur.addContent(nom\_naissance);  
 acteur.addContent(biographie);  
 acteur.addContent(roles);  
 acteur.addContent(date\_naissance);  
 acteur.addContent(date\_deces);  
  
  
 acteurs.addContent(acteur);  
 }  
  
 film.addContent(titre);  
 film.addContent(synopsis);  
 film.addContent(duree);  
 film.addContent(critiques);  
 film.addContent(genres);  
 film.addContent(mots\_cles);  
 film.addContent(langages);  
 film.addContent(photo);  
 film.addContent(acteurs);  
  
  
 projection.addContent(date);  
 projection.addContent(salle);  
 projection.addContent(film);  
  
 projections.addContent(projection);  
 }  
  
 doc.addContent(projections);  
  
 DocType docType = **new** DocType(**"projections"**, **"Plex.dtd"**);  
 doc.setDocType(docType);  
 outp.output(doc, **new** FileOutputStream(**"plex.xml"**));  
  
 *mainGUI*.setAcknoledgeMessage(**"Creation XML... FINISHED"**);  
  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 }  
 } **catch** (Exception e) {  
 *mainGUI*.setErrorMessage(**"Construction XML impossible"**, e.toString());  
 }  
 }  
 }.start();  
}

## ControleurMedia

**public void** sendJSONToMedia(){  
 **new** Thread(){  
 **public void** run(){  
 *mainGUI*.setAcknoledgeMessage(**"Envoi JSON ... WAIT"**);  
 **try** {  
 **globalData** = **ormAccess**.GET\_GLOBAL\_DATA();  
  
 Gson moteurJson = **new** GsonBuilder().setPrettyPrinting().create();  
  
 BufferedWriter out = **new** BufferedWriter(**new** OutputStreamWriter(**new** FileOutputStream(**"plex.json"**), **"UTF-8"**));  
 JsonObject data = **new** JsonObject();  
 JsonArray projectionsList = **new** JsonArray();  
  
 **for** (Projection proj : **globalData**.getProjections()){  
 JsonObject projection = **new** JsonObject();  
  
 *// Date* JsonObject date = **new** JsonObject();  
 Calendar projectionDate = proj.getDateHeure();  
 date.add(**"jour"**, **new** JsonPrimitive(projectionDate.get(Calendar.***DAY\_OF\_MONTH***)));  
 date.add(**"mois"**, **new** JsonPrimitive(projectionDate.get(Calendar.***MONTH***)));  
 date.add(**"annee"**, **new** JsonPrimitive(projectionDate.get(Calendar.***YEAR***)));  
 date.add(**"heure"**, **new** JsonPrimitive(projectionDate.get(Calendar.***HOUR\_OF\_DAY***)));  
 date.add(**"minute"**, **new** JsonPrimitive(projectionDate.get(Calendar.***MINUTE***)));  
 projection.add(**"date"**, date);  
  
 *// Film* JsonObject film = **new** JsonObject();  
 film.add(**"titre"**, **new** JsonPrimitive(proj.getFilm().getTitre()));  
  
 *// Acteur* JsonArray actorList = **new** JsonArray();  
 **for**(RoleActeur role : proj.getFilm().getRoles()){  
 **if**(role.getPlace() == 1 || role.getPlace() == 2){  
 JsonObject actor = **new** JsonObject();  
 **if**(role.getActeur().getNomNaissance() != **null**)  
 actor.add(**"nom\_naissance"**, **new** JsonPrimitive(role.getActeur().getNomNaissance()));  
 **else** actor.add(**"nom"**, **new** JsonPrimitive(role.getActeur().getNom()));  
  
 actorList.add(actor);  
 }  
 }  
  
 film.add(**"acteurs"**, actorList);  
 projection.add(**"film"**, film);  
  
 projectionsList.add(projection);  
 }  
  
 data.add(**"projections"**, projectionsList);  
  
 out.write(moteurJson.toJson(data));  
 out.close();  
  
 *mainGUI*.setAcknoledgeMessage(**"Envoi JSON ... FINISHED"**);  
 }  
 **catch** (Exception e){  
 *mainGUI*.setErrorMessage(**"Construction XML impossible"**, e.toString());  
 }  
 }  
 }.start();  
}

# Commentez votre solution (méthodes rajoutées, complétées, leur objectif, points remarquables).

# Présenter un extrait représentatif des deux fichiers que vous aurez générés.

# Bilan - Conclusion