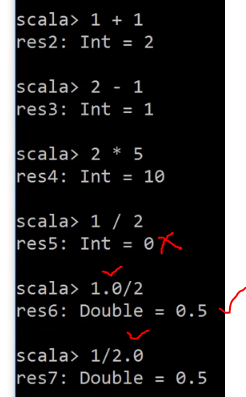
**SCALA Basics**



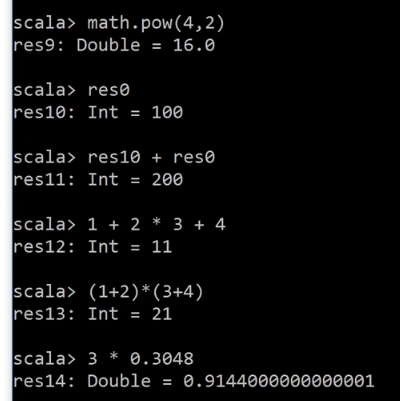




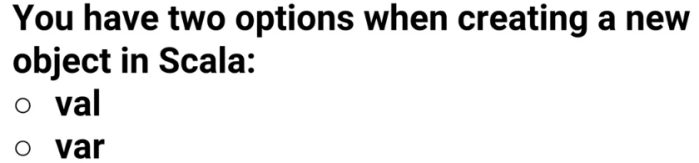
scala>100

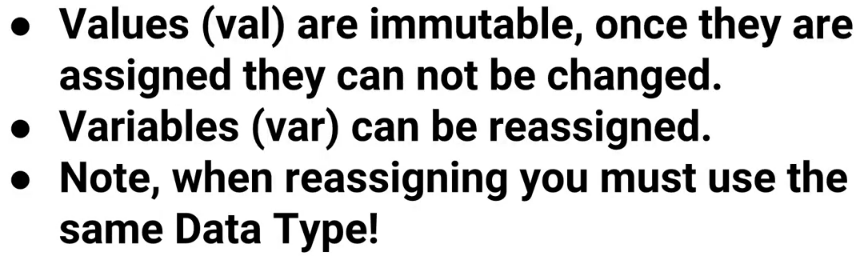
res0:100

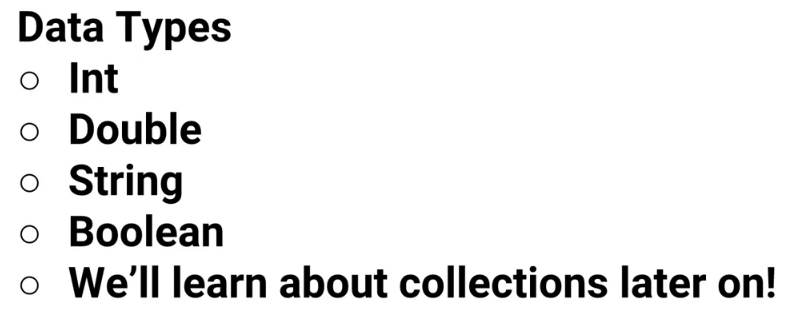
math.pow(2,5)

