

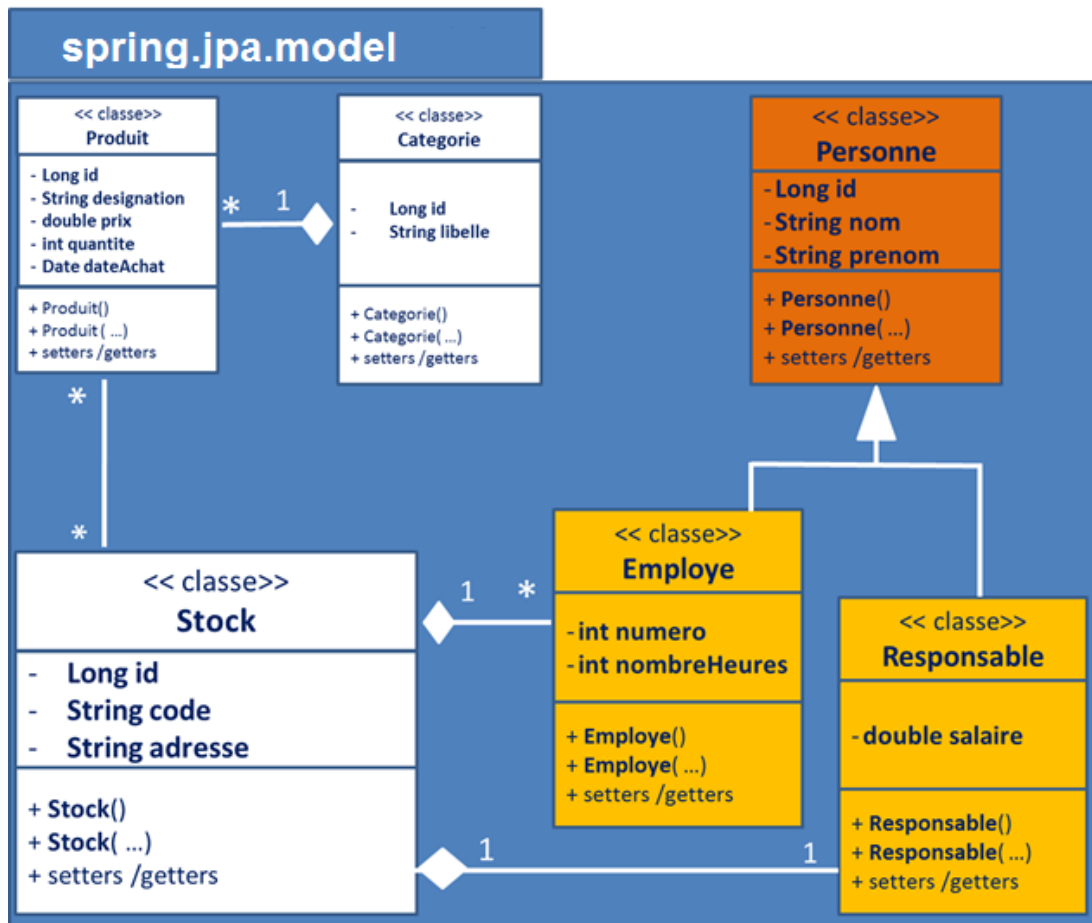
# Atelier Spring -03- JPA –Partie10

## Objectifs

- **JPA et Spring Data**
  - Mettre en œuvre d'une relation d'héritage JPA
  - Stratégie : **@Inheritance : Single Table**
  - Réaliser les requêtes polymorphes

Voici un modèle conceptuel qui modélise une relation d'héritage :

- Une sous-classe « **Responsable** » caractérisée par un **salaire**
- Une sous-classe « **Employe** » caractérisée par deux attributs (**numero** et **nombreHeures**)
- Une classe mère « **Personne** » qui présente les attributs communs (**id**, **nom** et **prenom**)



## • La stratégie « @Inheritance : Single Table »

1. Prendre une copie du projet «jpa-spring-data-spring-boot-2-relations-3-heritage1» et le nommer «jpa-spring-data-spring-boot-2-relations-3-heritage2»
2. Modifier la classe «**Personne**» (classe mère comme entité) en déclarant les annotations suivantes avant la définition de la classe :

```
.....  
//pour déclarer une classe mère (entité)  
@Entity  
@Inheritance(strategy = InheritanceType.SINGLE_TABLE)  
@DiscriminatorColumn(name = "TYPE_PERSONNE")  
@DiscriminatorValue("Personne")  
public class Personne {  
    // code de la classe ...  
}
```

3. Modifier la classe «**Employe**» (sous-classe) en déclarant les annotations suivantes avant la définition de la classe :

```
@Entity  
@DiscriminatorValue("employe")  
public class Employe extends Personne {  
    // code la classe ...  
}
```



4. Modifier le code de l'entité «**Responsable**» (sous-classe) en déclarant les annotations suivantes avant la définition de la classe :

```
@Entity  
@DiscriminatorValue("responsable")  
public class Responsable extends Personne {  
    // code la classe ...  
}
```

5. Lancer l'exécution du projet en utilisant une autre base de données « **spring-jpa-2** » (créer cette base de données et la référencer dans le fichier « **application.properties** ») et remarquer :
  - ✓ La génération d'une seule table «**Personne**» pour toute la hiérarchie :

Table
<input type="checkbox"/> categorie
<input type="checkbox"/> hibernate_sequence
<input type="checkbox"/> <b>personne</b>
<input type="checkbox"/> produit
<input type="checkbox"/> produit_stocks
<input type="checkbox"/> stock

- ✓ Cette table « **Personne** » englobe tous la attributs de tous les sous-classes :

	#	Nom	Type
<input type="checkbox"/>	1	<b>type_personne</b>	varchar(31)
<input type="checkbox"/>	2	<b>id</b> 	bigint(20)
<input type="checkbox"/>	3	<b>nom</b>	varchar(255)
<input type="checkbox"/>	4	<b>prenom</b>	varchar(255)
<input type="checkbox"/>	5	<b>nombre_heures</b>	int(11)
<input type="checkbox"/>	6	<b>numero</b>	int(11)
<input type="checkbox"/>	7	<b>salaire</b>	double
<input type="checkbox"/>	8	<b>stock_id</b> 	bigint(20)

- ✓ La colonne « **type\_personne** » permet de distinguer les données de chaque entité.

<b>type_personne</b>	<b>id</b>	<b>nom</b>	<b>prenom</b>	<b>nombre_heures</b>	<b>numero</b>	<b>salaire</b>	<b>stock_id</b>
responsable	2	Ben Saleh	Ali	NULL	NULL	0	NULL
responsable	4	Ben Ahmed	Omar	NULL	NULL	0	NULL
responsable	6	Salleme	Samira	NULL	NULL	0	NULL
responsable	7	Zouari	Zied	NULL	NULL	0	NULL
employe	9	Triki	Samir	0	1	NULL	1