

# Comp 311

# Functional Programming

Nick Vrvilo, Two Sigma Investments  
Robert “Corky” Cartwright, Rice University

September 6, 2018  
Supplementary Slides

# Using DrScala

# DrScala

- Available from the course homepage:

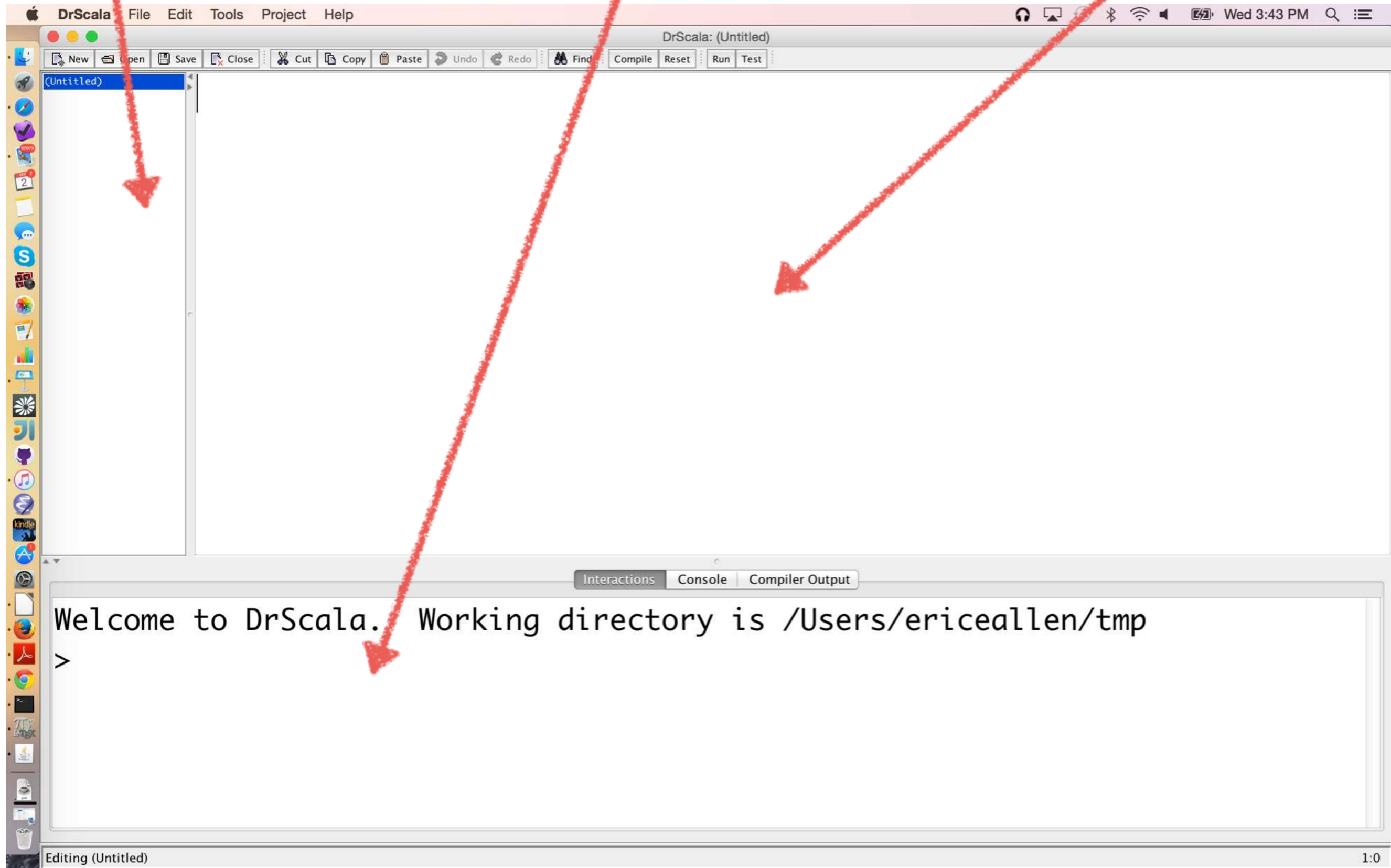
<https://comp311.rice.edu>

- A lightweight development environment well-suited to the exercises we will do in this class