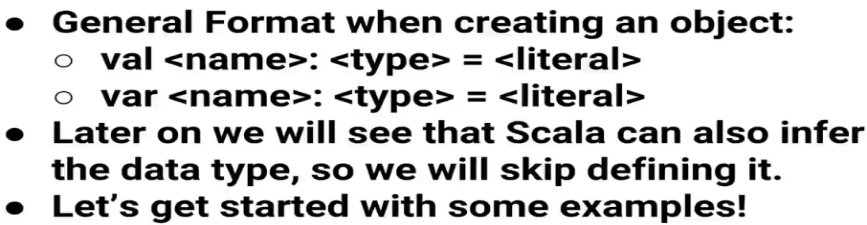


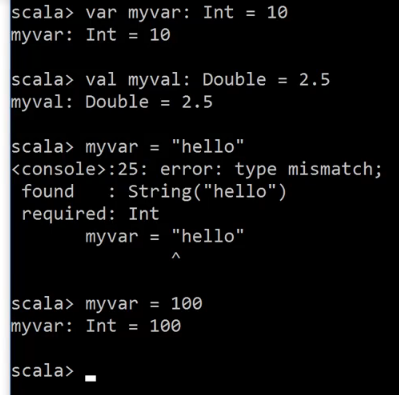
**Values and Variables**

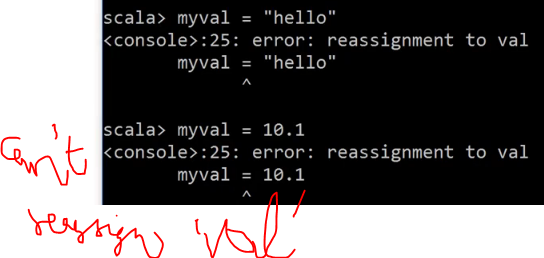
****

****

****

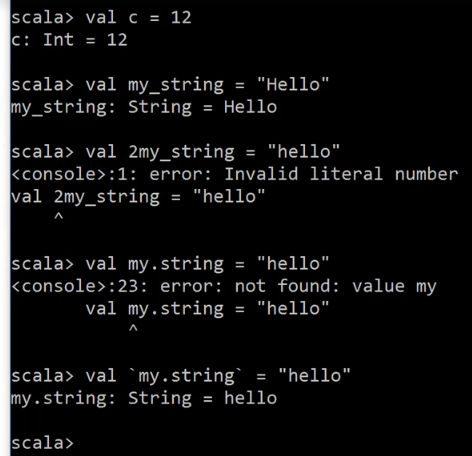
****

****

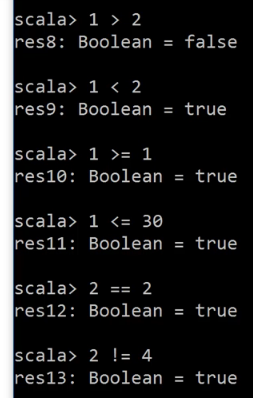
****

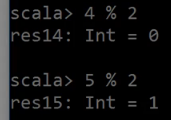
**Scala Infer datatype (no need to specify datatype )**

**literal cannotstart with number.**

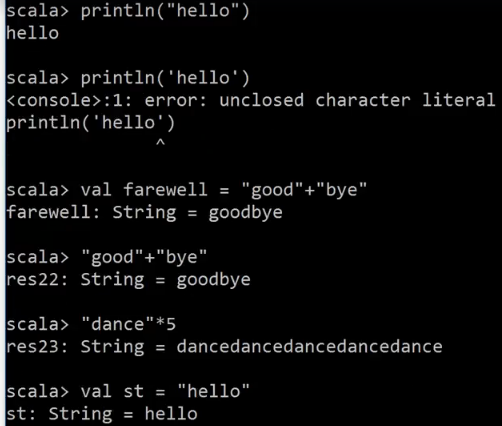
****

****

****

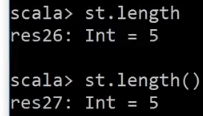
****

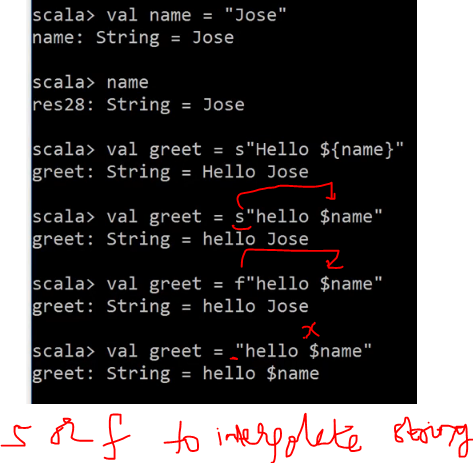
****

****

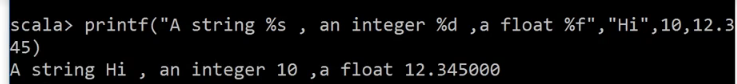
**st.(hit tab ..it shows all methods that can be applied)**

