**Getting access to python**

<https://docs.python.org/3.6/using/index.html>

**Getting access to scala**

<https://docs.scala-lang.org/>

**Helpful resource for scala**

<https://www.artima.com/scalazine>

## Scala Web Frameworks

Scala is being used everywhere and importantly in enterprise web applications. You can check a few of the most popular Scala web frameworks −

* [The Lift Framework](http://liftweb.net)
* [The Play framework](http://www.playframework.org/)
* [The Bowler framework](https://github.com/bowler-framework/bowler-quickstart)

A website with questions and answers to help learning scala.

* <https://www.codequizzes.com/scala>

The official scala documentation; a beginning.

* <https://docs.scala-lang.org/tour/basics.html>

To access Python-3 at UB

**use python3**

To access Scala at UB

**use scala**

To start the Scala IDE for Eclipse at UB

**use eclipse-scala**

Additional resource material

https://www.codequizzes.com/scala

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

<https://www.tutorialspoint.com/scala/scala_basic_syntax.htm>

<https://www.tutorialspoint.com/scala/scala_data_types.htm>

<https://www.tutorialspoint.com/scala/scala_variables.htm>

<https://www.tutorialspoint.com/scala/scala_operators.htm>

<https://www.tutorialspoint.com/scala/scala_classes_objects.htm>

<https://www.tutorialspoint.com/scala/scala_functions.htm>

<https://www.tutorialspoint.com/scala/scala_closures.htm>

**Exercises – Day01 – Value 10 pts.**

Verify access to Python

Verify access to scala via Eclipse (or Intellij)

**work tutorial**

robertovormittag.net/tutorials/ learn-scala-in-30-minutes-a-tutorial-for-java-programmers/

Please write the following programs in Scala.

1. Write and test a function that takes the user’s name and age as parameters. Print a message which addresses the user by name and tells the person the year that they will turn 65 years of age.

Test with ages 19, 65 and 95

1. Write and test a function that returns the maximum of two numbers.
2. Write and test a function called **summer\_winter**that takes a number.
   1. If the number is divisible by 3, it should return “Summer”.
   2. If it is divisible by 5, it should return “Winter”.
   3. If it is divisible by both 3 and 5, it should return “SummerWinter”.
   4. Otherwise, it should return the original number.

Test with numbers, 15, 33, 97 and 100

1. Write and test a function for checking the speed of automobile drivers. This function should have one parameter: speed.
   1. If speed is less than 55, it should print “Ok”.
   2. Otherwise, for every 5 above the speed limit (55), it should give the driver one demerit point and print the total number of demerit points. For example, if the speed is 80, it should print: “Points: 5”.
   3. If the driver gets more than 12 points, the function should print: “License suspended”

Test with speeds of 60, 90, 119 and 120

1. Write and test a function called **oddEven**that takes a parameter called **limit.**It should print all the numbers between 0 and **limit** with a label to identify the even and odd numbers. For example, if the **limit** is 3, it should print:

0 EVEN

1 ODD

2 EVEN

3 ODD

Test with the number 7.

1. Write and test a function that returns the sum of multiples of 3 and 5 between 0 and **limit** (parameter). For example, if limit is 20, it should return the sum of 3, 5, 6, 9, 10, 12, 15, 18, 20.

Test with 20, 28 and 96

1. Write and test a function that takes a numeric argument and returns true if the argument is a prime number (and false if not).

Test with 7, 49 and 97.

1. Write and test a function that prints all the prime numbers between 0 and **limit**where limit is a parameter to the function.

Test with a limit of 100

1. Write and test a function named sort that accepts 3 numeric arguments and prints the 3 numbers in numeric order. Ex: sort(27, 66, 9) prints 9, 27, 66

Test with all six possible orders;

1, 2, 3

1, 3, 2

2, 1, 3

2, 3, 1

3, 1, 2

3, 2, 1

1. Write and test a function that accepts a positive number as the parameter and determines if the number is a “Perfect Number”. A Perfect Number is one whose divisors sum to the number itself.

Ex: 28 = 1 + 2 + 4 + 7 + 14

Test with 28, 272, 496 and 8128.