# Open Files

# Interactions Pane

# Definitions Pane



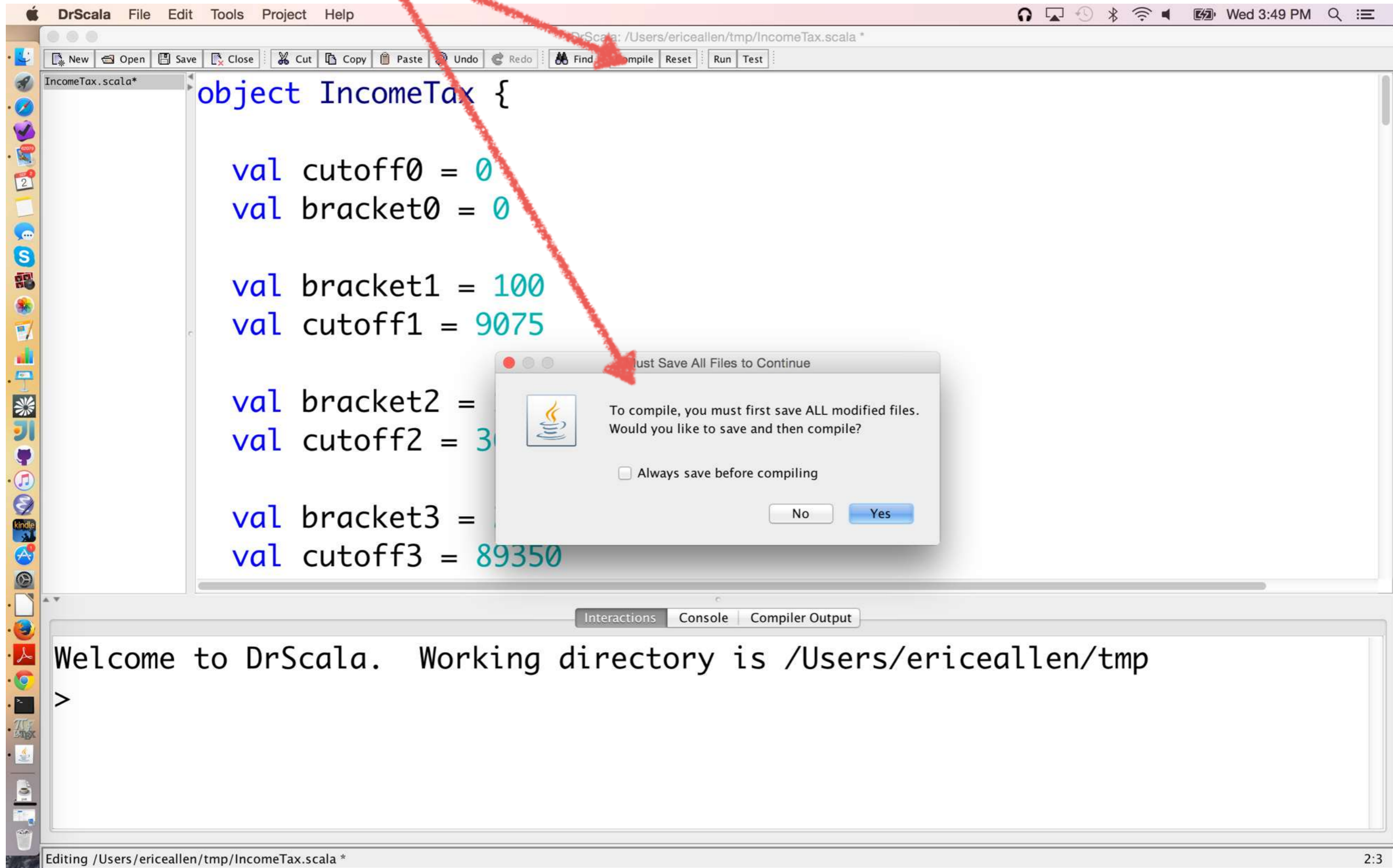
# Define your program in the definitions pane

The screenshot shows the DrScala IDE interface. The top menu bar includes 'DrScala', 'File', 'Edit', 'Tools', 'Project', and 'Help'. The title bar indicates the current file is 'DrScala: /Users/ericeallen/tmp/IncomeTax.scala'. The toolbar contains icons for 'New', 'Open', 'Save', 'Close', 'Cut', 'Copy', 'Paste', 'Undo', 'Redo', 'Find', 'Compile', 'Reset', 'Run', and 'Test'. The main editor area displays the following Scala code:

```
object IncomeTax {  
  |  
  val cutoff0 = 0  
  val bracket0 = 0  
  
  val bracket1 = 100  
  val cutoff1 = 9075  
  
  val bracket2 = 150  
  val cutoff2 = 36900  
  
  val bracket3 = 250  
  val cutoff3 = 89350
```

A red arrow points from the title 'Define your program in the definitions pane' to the code in the editor. Below the editor is a console pane with tabs for 'Interactions', 'Console', and 'Compiler Output'. The console displays the message: 'Welcome to DrScala. Working directory is /Users/ericeallen/tmp' followed by a prompt '>'. The status bar at the bottom shows 'Editing /Users/ericeallen/tmp/IncomeTax.scala \*' and the page number '2:3'.

# A prompt to save your program after hitting the Compile button



The screenshot shows the DrScala IDE interface. The main editor window displays the following Scala code:

```
object IncomeTax {  
  
  val cutoff0 = 0  
  val bracket0 = 0  
  
  val bracket1 = 100  
  val cutoff1 = 9075  
  
  val bracket2 =  
  val cutoff2 = 3  
  
  val bracket3 =  
  val cutoff3 = 89350
```

A dialog box titled "Must Save All Files to Continue" is displayed in the foreground. The dialog contains the following text:

To compile, you must first save ALL modified files.  
Would you like to save and then compile?

Always save before compiling

Buttons: No, Yes

The console window at the bottom of the IDE shows the following output:

```
Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
>
```

At the bottom of the IDE, the status bar indicates "Editing /Users/ericeallen/tmp/IncomeTax.scala \*".

The screenshot shows the DrScala IDE interface. The main editor window displays the following Scala code:

```
object IncomeTax {  
  |  
  val cutoff0 = 0  
  val bracket0 = 0  
  
  val bracket1 = 100  
  val cutoff1 = 9075  
  
  val bracket2 = 150  
  val cutoff2 = 36900  
  
  val bracket3 = 250  
  val cutoff3 = 89350  
}
```

Below the code editor, the 'Compiler Output' tab is active, showing the message: "Compilation completed. Output directory is: /Users/ericeallen/tmp". To the right of this message, a 'Compiler' dropdown menu is set to 'Scala version 2.12.0-M2', and there is a checked checkbox for 'Highlight source'. A red arrow points from the text 'Successful compilation reported in the Compiler Output tab' to the 'Compiler Output' tab.