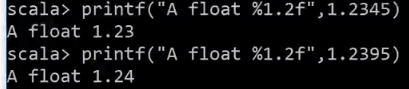
****

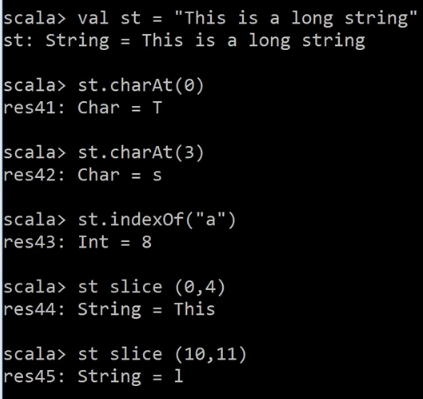
****

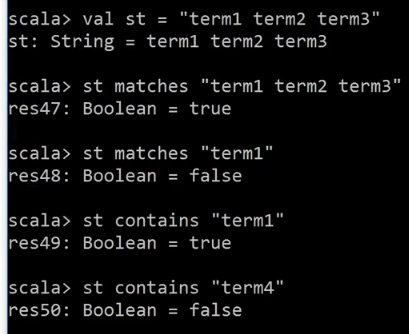
****

**Interpolate string with printf**

****

****

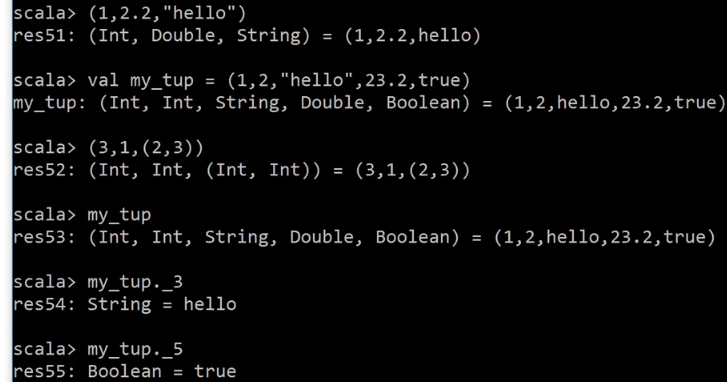
****

****

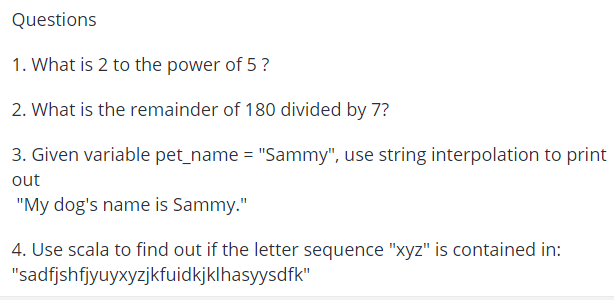
****

**Ordered sequence of elements that holds multiple types.**

**Index starts from 1.**

****

**Assessment**

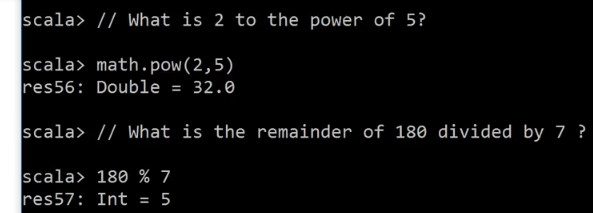
****

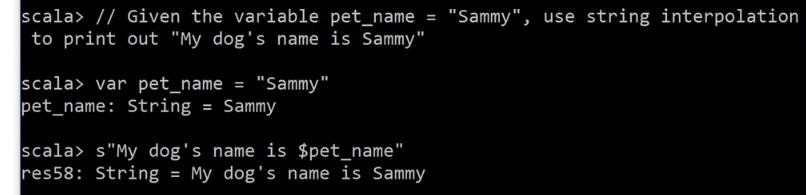
1.math.pow(2,5)

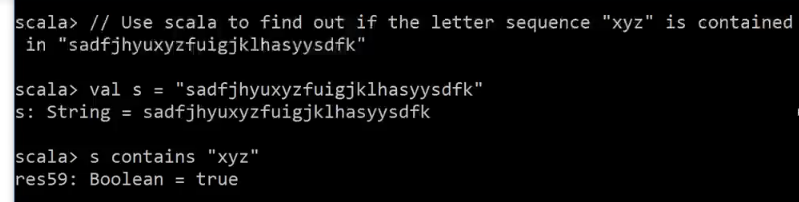
2.180%7

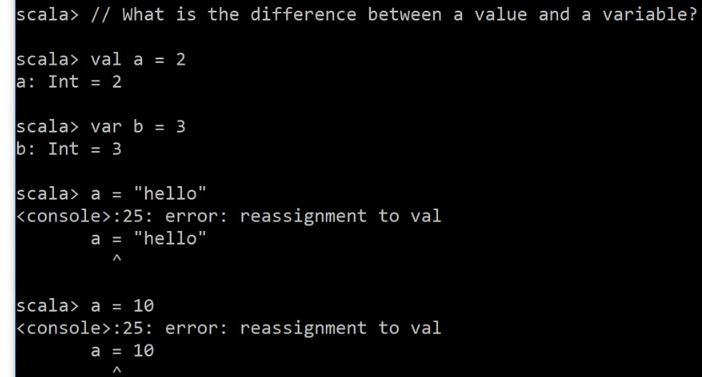
3.val pet\_name = "Sammy"

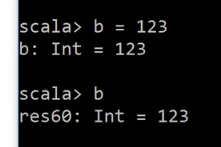
printf (s"My dog's name is $pet\_name")



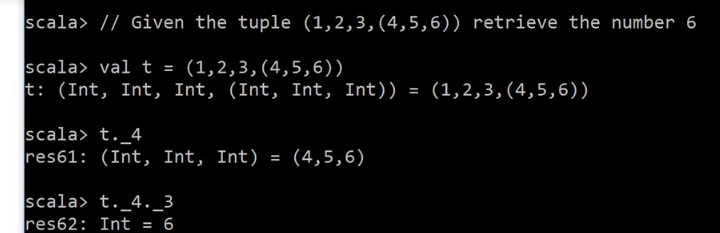


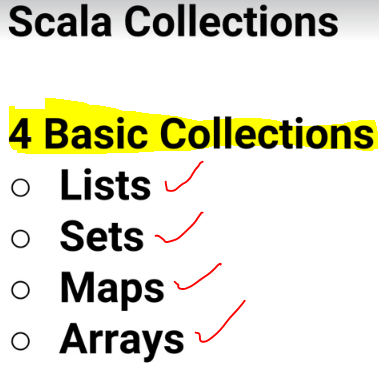






val cannot be reassigned whereas var can be reassigned.





