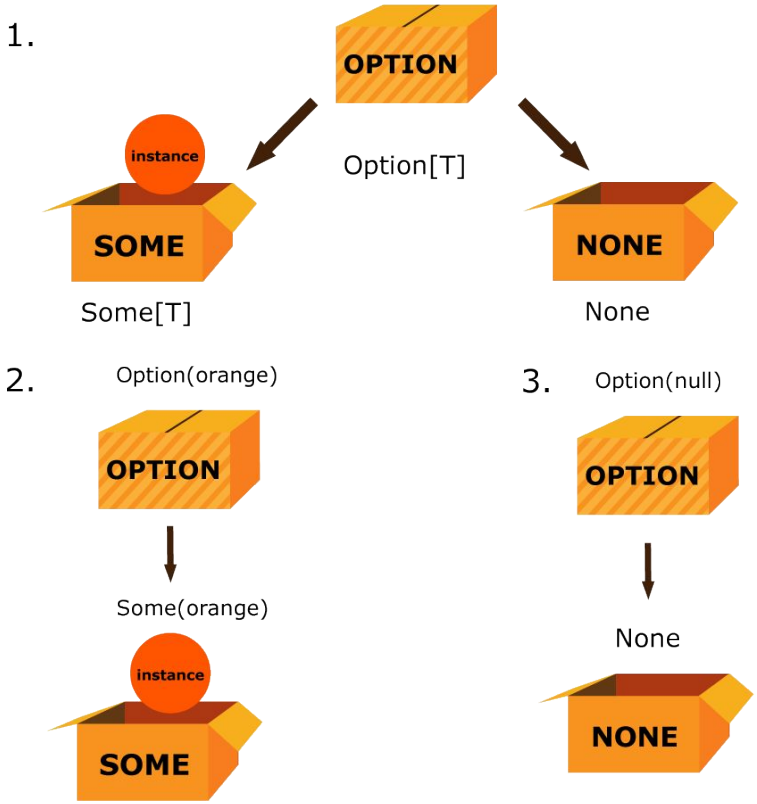
План презентации

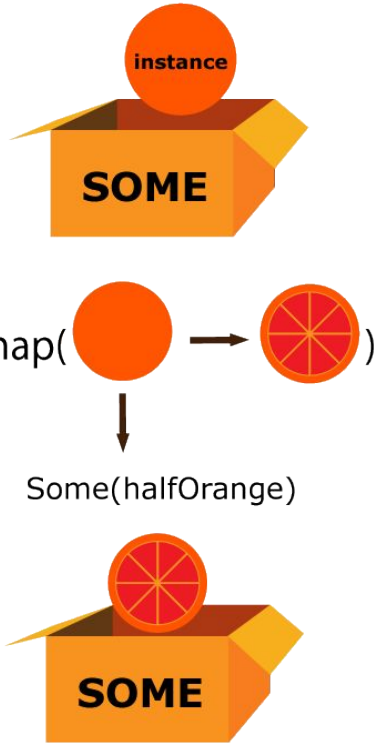
1. Option
2. List
3. Try
4. For и сахар

Option


1.
OPTION
Option[T]
instance
SOME
Some[T]
NONE
None
2.
Option(orange)
OPTION
Some(orange)
instance
SOME
3.
Option(null)
OPTION
None
NONE

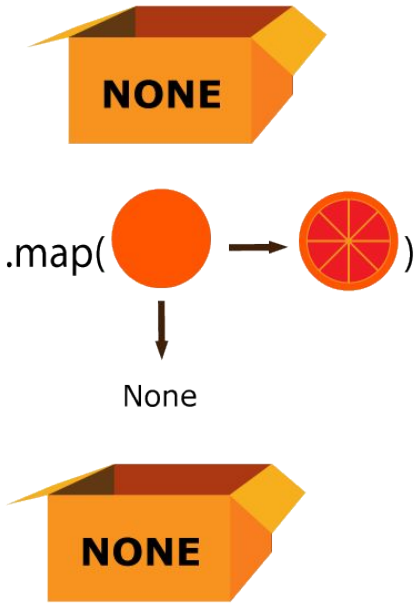
4.

