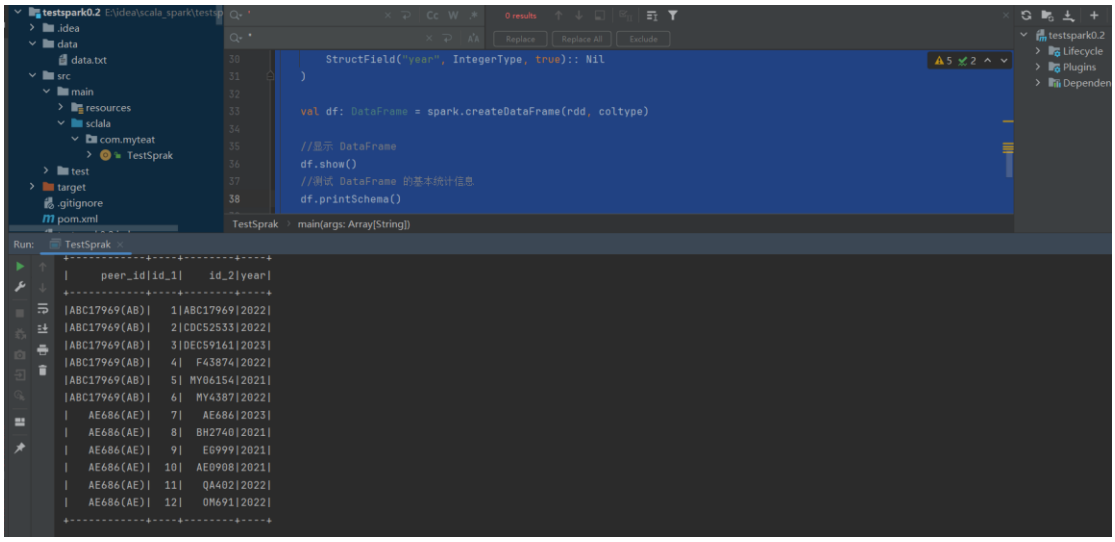
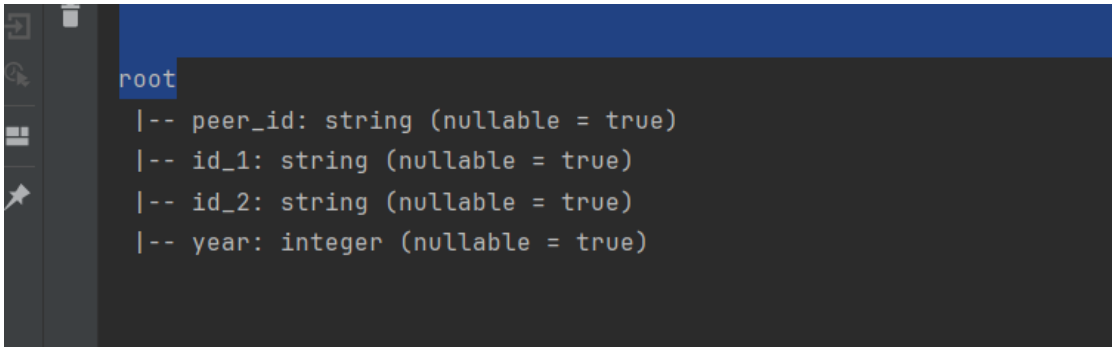
# RDD to DataFrame



Step1、For each peer_id, get the year when peer_id contains id_2, for example for ‘ABC17969(AB)’ year is 2022.

```
WARNING: All illegal access operations will be denied in a future re
step1 is ====
[ABC17969(AB),1,ABC17969,2022,2022]
[ABC17969(AB),2,CDC52533,2022,null]
[ABC17969(AB),3,DEC59161,2023,null]
[ABC17969(AB),4,F43874,2022,null]
[ABC17969(AB),5,MY06154,2021,null]
[ABC17969(AB),6,MY4387,2022,null]
[AE686(AE),7,AE686,2023,2023]
[AE686(AE),8,BH2740,2021,null]
[AE686(AE),9,EG999,2021,null]
[AE686(AE),10,AE0908,2021,null]
[AE686(AE),11,QA402,2022,null]
[AE686(AE),12,OM691,2022,null]
```

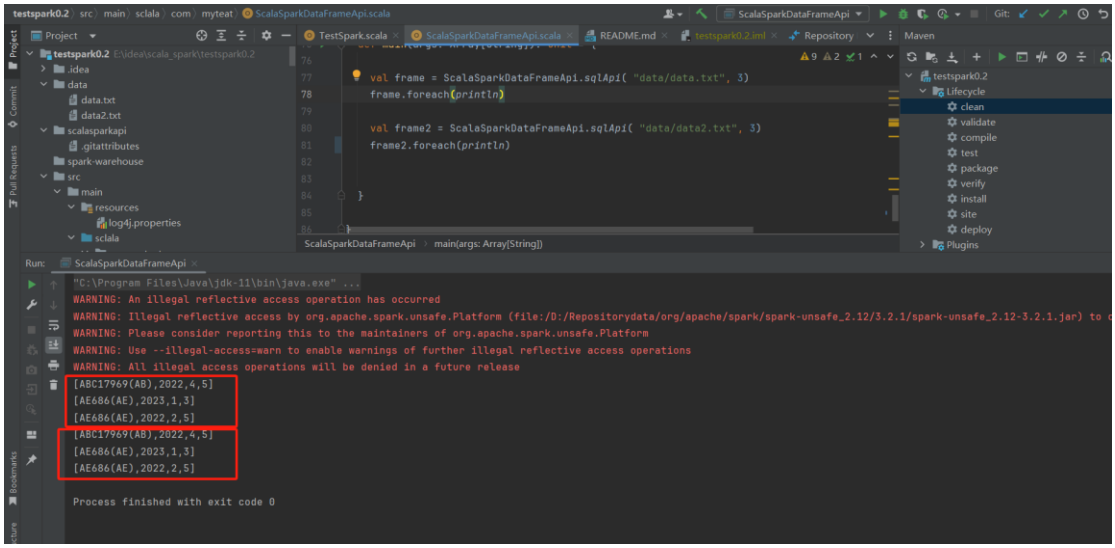
**Step2、Given a size number, for example 3. For each peer_id count the number of each year (which is smaller or equal than the year in step1).**

```
step2 is ====
[2022,4]
[2021,1]
[2023,1]
[2021,3]
[2022,2]
```

Step3、Order the value in step 2 by year and check if the count number of the first year is bigger or equal than the given size number. If yes, just return the year. If not, plus the count number from the biggest year to next year until the count number is bigger or equal than the given number. For example, for 'AE686(AE)', the year is 2023, and count are

```
step3 is ====
[ABC17969(AB),2022,4,5]
[AE686(AE),2023,1,3]
[AE686(AE),2022,2,5]
```

# ScalaSparkApi




# ScalaTest unit test