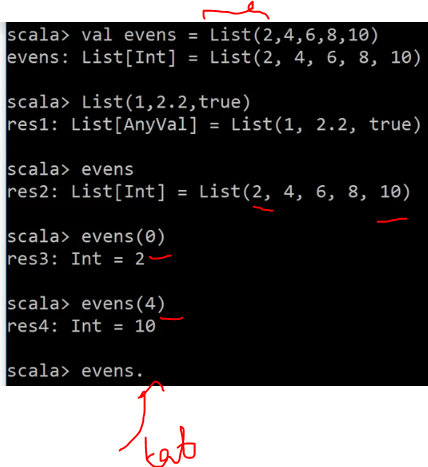
scala shell > scala interpreter

**Lists**

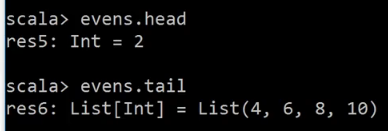
- Singly linked lists

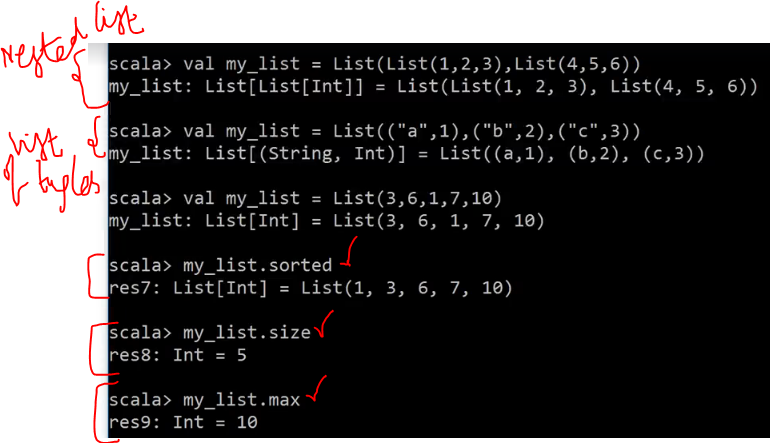
- Immutable sequence of elements

- Zero-based index

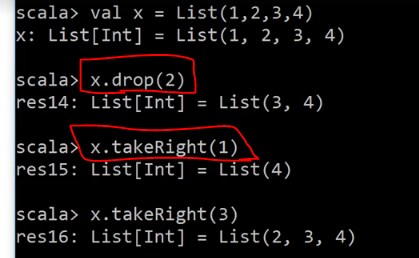


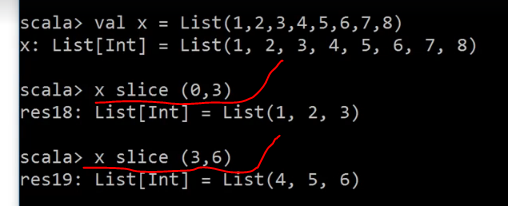






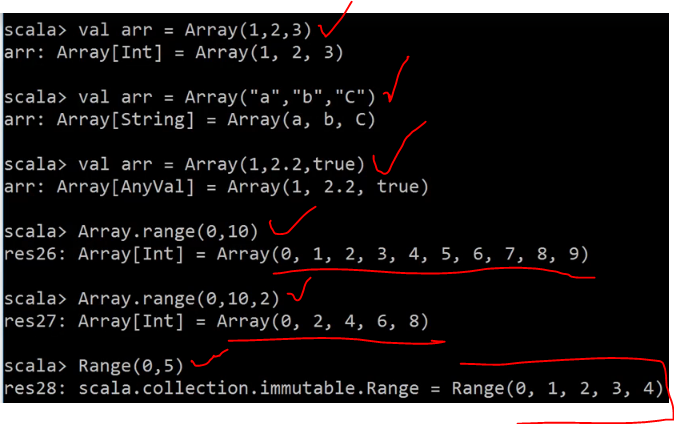






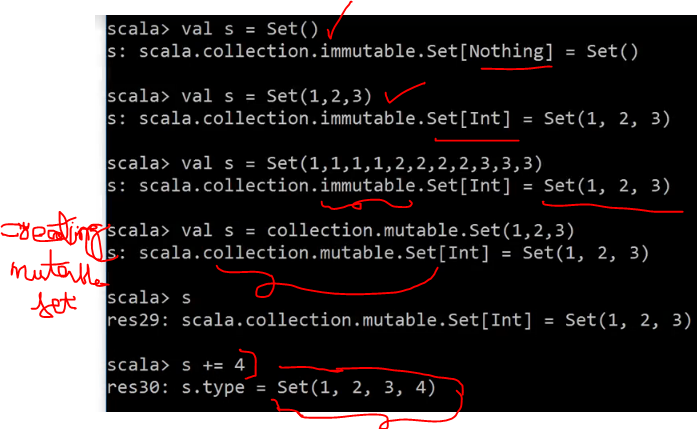
**Arrays**

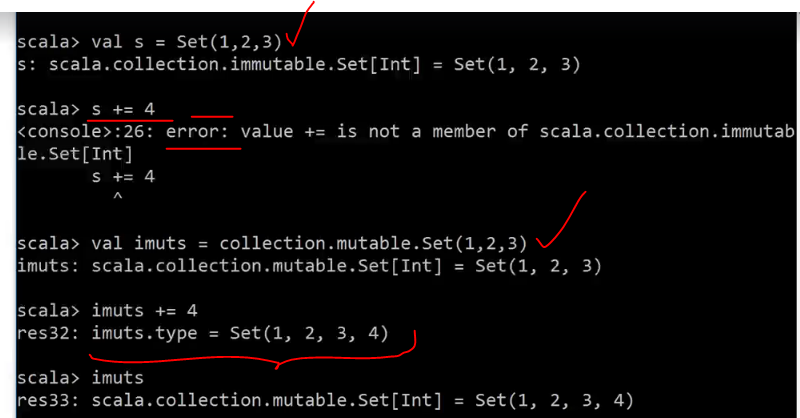
****

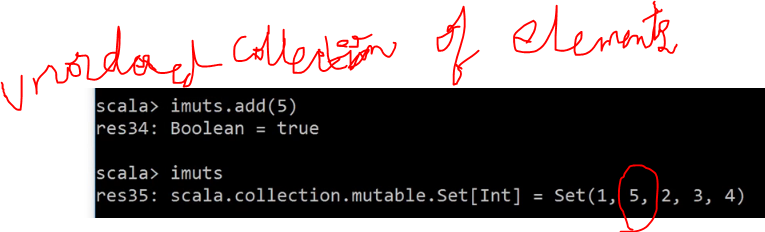
****

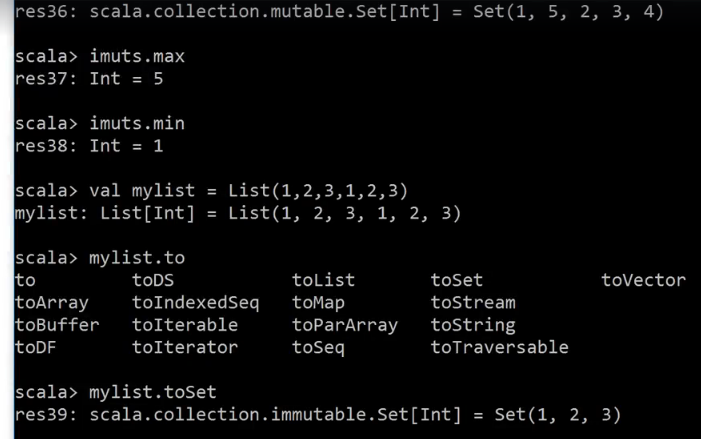
**Sets**

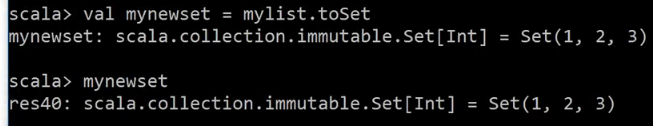
Set of unique elements /no duplicates.





