

# Homework 6: Functional Programming with Racket

In this homework, you will complete two Racket programming exercises. The purpose of the exercises is to familiarize you with a LISP-like functional programming language, namely Racket. You will mainly work on the exercises in class, but if you're unable to finish them in class, you must complete them as homework.

## Step 1. Check out the project

You must first checkout the homework files (probably using Eclipse):

[https://utopia.cs.memphis.edu/course/comp4040-2013fall/uuids/YOUR\\_UUID/homework6/](https://utopia.cs.memphis.edu/course/comp4040-2013fall/uuids/YOUR_UUID/homework6/)

You should see a **part1** folder in the project (and possibly a **part2** folder).

## Step 2. "Quick" Activity

First, you must download and install Racket (<http://racket-lang.org/>). Then, fire up the **DrRacket** program. Finally, open the two files **part1**, **quick.rkt** and **use.rkt** in **DrRacket**.

Now, read through the Racket "Quick" introduction:

<http://docs.racket-lang.org/quick/>

As you read through it, execute each instruction given in **DrRacket**. Also, save each instruction in the script area of **DrRacket** for **quick.rkt**. Note the document will tell you when to add code to **use.rkt**.

In the script, use comments, such as

```
; p4
```

To separate the instructions by the page in the Quick document where the instructions were given. In the above example, the Racket expressions given on page 4 would immediately follow the comment in the script.

At the end, your script should run cleanly. If the Quick document would have you enter syntactically invalid code, then you should include it in your script, but comment it out. If the document would have you introduce a name conflict in your code, you may do some renaming to avoid the conflict.

## Step 3. To be announced

For this part of the homework, you will work through another Racket tutorial, this one on how build a web app in Racket:

<http://docs.racket-lang.org/continue/>

Before you start reading, do a **Team > Update** on your **homework6** project in Eclipse. You will see a **part2** folder has been added, and it contains a set of Racket source files: **sec01.rkt**, **sec04.rkt–sec07.rkt**.

As you work through the tutorial, the authors incrementally add functionality to a web app. Each of the Racket files in the part2 folder corresponds to a different version of the web app.

What to turn in: Fill in each Racket source file with the appropriate version of the web app (through Section 7). The code you submit must run properly. Note that this should not be difficult, as the complete source for each version is given at the end of each section. Given how easy this is, you should keep in mind that the real objective of this homework is to have you work through the tutorial—there will be an exam!

## **Step 4. Submit your work**

To submit, simply commit your files to the SVN repository. Again, you'll probably want to use Eclipse (**Team > Commit**).