# [how can i match two collections in scala by a field equality](https://stackoverflow.com/questions/64509092/how-can-i-match-two-collections-in-scala-by-a-field-equality)

[Ask Question](https://stackoverflow.com/questions/ask)

Asked

Active [today](https://stackoverflow.com/questions/64509092/how-can-i-match-two-collections-in-scala-by-a-field-equality?lastactivity)

Viewed 16 times

0

i have this code :

case class DataText(name:String)

val dataModels = Seq(DataText("a.dm"),DataText("b.dm"),DataText("c.dm"),DataText("d.dm"),DataText("e.dm"),DataText("f.dm"))

val dataReports = Seq(DataText("a.d0"),DataText("b1.do"),DataText("c2.do"),DataText("d.do"),DataText("e3.do"),DataText("f5.do"))

how can i match dataModels and dataReports, when an item in dataModels split by "." like name.split(".").head can match to dataReports split by "." like name.split(".").head

The result could be:

Seq(DataText("a.dm"),DataText("d.dm"))

i have tried with map and an embedded filter but don't work.

[scala](https://stackoverflow.com/questions/tagged/scala)[collections](https://stackoverflow.com/questions/tagged/collections)

[share](https://stackoverflow.com/q/64509092/4728560)  [edit](https://stackoverflow.com/posts/64509092/edit)    [flag](https://stackoverflow.com/questions/64509092/how-can-i-match-two-collections-in-scala-by-a-field-equality)

asked 3 hours ago

[javier\_orta](https://stackoverflow.com/users/8817285/javier-orta)



**189**7

7 bronze badges

[add a comment](https://stackoverflow.com/questions/64509092/how-can-i-match-two-collections-in-scala-by-a-field-equality)

## 1 Answer

[Active](https://stackoverflow.com/questions/64509092/how-can-i-match-two-collections-in-scala-by-a-field-equality?answertab=active" \l "tab-top)[Oldest](https://stackoverflow.com/questions/64509092/how-can-i-match-two-collections-in-scala-by-a-field-equality?answertab=oldest" \l "tab-top)[Votes](https://stackoverflow.com/questions/64509092/how-can-i-match-two-collections-in-scala-by-a-field-equality?answertab=votes" \l "tab-top)

0

I would transform dataReports into a Set of the target sub-elements for filtering via contains (which is a constant time O(1) operation):

val dataReportsSet = dataReports.map(\_.name.split("\\.")(0)).toSet

dataModels.filter(dm => dataReportsSet.contains(dm.name.split("\\.")(0)))

// res1: Seq[DataText] = List(DataText(a.dm), DataText(d.dm))

**Java HashSet** class is used to create a collection that uses a hash table for storage. It inherits the AbstractSet class and implements Set interface. The important points about **Java HashSet** class are: **HashSet** stores the elements by using a mechanism called hashing.

case class DataText(name:String)

val dataModels = Seq(DataText("a.dm"),DataText("b.dm"),DataText("c.dm"),DataText("d.dm"),DataText("e.dm"),DataText("f.dm"))

val dataReports = Seq(DataText("a.d0"),DataText("b1.do"),DataText("c2.do"),DataText("d.do"),DataText("e3.do"),DataText("f5.do"))

val dataReportsSet = dataReports.map(theData =>{

println(theData)

theData.name.split("\\.")(0)}).toSet

println(dataReportsSet)

val result = dataModels.filter(dm => dataReportsSet.contains(dm.name.split("\\.")(0)))

println(result)

// res1: Seq[DataText] = List(DataText(a.dm), DataText(d.dm))

Output:

DataText(a.d0)

DataText(b1.do)

DataText(c2.do)

DataText(d.do)

DataText(e3.do)

DataText(f5.do)

HashSet(a, f5, e3, d, c2, b1)

List(DataText(a.dm), DataText(d.dm))