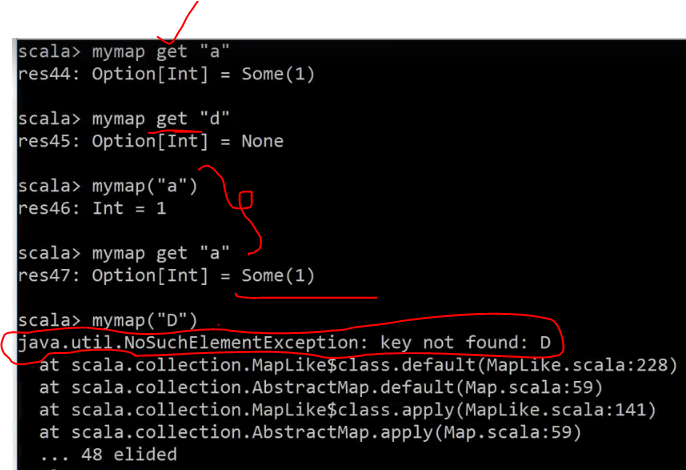


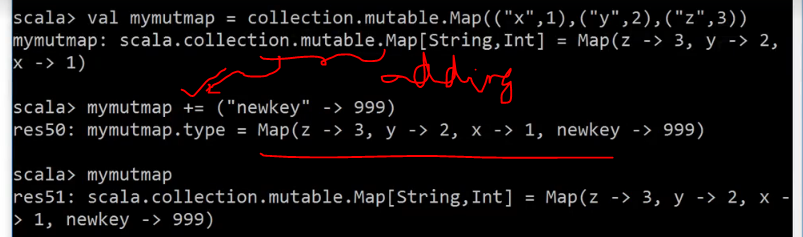




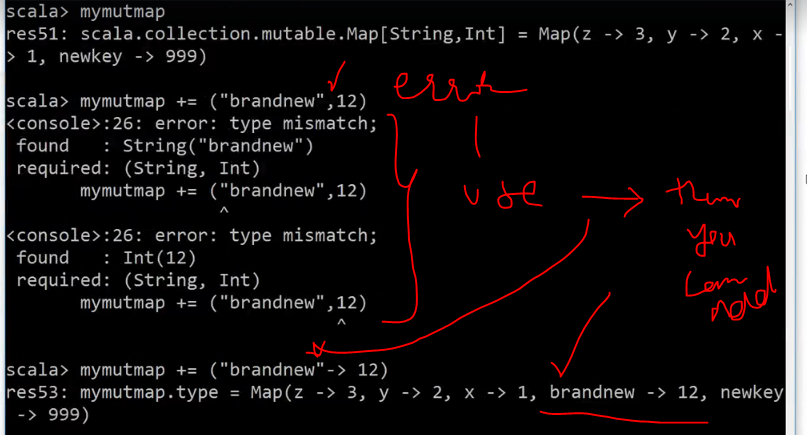
**Maps**

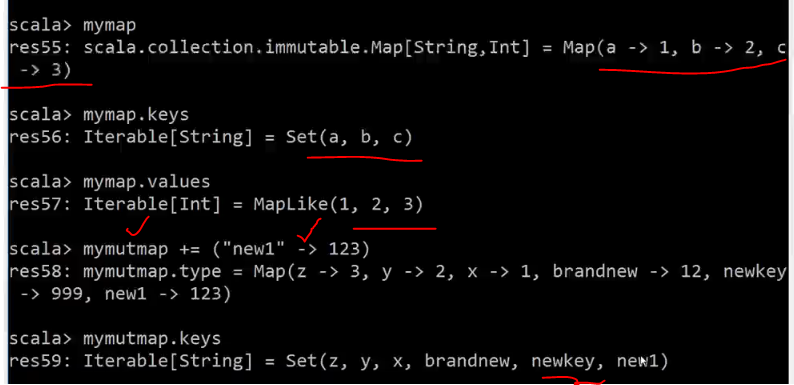


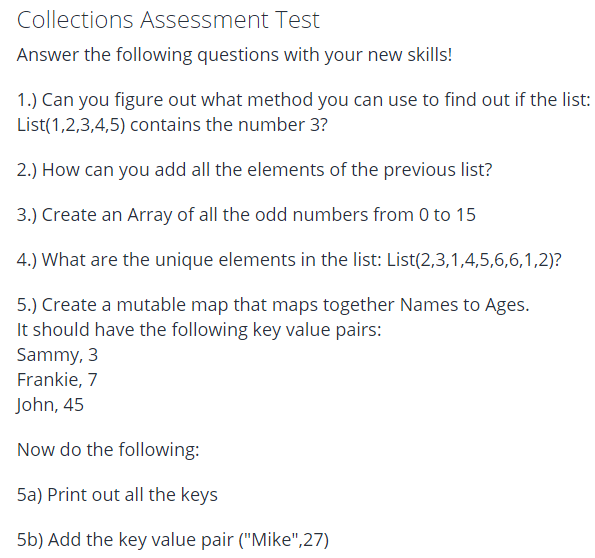




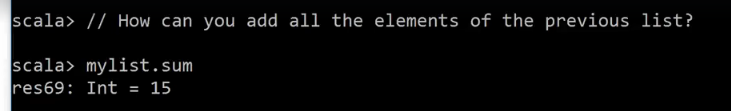
while adding use -> (arrow) otherwise error.

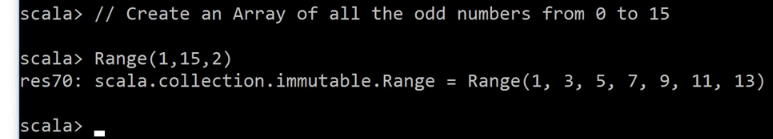




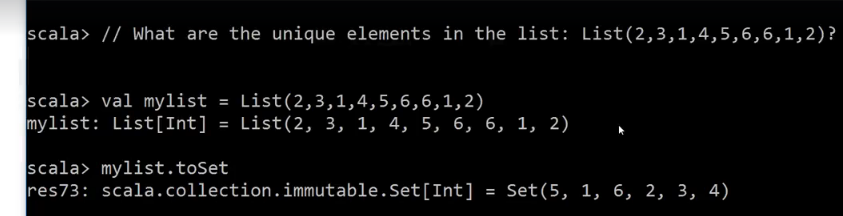


1. 

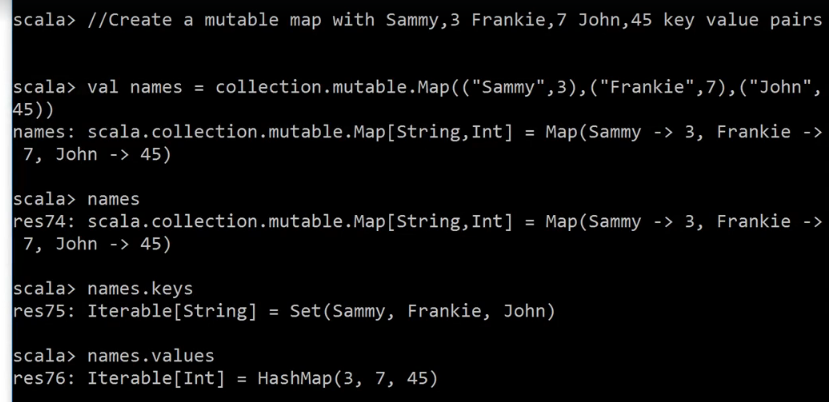
2. 

