

Homework 3: A Simple DVD Database

- Preliminaries
- Introduction
- Problems
 - Problem 1: Explanation
 - Problem 2: Verifying DVDs
 - Problem 3: Adding Information
 - Problem 4: Searching by Rating
 - Problem 5: Fixing Capitalization Bugs
- Extra Credit
- Turning it In

Preliminaries

Assigned: Friday, 16 February 2001

Due: Friday, 23 February 2001

No extensions without prior permission!

Note: This homework is optional. However, you must do either this homework or laboratory writeup 1. If you do both, you will receive extra credit.

Summary: In this assignment, you will understand and update an existing Scheme based CGI script so as to provide more capabilities.

Purpose: To enhance your skills with recursive procedures; to give you experience reading, thinking about, and modifying other people's code

Group Work: You may work in groups of size 2, 3, or 4. You may not work alone. You may not work in groups of size 5 or greater. You should feel free to discuss your work with anyone you deem appropriate. If you receive help from others outside your group, make sure to document that help.

Time: I try to keep the core of my assignments under eight hours (extra credit may, of course, take extra time). When a group reaches five hours (that is, five hours per person), that group should notify me. If the first groups to reach five hours don't feel close to completion, I will update the assignment (and find a way to compensate those groups for starting early).

Required Files:

- `webutils.ss` (don't copy, just look at).
- `dvds.ss` (make a copy)
- `dvdweb.ss` (make a copy)
- `dvdweb.cgi` (link only works in MathLAN; you can guess the contents)

Introduction

While much of your work in 151 involves building new procedures and solving new problems, much of the “real world” programming that goes on involves supporting and updating working code. In this homework, you’ll have the opportunity to extend a simple, but working, CGI-based sample database.

There are three central files to the database:

- `dvds.ss` is the primary code that defines the data in the database and the procedures for operating on those data;
- `dvdweb.ss` is the Scheme file that provides a Web interface to the database (you can also interact with the database directly in DrScheme, but few people would want to do that.
- `dvdweb.cgi` is the CGI file that loads `dvdweb.ss`.

Your first step should be to read through the three files (along with the library file they rely on), take some notes, and make sure you understand what’s going on. (You should also make copies of the files.)

Problems

Problem 1: Explanation

Once you copy and set up the files appropriately, you’ll find that you can load `dvdweb.cgi` directly, without a corresponding HTML page. **Explain why!**

Problem 2: Verifying DVDs

Right now, the `dvd?` procedure is not quite correct. In particular, consider the following

```
> (define notquite (list "A" "B" (list 1 2 3) "D"))
> (dvd? notquite)
#t
> (dvd->html notquite)
string-append: expects type <string> as 1st argument,
given: 2; other arguments were: ", " 3
```

Fix `dvd?` so that it only accepts valid DVDs.

Problem 3: Adding Information

Update `dvds.ss` so that DVDs now have the format:

```
(title studio rating evaluation genres notes)
```

where rating is one of "G", "PG", "R", and "U" (for unrated) and evaluation is an integer between 1 and 10.

For example

```
(define titanea
  (list "Titan A.E."
        "Fox"
        "PG"
        5
        (list "animated" "sf")
        ""))
```

Many parts of the code assume a particular format for DVDs, so be careful to update them all. Once you've made the update, test your updated code.

Problem 4: Searching by Rating

Update `dvdsweb.ss` so that it is also possible to search by rating.

Problem 5: Fixing Capitalization Bugs

As you've played with the program, you may have noted that it doesn't always seem to work right. You can find the problem in the Web interface and in DrScheme. For example,

```
> (find-dvds-by-studio "Disney" samsdvds)
(("Tim Burton's the Nightmare Before Christmas"
  "Disney"
  ("animated" "holiday")
  "Also includes early Tim Burton films and other cool stuff.")
 ("Tarzan"
  "Disney"
  ("animated" "childrens")
  "Collector's Edition - Two Discs"))
> (find-dvds-by-studio "disney" samsdvds)
()
```

Similarly, if you use the Web interface to search for genre ANIMATED you'll get no results, but if you search for genre animated, you'll get lots of results.

- Explain the difference between the two `find-dvds-by-studio` expressions.
- Fix the code so that it works correctly for both kinds of queries.

Extra Credit

Those of you with extra time, extra enthusiasm, or extra desire for a high grade may wish to do more on this assignment. While I generally think it's up to you to come up with extensions, here are a few simple possibilities:

- Make the pages more attractive.
- Allow people to search for DVDs with a minimum evaluation value (e.g., "DVDs rated 6 or higher").
- Allow people to search for multiple genres (e.g., "all the animated holiday movies").

- Allow people to search by multiple categories (e.g., "all the PG-rated movies from Disney")

Turning it In

Create an HTML page with the answers to all the questions (e.g., "Explain"). This page should also summarize any changes you made to my code (e.g., in all the problems but 1).

Upload that page and your modified .ss files to Blackboard.