Some(orange)



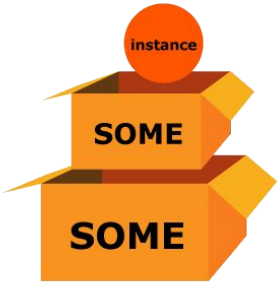
5.

None



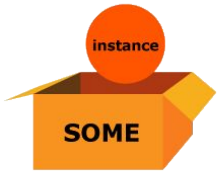
6.

Some(Some(orange))



.flatten

Some(orange)



7.

Some(None)

.flatten

None

8.

None

.flatten

None

`flatMap = flatten + map`

9.

Some(orange)

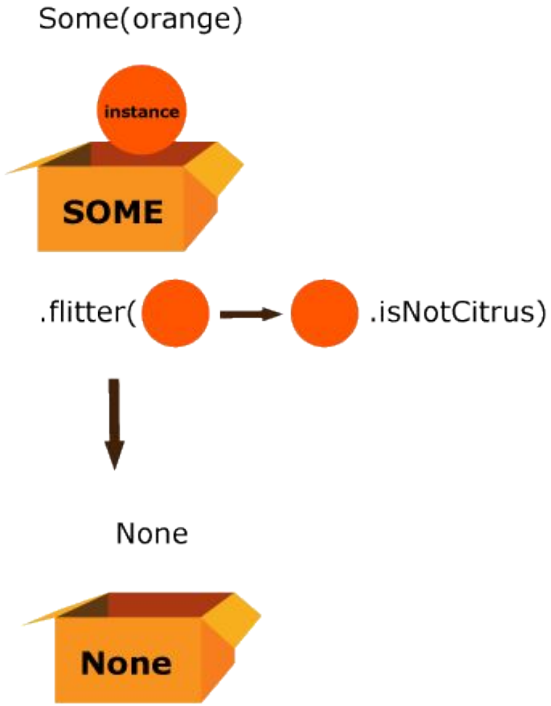
Some(orange)

10.

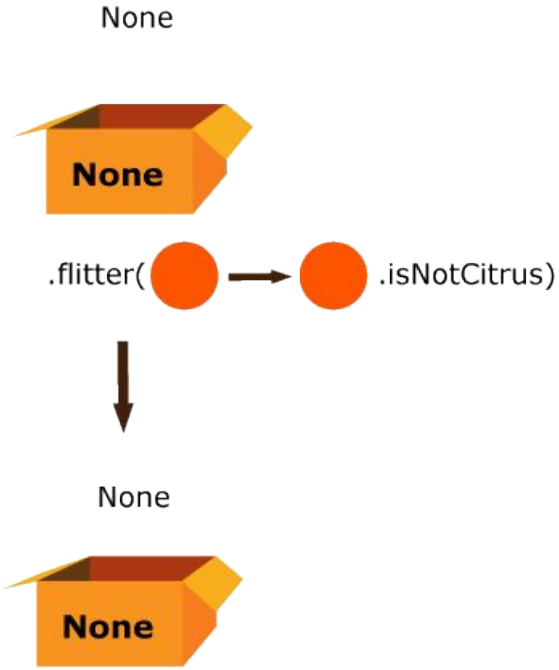
None

None

11.