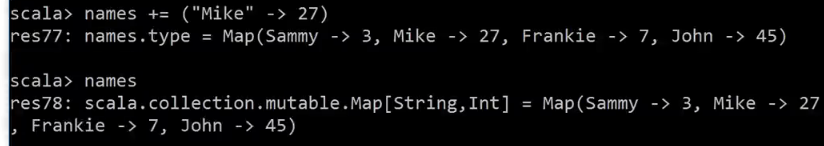
3. 

4.



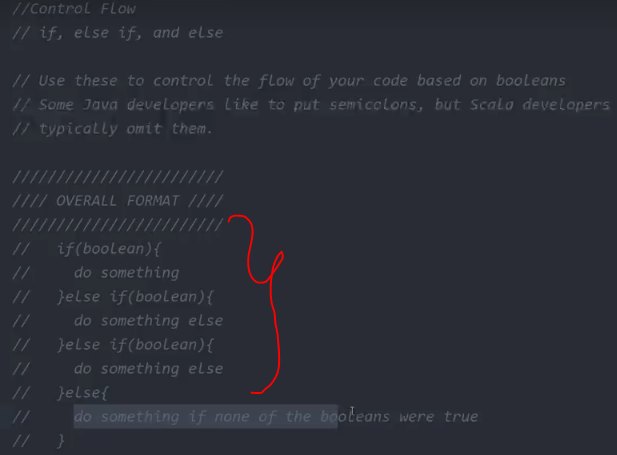
5.

