














- Course
-  CS-A1110
-  Course materials
-  Your points
-  Form a group
-  Code Vault 
-  Lab Queue 
-  Telegram chat 
-  Lab sessions
- Glossary
- Scala reference
-  O1Library docs 
- FAQ
- IntelliJ installation
- Learning goals
- Style guide
- Debugger
- Resources
- For the reader

This course has already ended.  
The latest instance of the course can be found at: **O1: 2023**

Luet oppimateriaalin englanninkielistä versiota. Mainitsit kuitenkin taustakyselyssä osaavasi suomea. Siksi **suosittelemme, että käytät suomenkielistä versiota**, joka on testatumpi ja hieman laajempi ja muutenkin mukava.

Suomenkielinen materiaali kyllä esittelee englanninkielisetkin termit. Myös suomenkielisessä materiaalissa käytetään ohjelmien koodissa englanninkielisiä nimiä kurssin alkupään johdantoesimerkkejä lukuunottamatta.

Voit vaihtaa kieltä A+:n valikon yläreunassa olevasta painikkeesta. Tai tästä: **Vaihda suomeksi**.

## For the Reader

This interactive ebook combines text with diverse practice activities and dynamic illustrations. The ebook has been tailored to the needs of the Aalto University course Programming 1 a.k.a. O1, but we’ve made it publicly available for personal use.

The ebook is divided into **weeks**, which set a tempo for Aalto students: each week has a deadline for submitting assignments.

Each week is further divided in chapters. Each chapter builds on earlier ones; the ebook is meant to be read in order, without skipping chapters. Of course, you may occasionally want to go back to an earlier chapter to refresh your memory, and there are a lot of links between the chapters to help you do that.

Chapter 1.1 serves as an introduction to the course.

The ebook comes with a lot of example programs. Some of them are embedded directly in the ebook’s text; others are packaged into downloadable files, which you’ll load into a separate programming environment.

The main chapters of the ebook are not designed for use as a reference and aren’t optimal for that purpose. When you want to look something up, take a look at the [Scala Reference page](#) or the [Glossary](#), or [do a word search within the ebook](#).

## Technical Requirements

Chapter 1.2 will introduce you to the environment that we use. Read this ebook on a computer where you can access the programming environment.

The ebook isn’t designed for use on mobile devices. Not that it’s forbidden, but you won’t be able to install all the necessary tools on a mobile device. Some parts of the ebook may also work imperfectly on such devices.

The ebook will work right only if you have an up-to-date web browser and JavaScript is enabled.

The ebook makes occasional use of sound. Obviously, this can work only if your computer setup has audio support switched on. If you study among other people, see if you can pack some headphones with you. On the other hand, we use sound only infrequently, and none of the chapters is impossible without sound.

## Notations Used in the Ebook

Gray-bordered boxes

Gray-bordered boxes like this contain optional material. Skipping these boxes won’t prevent you from learning the basics of programming; neither will it put your course grade in danger. Some of them do, however, contain additional tasks that you may find to be an effective form of practice. Others have links to further reading, and still others present you with challenge tasks that may take you well beyond official course requirements.

We recommend that you take a look at these boxes, too. But if you must skip *something*, skip them.

The ebook embeds quotations from students who have taken the course before — and even students who are taking the course right now. In the English version of the ebook, most of these have been translated from Finnish. Here’s an example:

It’s great to feel like you understand something after such a long and confused struggle! I was so hyped up that I submitted the assignment twice even though I already scored full points with the first try. :)

Boxes like the one below appear all over the ebook. They contain [program code](#). Many of these boxes use color to highlight the different parts the code consists of.

```
def withdraw(sum: Int) = {  
  val withdrawn = min(sum, this.balance)  
  this.balance = this.balance - withdrawn  
  withdrawn  
}
```

Chapter 1.3 introduces an interactive programming environment known as the [REPL](#); boxes like the one below contain examples of interactions in that environment. The text written by the programmer appears on a darker background and the automatic response from the REPL on a lighter one.

```
1 + 1res0: Int = 2  
50 + 50 > 100res1: Boolean = false
```

Many examples come with green boxes like the two below; they contain explanatory text. If you mouse over the explanations, the relevant part of the above example will be highlighted. You can also click the boxes to make the highlight stay on.

These are inputs from the programmer.