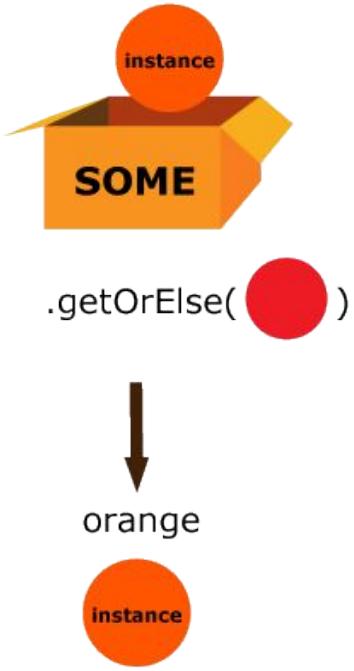

Some(orange)
instance
SOME
.flitter(
.isNotCitrus)
None
None

12.


None
None
.flitter(
.isNotCitrus)
None
None

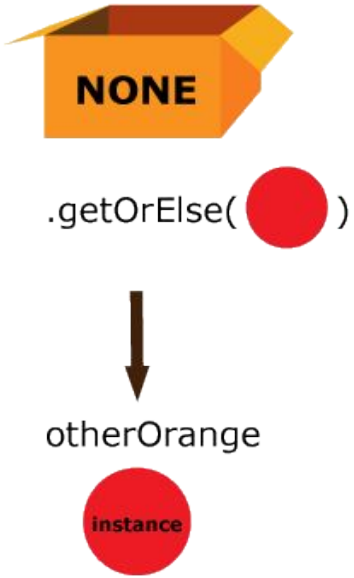
13.

Some(orange)



14.

None



15. Some(orange)

Some(orange)

16. None

Some(otherOrange)

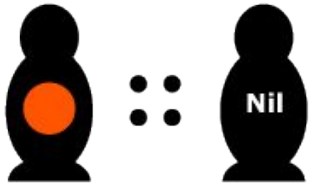
Some(orangeOrange)

17.

Some(orange)

.toList

List(orange)



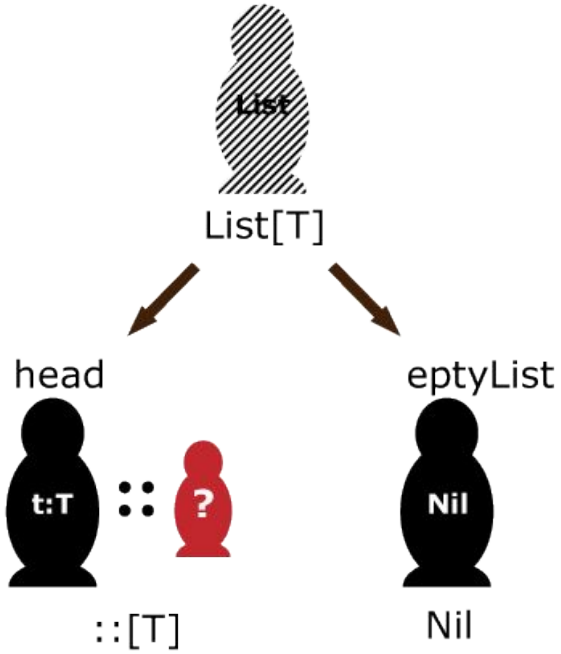
18.