Successful compilation reported  
in the Compiler Output tab

Interactions Console **Compiler Output**

Compilation completed. Output directory is:  
/Users/ericeallen/tmp

Compiler  
Scala version 2.12.0-M2

Highlight source

Editing /Users/ericeallen/tmp/IncomeTax.scala 2:3

DrScala File Edit Tools Project Help

DrScala: /Users/ericeallen/tmp/IncomeTax.scala

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

```
IncomeTax.scala
object IncomeTax {

    val cutoff0 = 0
    val bracket0 = 0

    val bracket1 = 100
    val cutoff1 = 9075

    val bracket2 = 150
    val cutoff2 = 36900

    val bracket3 = 250
    val cutoff3 = 89350
}
```

Interactions Console Compiler Output

Editing /Users/ericeallen/tmp/IncomeTax.scala

2:3

Console output from running  
a program printed here





The image shows a screenshot of the DrScala IDE. The main editor window displays the following Scala code for an `IncomeTax` object:

```
object IncomeTax {  
  
  val cutoff0 = 0  
  val bracket0 = 0  
  
  val bracket1 = 100  
  val cutoff1 = 9075  
  
  val bracket2 = 150  
  val cutoff2 = 36900  
  
  val bracket3 = 250  
  val cutoff3 = 89350  
}
```

Below the editor is an interactive console pane with tabs for "Interactions", "Console", and "Compiler Output". The "Interactions" tab is active, showing the following text:

```
Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
>
```

A red arrow points from the text "We can interact with the functions in our program directly in the interactions pane" to the console pane.

We can interact with the functions  
in our program directly in the  
interactions pane

DrScala File Edit Tools Project Help

DrScala: /Users/ericeallen/tmp/IncomeTax.scala

New JUnit Test Case...

IncomeTax {

```
tofff0 = 0
bracket0 = 0

bracket1 = 100
tofff1 = 9075

bracket2 = 150
tofff2 = 36900

val bracket3 = 250
val cutoff3 = 89350
```

Create a new JUnit TestCase class

Interactions Console Compiler Output Test Output

Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
TESTING Nothing  
>

Editing /Users/ericeallen/tmp/IncomeTax.scala 2:3

DrScala File Edit Tools Project Help

DrScala: /Users/ericeallen/tmp/IncomeTax.scala

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

```
IncomeTax.scala
object IncomeTax {

  val cutoff0 = 0
  val bracket0 = 0

  val bracket1 = 100
  val cutoff1 = 9075

  val bracket2 = 15
  val cutoff2 = 369

  val bracket3 = 25
  val cutoff3 = 89350
}
```

We are prompted for a name. Let's call it IncomeTaxTest

New JUnit Test Case

Please enter a name for the test class:

IncomeTaxTest

Cancel OK

Interactions Console Compiler Output Test Output

Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
TESTING Nothing  
>

Editing /Users/ericeallen/tmp/IncomeTax.scala

2:3

DrScala File Edit Tools Project Help Wed 3:54 PM

DrScala: (Untitled)\*

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

(Untitled)\*  
IncomeTax.scala

```
import junit.framework.TestCase
import junit.framework.Assert._

/**
 * A JUnit test case class.
 * Every method starting with the word "test" will be called when ru
 * the test with JUnit.
 */
class IncomeTaxTest(name: String) extends TestCase(name) {

  /**
   * A test method.
   * (Replace "X" with a name describing the test. You may write as
```

A new test class is created

Interactions Console Compiler Output Test Output

Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
TESTING Nothing  
>

Editing (Untitled)\* 26:0

DrScala File Edit Tools Project Help

DrScala: (Untitled)\*

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

(Untitled)\*  
IncomeTax.scala

```
import junit.framework.TestCase
import junit.framework.Assert._

/**
 * A JUnit test case class.
 * Every method starting with the word "test" will be called when ru
 * the test with JUnit.
 */
class IncomeTaxTest(name: String) extends TestCase(name) {

  /**
   * A test method.
   * (Replace "X" with a name describing the test. You may write as
```

Note that this is *not* a case class

Interactions Console Compiler Output Test Output

Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
TESTING Nothing  
>

Editing (Untitled)\*

26:0

DrScala File Edit Tools Project Help

DrScala: (Untitled)\*

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test


(Untitled)\*  
IncomeTax.scala

```
import junit.framework.TestCase
import junit.framework.Assert._

/**
 * A JUnit test case class.
 * Every method starting with the word "test" will be called when ru
 * the test with JUnit.
 */
class IncomeTaxTest(name: String) extends TestCase(name) {

  /**
   * A test method.
   * (Replace "X" with a name describing the test. You may write as
```

Ignore the extends clause for now



Interactions Console Compiler Output Test Output

Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
TESTING Nothing  
>

Editing (Untitled)\* 26:0

DrScala File Edit Tools Project Help

DrScala: (Untitled)\*

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

(Untitled)\*  
IncomeTax.scala

```
import junit.framework.TestCase
import junit.framework.Assert._

/**
 * A JUnit test case class.
 * Every method starting with the word "test" will be called when ru
 * the test with JUnit.
 */
class IncomeTaxTest(name: String) extends TestCase(name) {

  /**
   * A test method.
   * (Replace "X" with a name describing the test. You may write as
```

**Ignore the import statements for now**

Interactions Console Compiler Output Test Output

Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
TESTING Nothing  
>

Editing (Untitled)\*

Wed 3:54 PM

26:0

DrScala File Edit Tools Project Help

DrScala: (Untitled)\*

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

(Untitled)\*  
IncomeTax.scala

```
* many "testSomething" methods in this class as you wish, and eac
* one will be called when running JUnit over this class.)
*/
def testX() {
}

/** Sample test method which tests no program code. */
def testNothing() {
  assertTrue("Dummy Test", true)
  println("TESTING Nothing")
}
}
```

