```scala
package com.tavor.project
import scala.xml._
import java.net._
import scala.util.control.Breaks._
import org.apache.log4j.Logger;
import org.apache.log4j.PropertyConfigurator;

class Model() extends ModelInterFace {

  val url = new URL("http://www.boi.org.il/currency.xml")  //returns URL object
  val conn = url.openConnection              //returns URLConnection object
  val doc = XML.load(conn.getInputStream)    //returns Elem object represent XML element

  def getDate(): String = (doc \\ "LAST_UPDATE").text //return date of last update

  def getData(): Array[String] = {

    var vec: Array[String] = new Array[String](28)
    var i: Int = 0

    for (elm <- (doc \\ "CURRENCY")) { //parse the xml currencies, return data as an array
      var rate: Double = 0
      rate = (elm \\ "RATE").text.toDouble
      if ((elm \\ "UNIT").text.toDouble == 10) {
        rate = rate / 10
      }
      if ((elm \\ "UNIT").text.toDouble == 100) {
        rate = rate / 100
      }
      vec(i) = ((elm \\ "COUNTRY").text)
      i += 1
      vec(i) = rate.toString()
      i += 1
    }
    vec
  }

  def Convert(from: String, amount: Double, to: String): Double = {
    var fromRate: Double = 1
    var toRate: Double = 1
    var shekel: Double = 1
    var Result: Double = 0

    breakable {
```

```
45        for (elm <- (doc \\ "CURRENCY")) {
46          if (from == "ILS") { //if shekel then No-change
47            fromRate = 1
48            break
49          }
50          if (from == (elm \\ "CURRENCYCODE").text) { //if any "regular" coins
51            fromRate = ((elm \\ "RATE").text.toDouble)
52            break
53          }
54        }
55      }
56      if (from == "JPY") //JPY & LBP fix "to small" rates from 1 unit
57        fromRate = fromRate / 100
58      if (from == "LBP")
59        fromRate = fromRate / 10
60
61      breakable {
62        for (elm <- (doc \\ "CURRENCY")) {
63          if (to == (elm \\ "CURRENCYCODE").text) {
64            toRate = ((elm \\ "RATE").text.toDouble)
65            break;
66          }
67        }
68      }
69
70      if (from == "JPY")
71        fromRate = fromRate / 100 //JPY & LBP fix "to small" rates from 1 unit
72      if (from == "LBP")
73        fromRate = fromRate / 10
74
75      if (from != "ILS") { //calculate result
76        shekel = fromRate * amount
77        Result = shekel / toRate
78      }
79
80      if (from == "ILS") {
81        Result = amount / toRate
82      }
83      Result
84    }
85
86
87  }
88
89
```