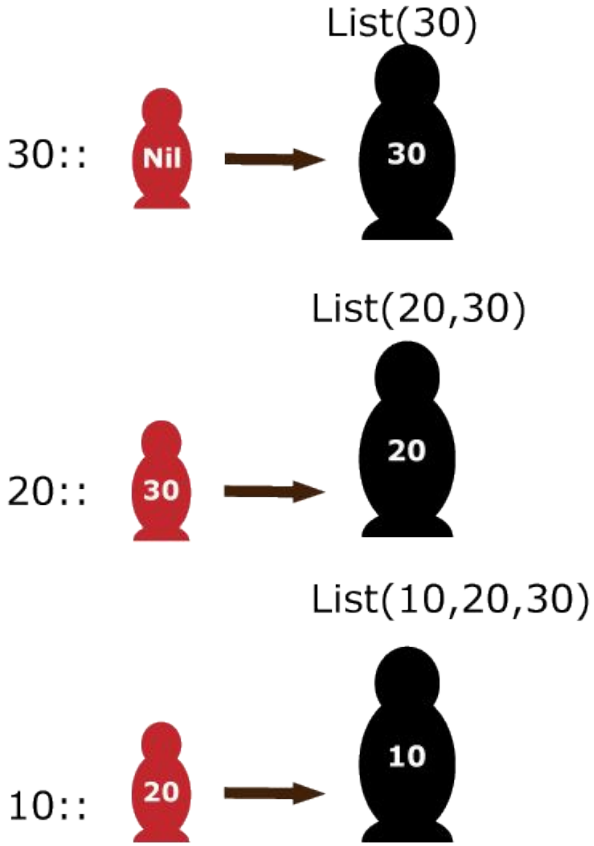
None

.toList

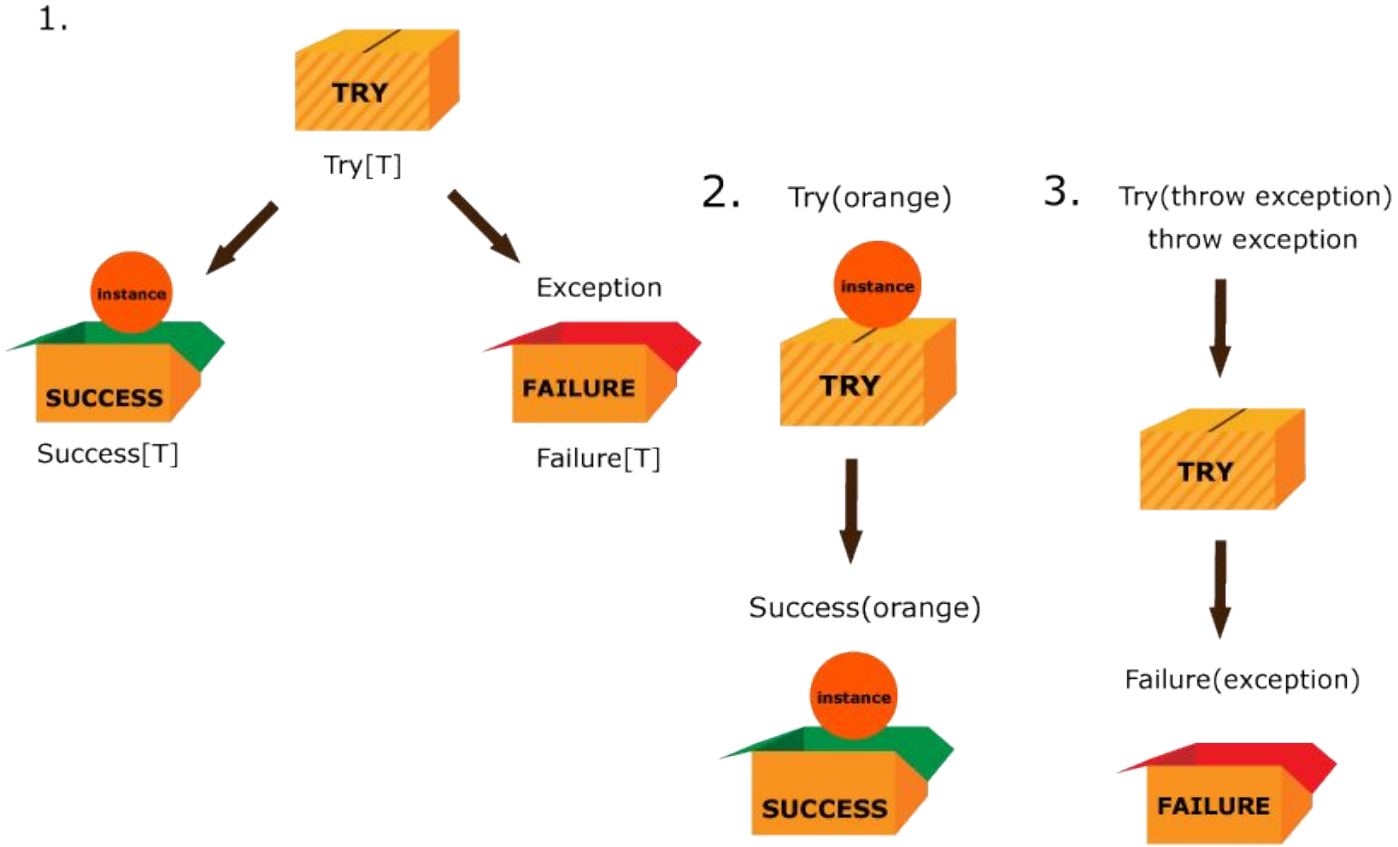
List()