The provided tests don't do very much

Interactions Console Compiler Output Test Output

Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
TESTING Nothing  
>

Editing (Untitled)\*

26:0



The screenshot shows the DrScala IDE interface. The top menu bar includes 'DrScala', 'File', 'Edit', 'Tools', 'Project', and 'Help'. The toolbar contains icons for 'New', 'Open', 'Save', 'Close', 'Cut', 'Copy', 'Paste', 'Undo', 'Redo', 'Find', 'Compile', 'Reset', 'Run', and 'Test'. The main editor window displays Scala code for a class named 'IncomeTax.scala'. The code includes a comment about 'testSomething' methods, a 'testX()' method definition, and a 'testNothing()' method definition. A red arrow points from the text 'All functions with names starting with "test" are treated as tests' to the 'testX()' method definition. The bottom panel shows the 'Test Output' tab with the message 'Welcome to DrScala. Working directory is /Users/ericeallen/tmp' and the output 'TESTING Nothing'.

```
DrScala: (Untitled) *
File Edit Tools Project Help
New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test
IncomeTax.scala
*/
def testX()
}

/** Sample test method which tests no program code. */
def testNothing() {
  assertTrue("Dummy Test", true)
  println("TESTING Nothing")
}
}

All functions with names starting
with "test" are treated as tests

Welcome to DrScala. Working directory is /Users/ericeallen/tmp
TESTING Nothing
>
```

The screenshot shows the DrScala IDE interface. The top menu bar includes 'DrScala', 'File', 'Edit', 'Tools', 'Project', and 'Help'. The toolbar contains icons for 'New', 'Open', 'Save', 'Close', 'Cut', 'Copy', 'Paste', 'Undo', 'Redo', 'Find', 'Compile', 'Reset', 'Run', and 'Test'. The main editor window displays the following Scala code:

```
* many "testSomething" methods in this class as you wish, and eac
* one will be called when running JUnit over this class.)
*/
def testX() {
}

/** Sample test method which tests no program code. */
def testNothing() {
  assertTrue("Dummy Test", true)
  println("TESTING Nothing")
}
}
```

A red arrow points from the text 'The assertTrue function is available to us in our tests.' to the `assertTrue` call in the `testNothing` method.

The bottom panel shows the 'Test Output' tab with the following text:

```
Welcome to DrScala. Working directory is /Users/ericeallen/tmp
TESTING Nothing
>
```

The status bar at the bottom left indicates 'Editing (Untitled) \*' and the bottom right shows the time '26:0'.

The assertTrue function is available to us in our tests.

DrScala File Edit Tools Project Help

DrScala: (Untitled)\*

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

(Untitled)\*  
IncomeTax.scala

```
* many "testSomething" methods in this class as you wish, and eac
* one will be called when running JUnit over this class.)
*/
def testX() {
}

/** Sample test method which tests no program code. */
def testNothing() {
  assertTrue("Dummy Test", true)
  println("TESTING Nothing")
}
}
```

The optional String is printed if the test fails.

Interactions Console Compiler Output Test Output

Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
TESTING Nothing  
>

Editing (Untitled)\*

Wed 3:54 PM

26:0

DrScala File Edit Tools Project Help

DrScala: (Untitled) \*

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

(Untitled)\*  
IncomeTax.scala

```
* many "testSomething" methods in this class as you wish, and eac
* one will be called when running JUnit over this class.)
*/
def testX() {
}

/** Sample test method which tests no program code. */
def testNothing() {
  assertTrue("Dummy Test", true)
  println("TESTING Nothing")
}
}
```

The test fails if this argument does not reduce to true.

Interactions Console Compiler Output Test Output

Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
TESTING Nothing  
>

Editing (Untitled) \* 26:0

# assertEquals fails if its two arguments are not equal

```
/**
 * Testing simple income tax computations.
 */
def testIncomeTax() {
  assertEquals(100, IncomeTax.incomeTax(1000))
  assertEquals(907, IncomeTax.incomeTax(9075))
  assertEquals(907 + 138, IncomeTax.incomeTax(10000))
}

/** Sample test method which tests no program code. */
def testNothing() {
  assertTrue("Dummy Test", true)
  println("TESTING Nothing")
}
```

Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
TESTING Nothing  
> |

Bracket matches: def testIncomeTax() {

# Hitting the Test button prompts us to compile

The screenshot shows the DrScala IDE interface. The main editor displays the following Scala code:

```
class IncomeTaxTest(name: String) extends TestCase(name) {  
  
  /**  
   * Testing simple income tax computations.  
   */  
  def testInc  
  assertEquals  
  assertEquals  
  assertEquals  
  
}
```

A dialog box titled "Must Compile All Source Files to Run Unit Tests" is overlaid on the code. It contains the following text:

Before you can run unit tests, you must first compile all out of sync source files. The files below are out of sync. Would you like to compile all files and run the specified test(s)?

IncomeTaxTest.scala

Yes No

The IDE's console at the bottom shows the following output:

```
Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
>
```

The status bar at the bottom left indicates "Running All Open Unit Tests" and the bottom right shows the time "14:19".

# Agreeing to compile prompts us to save

The screenshot shows the DrScala IDE interface. The main editor window displays Scala code for a test class:

```
class IncomeTaxTest(name: String) extends TestCase(name) {  
  
  /**  
   * Testing simple income tax computations.  
   */  
  def testIncomeTax() {  
    assertEquals(  
    assertEquals(  
    assertEquals(  
  
  }  
}
```

A dialog box titled "Must Save All Files to Continue" is overlaid on the code. It contains the following text:

To compile, you must first save ALL modified files.  
Would you like to save and then compile?