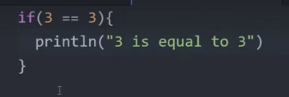




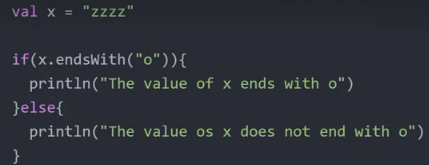
**Control Flows**

**if,else if ,else**

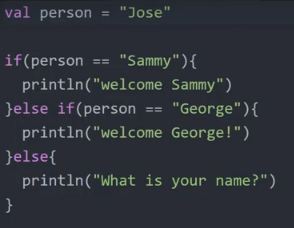
****

****

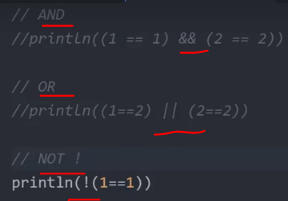
**3 is equal to 3**

****

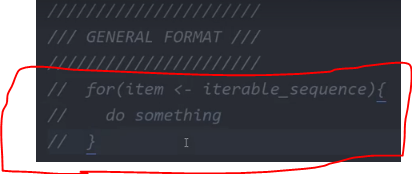
****

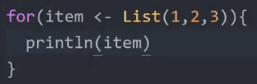
****

**What is your name?**

****

**for**

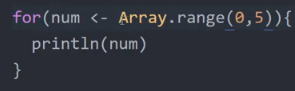
****

****

**1**

**2**

**3**

****

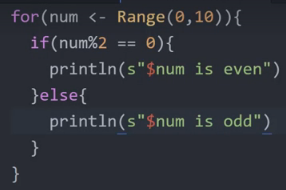
**0**

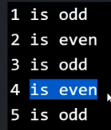
**1**

**2**

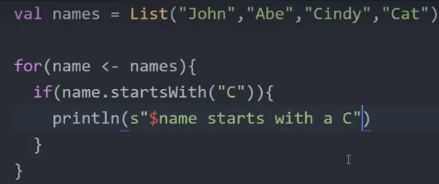
**3**

**4**

**Print odd or even  
**

****

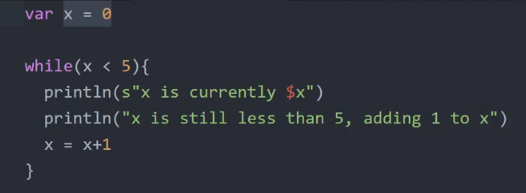
**Eg:**

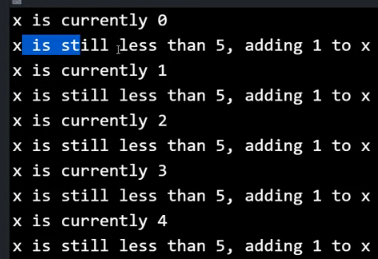
****

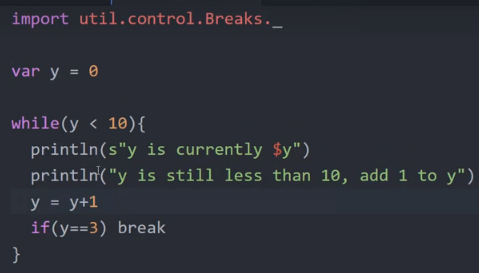