List


List
List[T]
head
eptyList
t:T
::
?
Nil
::[T]
Nil

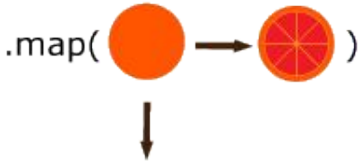

List(30)
30::
Nil
30
List(20,30)
20::
30
20
List(10,20,30)
10::
20
10

Try


1.
TRY
Try[T]
instance
SUCCESS
Success[T]
Exception
FAILURE
Failure[T]
2.
Try(orange)
instance
TRY
Success(orange)
instance
SUCCESS
3.
Try(throw exception)
throw exception
TRY
Failure(exception)
FAILURE

4.

Success(orange)

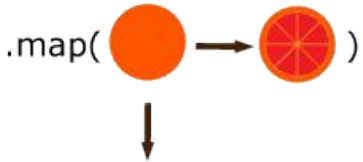


Success(halfOrange)

5.

Failure(exception)

exception



Failure(exception)

exception

5.

Success(Success(orange))

.flatten

Success(orange)

6.

Success(Failure(exception))

exception

.flatten

Failure(exception)

exception

7.

Failure(exception)

exception

.flatten

Failure(exception)

exception

8.

Success(orange)

Success(orange)

9.

Failure(exception)

Failure(exception)

exception

10.

Success(orange)

Failure(exception)

exception

11.

Failure(exception)

exception

Failure(exception)

exception

12.

Success(orange)

.recover(exception →)

Success(orange)

13.

Failure(exception)

exception

.recover(exception →)

Success(otherOrange)

14. Success(orange)

.recoverWith(exception →

)

Success(orange)

15. Failure(exception)

exception

.recoverWith(exception →

)

Success(otherOrange)

16.

Success(orange)

.getOrElse ( )

Orange

17.

Failure(exception)

exception

.getOrElse ( )

otherOrange

18.

Success(orange).orElse(Success(otherOrange))

Success(orange)

19.

Success(orange).orElse(Success(otherOrange))

Success(otherOrange)

20.

Success(orange)

.toOption

Some(orange)

21.

Failure(exception)

exception

.toOption

None

Контакты:

Субботин Александр, разработчик, Data engineer

телеграм: @sa1amandraa

Рыженкова Татьяна, дизайн и оформление презентации

телеграм: @TatianaRyzhenkova

Полезные ссылки:

https://github.com/anton-k/ru-neophyte-guide-to-scala..

https://medium.com/@mdubakov/абстракция-и-сложность-6..

https://docs.scala-lang.org/ru/tour/tour-of-scala.html

https://docs.scala-lang.org/tour/tour-of-scala.html

Ссылки на курсы:

https://www.udemy.com/course/scala-for-java-developer..

https://www.coursera.org/specializations/scala

https://stepik.org/course/16243/syllabus?auth=registr..

Ссылка не текущую презентацию и код:

https://github.com/salamandraa/ScalaInPictures

Ссылка на предыдущую презентацию и код:

https://github.com/salamandraa/ScalaForLittle

СПАСИБО ЗА ВНИМАНИЕ !