Always save before compiling

No Yes

The console at the bottom of the IDE shows the following output:

```
Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
>
```

# A green bar indicates that all tests passed

The screenshot shows the DrScala IDE interface. The main editor displays the following Scala code:

```
class IncomeTaxTest(name: String) extends TestCase(name) {  
  
  /**  
   * Testing simple income tax computations.  
   */  
  def testIncomeTax() {  
    assertEquals(100, IncomeTax.incomeTax(1000))  
    assertEquals(907, IncomeTax.incomeTax(9075))  
    assertEquals(907 + 138, IncomeTax.incomeTax(10000))  
  }  
}
```

Below the code editor, the Test Output panel is visible, showing the following text:

```
All tests completed successfully.  
IncomeTaxTest  
  testNothing  
  testIncomeTax
```

On the right side of the Test Output panel, there is a 'Test Progress' section with a green progress bar that is completely filled, indicating that all tests passed. A red arrow points from the title text to this green bar.

At the bottom of the IDE, the status bar shows 'Editing /Users/ericallen/tmp/IncomeTaxTest.scala' and the time '14:19'.



DrScala File Edit Tools Project Help Wed 8:15 PM

DrScala: /Users/ericeallen/tmp/IncomeTaxTest.scala

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

IncomeTax.scala  
IncomeTaxTest.scala

```
def testIncomeTax() {
  assertEquals(100, IncomeTax.incomeTax(1000))
  assertEquals(907, IncomeTax.incomeTax(9075))
  assertEquals(907 + 138, IncomeTax.incomeTax(10000))
}

def testThatFails() {
  assertTrue(false)
}

/** Sample test method which tests no program code.
def testNothing() {
  assertTrue("Dummy Test", true)
  println("TESTING Nothing")
}
```

A red bar indicates a test failure

Interactions Console Compiler Output Test Output

IncomeTaxTest  
testIncomeTax  
testThatFails  
testNothing

Test Progress  
Show Stack Trace  
 Highlight source

Editing /Users/ericeallen/tmp/IncomeTaxTest.scala 21:0

DrScala File Edit Tools Project Help Wed 8:15 PM

DrScala: /Users/ericeallen/tmp/IncomeTaxTest.scala

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

IncomeTax.scala  
IncomeTaxTest.scala

```
def testIncomeTax() {
  assertEquals(100, IncomeTax.incomeTax(1000))
  assertEquals(907, IncomeTax.incomeTax(9075))
  assertEquals(907 + 138, IncomeTax.incomeTax(10000))
}

def testThatFails() {
  assertTrue(false)
}

/** Sample test method which tests no program code.
def testNothing() {
  assertTrue("Dummy Test", true)
  println("TESTING Nothing")
}
```

The failing test is highlighted in yellow

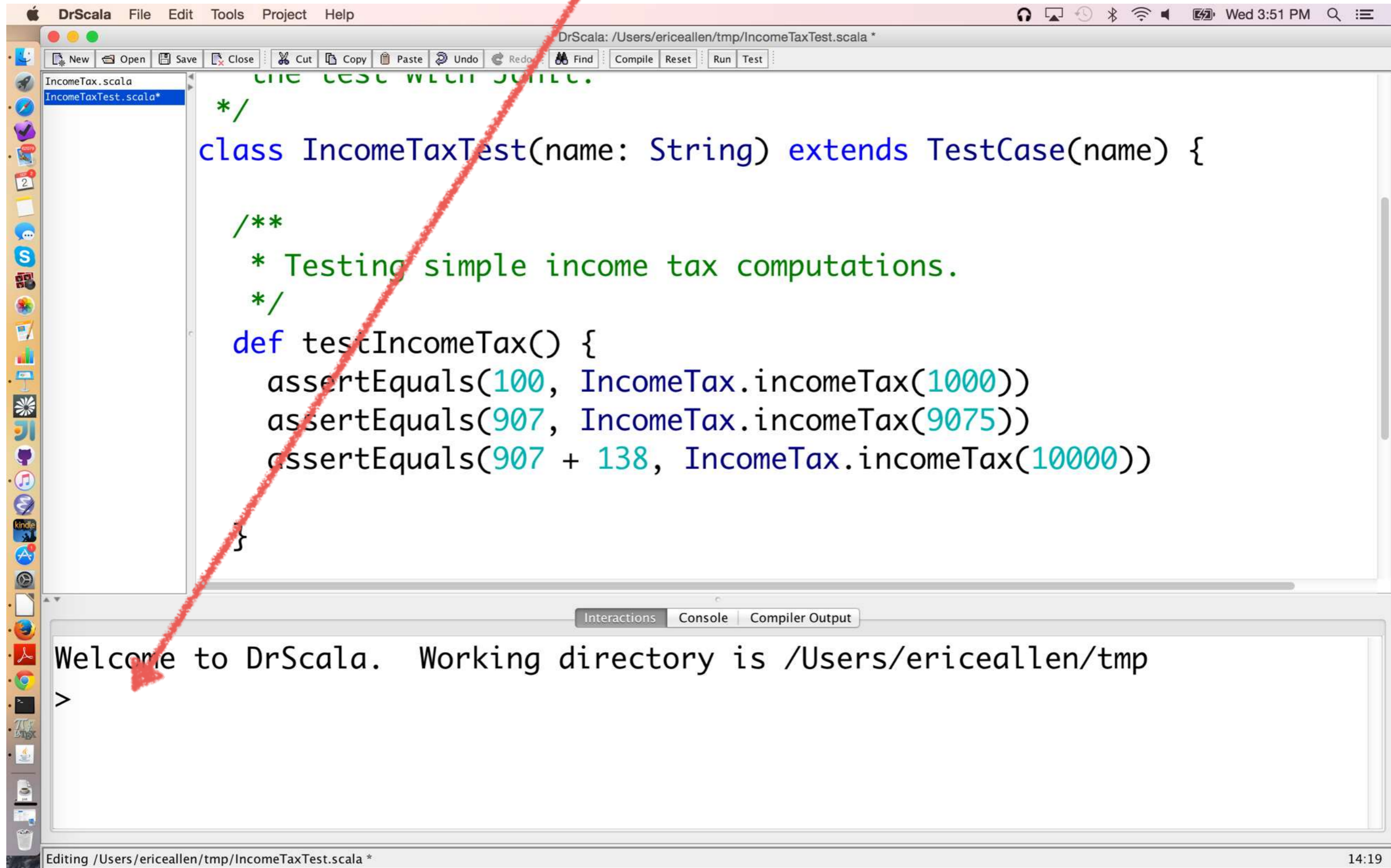
Interactions Console Compiler Output Test Output