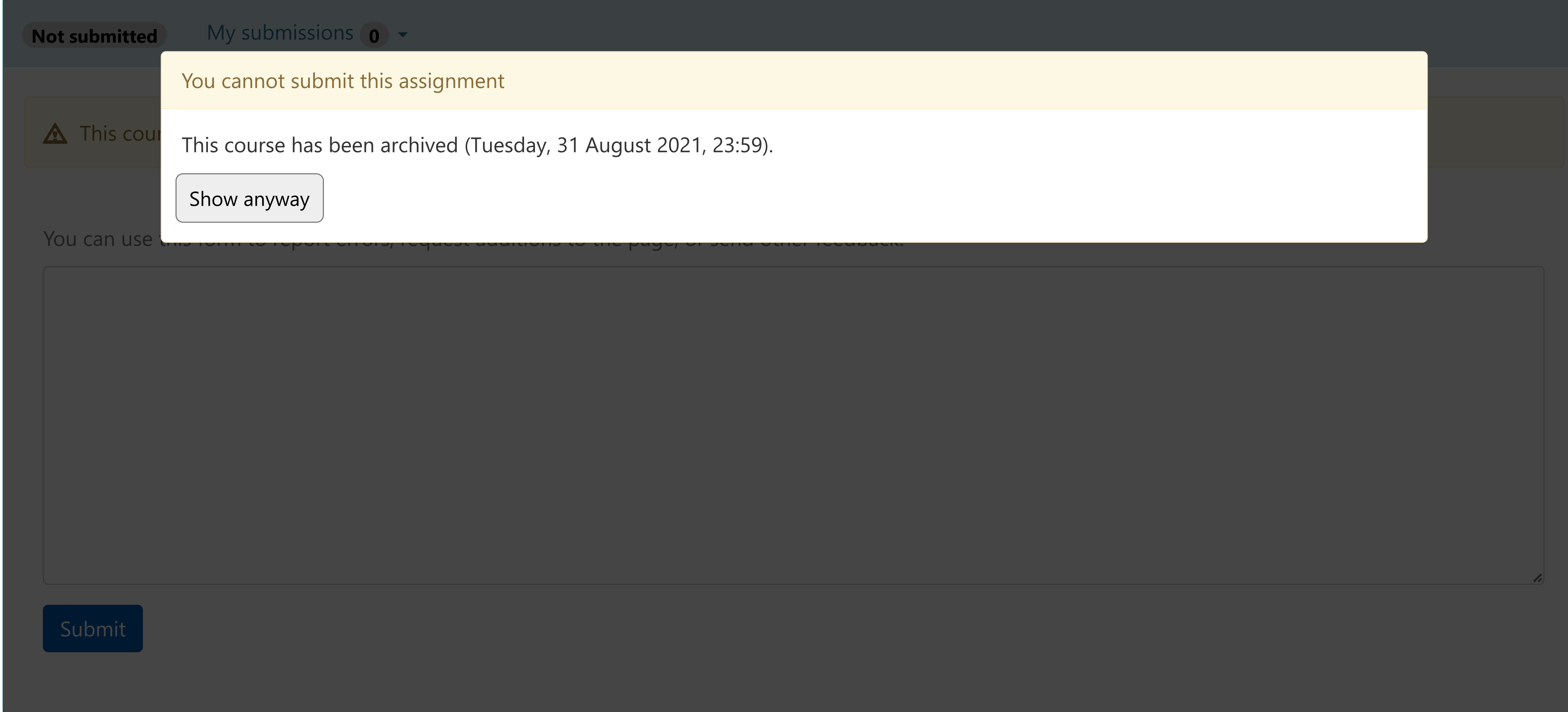
These are values automatically computed as a response to the inputs.

Peachy boxes contain [pseudocode](#): text that resembles actual program code but is meant for human readers only. You’ll see these boxes from Chapter 2.5 onwards.

```
def totalPrice = {  
  For each of the auctions in this.items in turn:  
    - Determine the current price of the item and add it to the sum.  
  Finally, return the sum.  
}
```

The ebook also contains many dynamic elements, such as multiple-choice questions and interactive animations that you the reader can control. These elements are either self-explanatory or explained where they appear.

## Feedback



## Credits

Thousands of students have given feedback that has contributed to this ebook’s design. Thank you!

The ebook’s chapters, programming assignments, and weekly bulletins have been written in Finnish and translated into English by Juha Sorva.

The appendices ([glossary](#), [Scala reference](#), [FAQ](#), etc.) are by Juha Sorva unless otherwise specified on the page.

The automatic assessment of the assignments has been developed by: (in alphabetical order) Riku Autio, Nikolas Drosdek, Joonatan Honkamaa, Jaakko Kantojärvi, Niklas Kröger, Teemu Lehtinen, Strasdosky Otewa, Timi Seppälä, Teemu Sirkiä, and Aleksi Vartiainen.

The illustrations at the top of each chapter, and the similar drawings elsewhere in the ebook, are the work of Christina Lassheikki.

The animations that detail the execution Scala programs have been designed by Juha Sorva and Teemu Sirkiä. Teemu Sirkiä and Riku Autio did the technical implementation, relying on Teemu’s [Jsvee](#) and [Kelmu](#) toolkits.

The other diagrams and interactive presentations in the ebook are by Juha Sorva.

The [O1Library](#) software has been developed by Aleksi Lukkarinen and Juha Sorva. Several of its key components are built upon Aleksi’s [SMCL](#) library.

The pedagogy of using O1Library for simple graphical programming (such as [Pic](#)) is inspired by the textbooks *How to Design Programs* by Flatt, Felleisen, Findler, and Krishnamurthi and *Picturing Programs* by Stephen Bloch.

The course platform A+ was originally created at Aalto’s [LeTech](#) research group as a student project. The open-source [project](#) is now shepherded by the Computer Science department’s [edu-tech team](#) and hosted by the department’s [IT services](#). Markku Riekkinen is the current lead developer; [dozens of Aalto students and others](#) have also contributed.

The [A+ Courses](#) plugin, which supports A+ and O1 in IntelliJ IDEA, is another open-source [project](#). It was created by Nikolai Denissov, Olli Kiljunen, and Nikolas Drosdek with input from Juha Sorva, Otto Seppälä, Arto Hellas, and others.

For O1’s current teaching staff, please see Chapter 1.1.

Additional credits appear at the ends of some chapters.

