## Compile Time Checking

Distinguer deux objets génériques par leur paramètres génériques n’est pas quelque chose que l’on peut faire en java

On ne peut pas distinguer le type de parametre dentree generique dans une liste d objet generique au moment du run time

Cf

<https://stackoverflow.com/questions/12320429/java-how-to-check-the-type-of-an-arraylist-as-a-whole>

<https://stackoverflow.com/questions/16745355/check-which-type-of-object-list-contains/16745403>

Les types Génériques ne fournissent que des vérifications au moment de la compilation

Au moment de l’exécution, ces vérifications disparaissent.

Cf « Type Erasure »

Generics only provide **compile-time checks**. At runtime, they are completely gone. This is known as type erasure. So at runtime, your code looks like this:

List rhsList1 = new ArrayList();

rhsList1.add(segmentDetailInfo);

List rhsList2 = new ArrayList();

rhsList2.add(segReqInfoBean);

doProspecListCompareCheck(rhsList1);

doProspecListCompareCheck(rhsList2);

}

private static void doProspecListCompareCheck(Object rhsList) {

if (rhsList instanceof List) //wrong Check

//DoTHIS

else if(rhsList instanceof List) //wrong Check

//Do THIS

}

Distinguishing two generic objects by their generic parameter is simply not something you can do in Java.

for (Object aList : list) {

Class cls = aList.getClass();

System.out.println("The type of the object is: " + cls.getName());

}

## Run time type Checking

Une solution est trouvée pour faire des vérifications au moment du run time

<https://stackoverflow.com/questions/25507741/run-time-type-checking-and-casting-in-java>

The warning will exist as part of compile type checking against casting Objects of super type to an Object of specific type.It is the way by which compiler is telling you.

***"Hey,I cant assure whether the Object you are casting is an Object of ArrayList<LinkedHashMap<String, Double>> and when you treat this Object as this type,be prepared for RuntimeExceptions.So from now on ,you are on your own."***

If you only want to rid of this exception and you are prepared for any RuntimeException which may occur,go ahead and mark it with **@SuppressedWarning ("unchecked")** .But I suggest you do some defensive mechansim and edit you code thus

ArrayList<LinkedHashMap<String, Double>> targetDetailContainer = null;

Map<String, Object> confData = getConfData();

if (confData.containsKey("Target-Details")) {

Object value = confData.get("Target-Details");

if (value instanceof ArrayList<?>) {

ArrayList temp = (ArrayList) value;

for (Object vals : temp) {

boolean flag = false;

if (vals instanceof LinkedHashMap<?, ?>) {

LinkedHashMap<?, ?> map = (LinkedHashMap) vals;

for (Map.Entry entry : map.entrySet()) {

if (entry.getKey() instanceof String

&& entry.getValue() instanceof Double) {

flag = true;

}

if (!flag) {

throw new RuntimeException(

"Objects are of different types");

}

}

}

targetDetailContainer = (ArrayList<LinkedHashMap<String, Double>>) confData

.get((Object) "Target-Details");

}

}

}

<http://resources.esri.com/help/9.3/arcgisengine/java/doc/7c061b6b-7317-4d53-aa15-a9c0a43f278b.htm>

List< ? extends TreeNode<T>

Appeler une classe generique a partir d’une classe abstraite

<https://stackoverflow.com/questions/18257159/how-to-define-a-generic-list-in-an-abstract-super-class>

ParametreEntreeAbstract compare elt=1

ParametreEntreeAbstract compare elt=2

ParametreEntreeAbstract compare elt=3

ParametreEntreeAbstract compare elt=4

ParametreEntreeAbstract compare elt=5

[ParametreEntreeAbstract] 1ère boucle val=1

[ParametreEntreeAbstract] l'élément=1 est de type Integer

[ParametreEntreeAbstract] 2ème boucle val2=1

[ParametreEntreeAbstract] 2ème boucle val2=2

[ParametreEntreeAbstract] 2ème boucle val2=3

[ParametreEntreeAbstract] 2ème boucle val2=4

[ParametreEntreeAbstract] 2ème boucle val2=5

[ParametreEntreeAbstract] Création d'un paramètre d'entrée de type Integer

Exception in thread "main" java.lang.ClassCastException: class fr.adneom.question1.ParametreEntreeEntier cannot be cast to class java.util.List (fr.adneom.question1.ParametreEntreeEntier is in unnamed module of loader 'app'; java.util.List is in module java.base of loader 'bootstrap')

at fr.adneom.question1.Handler.creerStrategiePartitionnement(Handler.java:129)

at fr.adneom.question1.Handler.main(Handler.java:169)

RunTime checks

<http://resources.esri.com/help/9.3/arcgisengine/java/doc/7c061b6b-7317-4d53-aa15-a9c0a43f278b.htm>

Check sur différents types dans une liste

<https://stackoverflow.com/questions/25507741/run-time-type-checking-and-casting-in-java>

Compile time checks / Run time checks

<https://stackoverflow.com/questions/16745355/check-which-type-of-object-list-contains/16745403>

Allocation espaec pour un tableau

<https://www.informit.com/articles/article.aspx?p=1021579&seqNum=3>

Java Generic Method

<https://www.journaldev.com/1663/java-generics-example-method-class-interface#java-generic-type>

<https://www.journaldev.com/1663/java-generics-example-method-class-interface>

## TODO

* Tester la taille du tableau
* Tester le type des elements entrés dans le tableau
* / !\ Créer des exceptions spécifiques ClassCastException / !\_
* Creer une classe pour la conversion des elements insérés en entrée

<https://www.informit.com/articles/article.aspx?p=1021579&seqNum=3>

* Utiliser les classes Génériques

<https://www.journaldev.com/1663/java-generics-example-method-class-interface#java-generic-type>

/ !\ auto boxing un boxing / !\

Golfi

Declaration d'une méthode utilisant une liste générique

AbstractUpaDao.java AbstractUpaDao.java

Utilisation de la méthode findAll utilisant une liste generique

DepartmentIntegrationTest.java BasicIntegrationTest.java

Classe abstraite declarant la méthode utilisant un type generique

2 sous classe étandant la classe abstraite et utilisant soit un type integer soit un type string

Les 2 sous classes utiliseront une methode getInstance

Utilisation méthode Class

cf CustomScopeContext.java CustomScopeContextHolder.java

Runtime type checking and casting

<https://stackoverflow.com/questions/25507741/run-time-type-checking-and-casting-in-java>