IncomeTaxTest  
testIncomeTax  
testThatFails  
testNothing

Test Progress  
Show Stack Trace  
 Highlight source

Editing /Users/ericeallen/tmp/IncomeTaxTest.scala 21:0

# To interact with our program, we use the Interactions Pane



The screenshot shows the DrScala IDE interface. The main editor window displays the following Scala code:

```
class IncomeTaxTest(name: String) extends TestCase(name) {  
  
  /**  
   * Testing simple income tax computations.  
   */  
  def testIncomeTax() {  
    assertEquals(100, IncomeTax.incomeTax(1000))  
    assertEquals(907, IncomeTax.incomeTax(9075))  
    assertEquals(907 + 138, IncomeTax.incomeTax(10000))  
  }  
}
```

Below the editor, the Interactions pane is active, showing the following output:

```
Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
>
```

A red arrow points from the top of the slide to the Interactions pane, indicating that this is where the user interacts with the program.

DrScala File Edit Tools Project Help

DrScala: /Users/ericeallen/tmp/IncomeTax.scala

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

(Untitled)\*  
IncomeTax.scala  
IncomeTaxTest.scala\*

```
/* Given an income in U.S. Dollars,  
 * returns the dollar value of tax  
 * owed for a single tax payer, using  
 * 2014-2015 IRS tax brackets.  
 */  
def incomeTax(income: Int): Int = {  
  require(income >= 0)  
  
  if (income <= cutoff0) {  
    bracket0
```

Interactions Console Compiler Output Test Output

```
Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
> 2 + 2  
res0: Int = 4  
> |
```

We can enter arbitrary Scala expressions

Editing /Users/ericeallen/tmp/IncomeTax.scala 2:3

DrScala File Edit Tools Project Help

DrScala: /Users/ericeallen/tmp/IncomeTax.scala

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

(Untitled)\*  
IncomeTax.scala  
IncomeTaxTest.scala\*

```
/* Given an income in U.S. Dollars,  
 * returns the dollar value of tax  
 * owed for a single tax payer, using  
 * 2014-2015 IRS tax brackets.  
 */  
def incomeTax(income: Int): Int = {  
  require(income >= 0)  
  
  if (income <= cutoff0) {  
    bracket0
```

Interactions Console Compiler Output Test Output

```
Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
> 2 + 2  
res0: Int = 4  
> |
```

The value our expression reduces to is displayed

Editing /Users/ericeallen/tmp/IncomeTax.scala 2:3

(Untitled)\*  
IncomeTax.scala  
IncomeTaxTest.scala\*

```
/* Given an income in U.S. Dollars,  
 * returns the dollar value of tax  
 * owed for a single tax payer, using  
 * 2014-2015 IRS tax brackets.  
 */  
def incomeTax(income: Int): Int = {  
  require(income >= 0)  
  
  if (income <= cutoff0) {  
    bracket0
```

```
Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
> 2 + 2  
res0: Int = 4  
> |
```

As is its type

DrScala File Edit Tools Project Help

DrScala: /Users/ericeallen/tmp/IncomeTax.scala

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

(Untitled)\*  
IncomeTax.scala  
IncomeTaxTest.scala\*

```
/* Given an income in U.S. Dollars,  
 * returns the dollar value of tax  
 * owed for a single tax payer, using  
 * 2014-2015 IRS tax brackets.  
 */  
def incomeTax(income: Int): Int = {  
  require(income >= 0)  
  
  if (income <= cutoff0) {  
    bracket0
```

Interactions Console Compiler Output Test Output

```
Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
> 2 + 2  
res0: Int = 4  
> |
```

And the value is bound to a fresh identifier

Editing /Users/ericeallen/tmp/IncomeTax.scala 2:3

DrScala File Edit Tools Project Help Wed 5:03 PM

DrScala: /Users/ericeallen/tmp/IncomeTax.scala

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

(Untitled)\*  
IncomeTax.scala  
IncomeTaxTest.scala\*

```
/* Given an income in U.S. Dollars,  
 * returns the dollar value of tax  
 * owed for a single tax payer, using  
 * 2014-2015 IRS tax brackets.  
 */  
def incomeTax(income: Int): Int = {  
  require(income >= 0)  
  
  if (income <= cutoff0) {  
    bracket0
```

Interactions Console Compiler Output Test Output

```
Welcome to DrScala. Working directory is /Users/ericeallen/tmp  
> 2 + 2  
res0: Int = 4  
> IncomeTax.incomeTax(100000)  
res1: Int = 21174  
>
```