**Cindy starts with a C**

**Cat starts with a C**

**While**

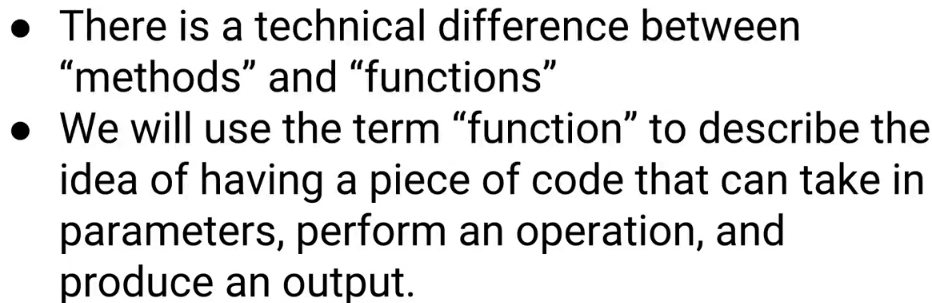
****

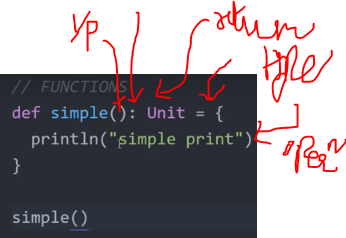
****

****

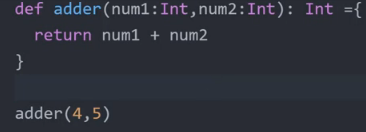
****

**Functional Programming**

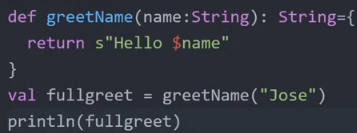
****

****

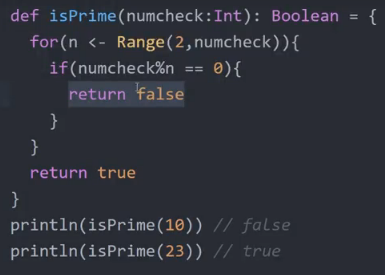
simple print



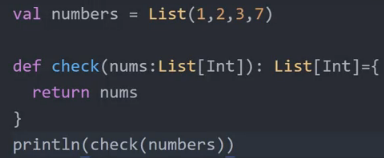
9



Hello Jose

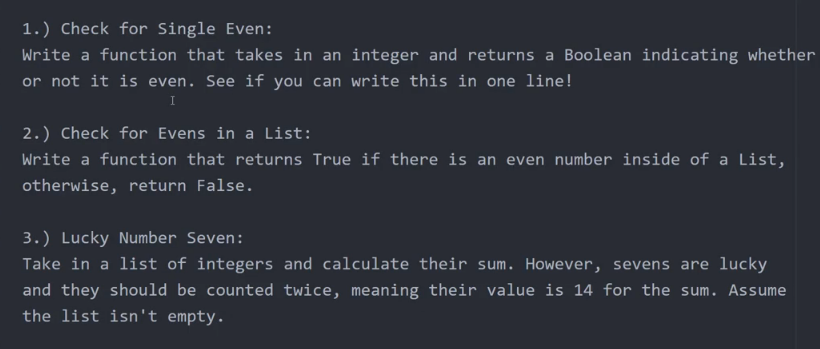


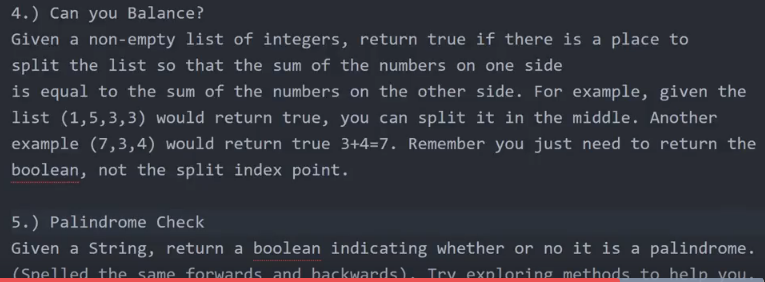
collections



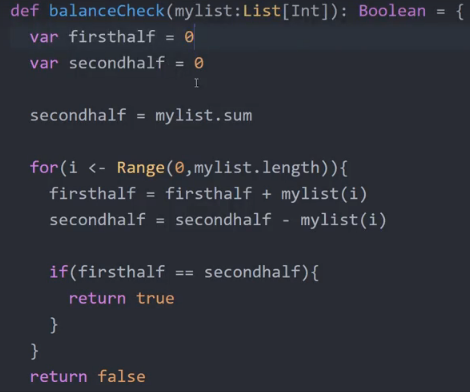
List(1,2,3,7)

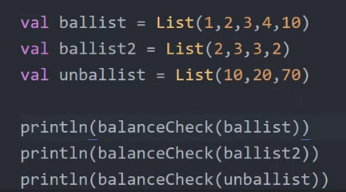
**Assessment**

****

****

**4.**

****

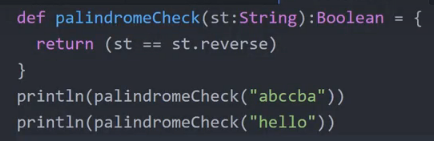
****

**true**

**true**

**false**

**5.palindrome**

****