The classes we have compiled in  
Definitions are in scope in Interactions

Editing /Users/ericeallen/tmp/IncomeTax.scala 2:3



The screenshot shows the DrScala IDE interface. The top menu bar includes 'DrScala', 'File', 'Edit', 'Tools', 'Project', and 'Help'. The title bar indicates the current file is 'IncomeTax.scala' located at '/Users/ericeallen/tmp/IncomeTax.scala'. The main editor window contains the following Scala code:

```
/* Given an income in U.S. Dollars,
 * returns the dollar value of tax
 * owed for a single tax payer, using
 * 2014-2015 IRS tax brackets.
 */
def incomeTax(income: Int): Int = {
  require(income >= 0)

  if (income <= cutoff0) {
    bracket0
  }
}
```

Below the editor is a REPL window with tabs for 'Interactions', 'Console', 'Compiler Output', and 'Test Output'. The REPL shows the following interactions:

```
Welcome to DrScala. Working directory is /Users/ericeallen/tmp
> 2 + 2
res0: Int = 4
> IncomeTax.incomeTax(100000)
res1: Int = 21174
> res0 * res1
res2: Int = 84696
>
```

A red arrow points from the text 'We can refer to previously bound identifiers in subsequent expressions' to the 'res1' value in the REPL output.

We can refer to previously bound identifiers in subsequent expressions

DrScala File Edit Tools Project Help Wed 5:04 PM

DrScala: /Users/ericeallen/tmp/IncomeTax.scala

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

(Untitled)\*  
IncomeTax.scala  
IncomeTaxTest.scala\*

```
/* Given an income in U.S. Dollars,  
 * returns the dollar value of tax  
 * owed for a single tax payer, using  
 * 2014-2015 IRS tax brackets.  
 */  
def incomeTax(income: Int): Int = {  
  require(income >= 0)  
  
  if (income <= cutoff0) {  
    bracket0
```

Interactions Console Compiler Output Test Output

```
res0: Int = 4  
> IncomeTax.incomeTax(100000)  
res1: Int = 21174  
> res0 * res1  
res2: Int = 84696  
> val pi = 3.14  
pi: Double = 3.14  
>
```

We can also bind new identifiers directly

Editing /Users/ericeallen/tmp/IncomeTax.scala 2:3

DrScala File Edit Tools Project Help

DrScala: /Users/ericeallen/tmp/IncomeTax.scala

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

(Untitled)\*  
IncomeTax.scala  
IncomeTaxTest.scala\*

```
/* Given an income in U.S. Dollars,  
 * returns the dollar value of tax  
 * owed for a single tax payer, using  
 * 2014-2015 IRS tax brackets.  
 */  
def incomeTax(income: Int): Int = {  
  require(income >= 0)  
  
  if (income <= cutoff0) {  
    bracket0
```

Interactions Console Compiler Output Test Output

```
res1: Int = 21174  
> res0 * res1  
res2: Int = 84696  
> val pi = 3.14  
pi: Double = 3.14  
> res1 * pi  
res3: Double = 66486.36  
>
```

And compute with them

Editing /Users/ericeallen/tmp/IncomeTax.scala 2:3

DrScala File Edit Tools Project Help

DrScala: /Users/ericeallen/tmp/IncomeTax.scala

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test


(Untitled)\*  
IncomeTax.scala  
IncomeTaxTest.scala\*

```
/* Given an income in U.S. Dollars,  
 * returns the dollar value of tax  
 * owed for a single tax payer, using  
 * 2014-2015 IRS tax brackets.  
 */  
def incomeTax(income: Int): Int = {  
  require(income >= 0)  
  
  if (income <= cutoff0) {  
    bracket0
```

Interactions Console Compiler Output Test Output

```
res2: Int = 84696  
> val pi = 3.14  
pi: Double = 3.14  
> res1 * pi  
res3: Double = 66486.36  
> def square(x: Double) = x * x  
square: (x: Double)Double  
>
```

We can also define new functions



Editing /Users/ericeallen/tmp/IncomeTax.scala 2:3

- (Untitled)\*
- IncomeTax.scala
- IncomeTaxTest.scala\*

```

* Given an income in U.S. Dollars,
* returns the dollar value of tax
* owed for a single tax payer, using
* 2014-2015 IRS tax brackets.
*/
def incomeTax(income: Int): Int = {
  require(income >= 0)

  if (income <= cutoff0) {
    bracket0
  }
}

```

```

> val pi = 3.14
pi: Double = 3.14
> res1 * pi
res3: Double = 66486.36
> def square(x: Double) = x * x
square: (x: Double)Double
> def abs(x: Double) =
  |

```

For definitions that are not syntactically complete, we are given a new line, indicated by a vertical bar



DrScala File Edit Tools Project Help Wed 5:07 PM

DrScala: /Users/ericeallen/tmp/IncomeTax.scala

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

(Untitled)\*  
IncomeTax.scala  
IncomeTaxTest.scala\*

```
/**  
 * Given an income in U.S. Dollars,  
 * returns the dollar value of tax  
 * owed for a single tax payer, using  
 * 2014-2015 IRS tax brackets.  
 */  
def incomeTax(income: Int): Int = {  
  require(income >= 0)  
  
  if (income <= cutoff0) {  
    bracket0
```

Interactions Console Compiler Output Test Output

Welcome to DrScala. Working directory is /Users/ericeallen/tmp

```
> def abs(x: Double) =  
  | if (x < 0) -x else  
  | x  
abs: (x: Double)Double
```

>

Resetting Interactions

2:3

The function is bound and an arrow type is displayed

DrScala File Edit Tools Project Help

DrScala: /Users/ericeallen/tmp/IncomeTax.scala

New Open Save Close Cut Copy Paste Undo Redo Find Compile Reset Run Test

```
/**
 * Given an income in U.S. Dollars,
 * returns the dollar value of tax
 * owed for a single tax payer, using
 * 2014-2015 IRS tax brackets.
 */
def incomeTax(income: Int): Int = {
  require(income >= 0)

  if (income <= cutoff0) {
    bracket0
```

Interactions Console Compiler Output Test Output

```
Welcome to DrScala. Working directory is /Users/ericeallen/tmp
> def abs(x: Double) =
  | if (x < 0) -x else
  | x
abs: (x: Double)Double
> abs(-5.0)
res0: Double = 5.0
>
```

Resetting Interactions

2:3

And we can refer to this function in subsequent expressions

# We can click on the file to appear in Definitions

The screenshot shows the DrScala IDE interface. The title bar reads "DrScala: /Users/ericeallen/tmp/IncomeTax.scala". The menu bar includes "DrScala", "File", "Edit", "Tools", "Project", and "Help". The toolbar contains "New", "Open", "Save", "Close", "Cut", "Copy", "Paste", "Undo", "Redo", "Find", "Compile", "Reset", "Run", and "Test".

The file explorer on the left shows a list of files: "(Untitled)\*", "IncomeTax.scala", and "IncomeTaxTest.scala\*". A red arrow points from the title of the slide to the "IncomeTax.scala" file in the explorer.

The main editor displays the following Scala code:

```
// Brackets are in tenths of percentage points,  
// thus percentage divisor must be 1000  
val divisor = 1000  
  
/**  
 * Given an income in U.S. Dollars,  
 * returns the dollar value of tax  
 * owed for a single tax payer, using  
 * 2014-2015 IRS tax brackets.  
 */  
def incomeTax(income: Int): Int = {  
  require(income >= 0)  
}
```

At the bottom, the console shows the message: "Welcome to DrScala. Working directory is /Users/ericeallen/tmp" followed by a prompt ">".

The status bar at the bottom left indicates "Editing /Users/ericeallen/tmp/IncomeTax.scala" and the bottom right shows "2:3".



# Files that have not been saved include an asterisk

The screenshot shows the DrScala IDE interface. The title bar reads "DrScala: /Users/ericeallen/tmp/IncomeTaxTest.scala \*". The menu bar includes "DrScala", "File", "Edit", "Tools", "Project", and "Help". The toolbar contains "New", "Open", "Save", "Close", "Cut", "Copy", "Paste", "Undo", "Redo", "Find", "Compile", "Reset", "Run", and "Test".

The file explorer on the left shows a list of files: "(Untitled)\*", "IncomeTax.scala", and "IncomeTaxTest.scala\*", with the latter selected. A red arrow points from the asterisk in the title bar to the asterisk in the filename.

The code editor contains the following Scala code:

```
* Testing simple income tax computations.
*/
def testIncomeTax() {
  assertEquals(100, IncomeTax.incomeTax(1000))
  assertEquals(907, IncomeTax.incomeTax(9075))
  assertEquals(907 + 138, IncomeTax.incomeTax(10000))
}

/** Sample test method which tests no program code.
def testNothing() {
  assertTrue("Dummy Test", true)
  println("TESTING Nothing")
}
```

Welcome to DrScala. Working directory is /Users/ericeallen/tmp

>

# Reset resets the Interactions session

The screenshot shows the DrScala IDE interface. The top menu bar includes 'DrScala', 'File', 'Edit', 'Tools', 'Project', and 'Help'. The toolbar contains buttons for 'New', 'Open', 'Save', 'Close', 'Cut', 'Copy', 'Paste', 'Undo', 'Redo', 'Find', 'Compile', 'Reset', 'Run', and 'Test'. A red arrow points to the 'Reset' button. The main editor window displays the following Scala code:

```
(Untitled)*
IncomeTax.scala
IncomeTaxTest.scala*

/* Testing simple income tax computations.
 */
def testIncomeTax() {
  assertEquals(100, IncomeTax.incomeTax(1000))
  assertEquals(907, IncomeTax.incomeTax(9075))
  assertEquals(907 + 138, IncomeTax.incomeTax(10000))
}

/** Sample test method which tests no program code.
 */
def testNothing() {
  assertTrue("Dummy Test", true)
  println("TESTING Nothing")
}
```

Below the editor, the 'Interactions' console shows the following text:

```
Welcome to DrScala. Working directory is /Users/ericeallen/tmp
>
```

The status bar at the bottom indicates 'Editing /Users/ericeallen/tmp/IncomeTaxTest.scala \*' and the time '18:9'.

# Run executes Definitions

The screenshot shows the DrScala IDE interface. The top menu bar includes 'DrScala', 'File', 'Edit', 'Tools', 'Project', and 'Help'. The title bar indicates the current file is 'DrScala: /Users/ericeallen/tmp/IncomeTaxTest.scala \*'. The toolbar contains buttons for 'New', 'Open', 'Save', 'Close', 'Cut', 'Copy', 'Paste', 'Undo', 'Redo', 'Find', 'Compile', 'Reset', 'Run', and 'Test'. A red arrow points to the 'Run' button.

The editor window displays the following Scala code:

```
(Untitled)*
IncomeTax.scala
IncomeTaxTest.scala*

/* Testing simple income tax computations.
 */
def testIncomeTax() {
  assertEquals(100, IncomeTax.incomeTax(1000))
  assertEquals(907, IncomeTax.incomeTax(9075))
  assertEquals(907 + 138, IncomeTax.incomeTax(10000))
}

/** Sample test method which tests no program code.
 */
def testNothing() {
  assertTrue("Dummy Test", true)
  println("TESTING Nothing")
}
```

The bottom panel shows the 'Test Output' tab with the following text:

```
Welcome to DrScala. Working directory is /Users/ericeallen/tmp
>
```

The status bar at the bottom indicates 'Editing /Users/ericeallen/tmp/IncomeTaxTest.scala \*' and the time '18:9'.