

# Compound Module

[Help](#)

In this module you will also learn how to design compound data definitions to represent information that consists of two or more naturally connected values, as well as how to use such data as a world state in HtDW problems














NOTE: Several of the videos this module are quite long. That is because they work through the entire process of designing an interactive program. As a result, much of what the videos cover is not new. Instead it is HtDF and HtDD problems you already know how to do. We suggest that you use these videos as a kind of TA tutorial hours - try to work the problem on your own and check in with the video now and then to see how your solution compares to ours.

The material in this module should take **approximately 6-7 hours** of dedicated time to complete, including working along with the lecture videos, doing the practice problems, doing the homework problems and the short quiz.

## Learning Goals

- Be able to identify domain information that should be represented as compound data.
- Be able to read and write `define-struct` definitions.
- Be able to design world programs that use compound world state.

## Lecture Videos, Notes and Starter Files

Topic	Length (mm:ss)	Starter File	Downloads
<a href="#">Compound - define-struct</a> A new mechanism from the BSL language allows us to build multi-part (or compound) values and later to deconstruct the compound values to get the individual values back.	5:27	<i>none</i>	    
<a href="#">Compound - Compound Data Definitions</a> How to form compound data definitions using the HtDD recipe.	6:19	<a href="#">compound-starter.rkt</a>	    
<a href="#">Compound - HtDW With Compound Data</a> This is a <b>very long video</b> . That is because it goes step-by-step through the development of a world program with compound world state. In that sense much of <b>the video is like TA office hours</b> - you can work through the problem on your own consulting the video now and then to compare your solution to ours. Part of	29:35	<a href="#">cowabunga-starter.rkt</a>	  

the video goes into more detail on how to handle the geometry of keeping moving elements inside the screen, which is an area that students often ask about. Another part introduces helper functions.



## Lecture Problems

Module Kind #	Assignment	Duration	Difficulty	Code Files	Requires Lecture
Compound L3	A cow that wanders back and forth across the screen.	120 min.	■	<a href="#">cowabunga-starter.rkt</a> <a href="#">cowabunga-v0.rkt</a> <a href="#">cowabunga-v1.rkt</a> <a href="#">cowabunga-v2.rkt</a> <a href="#">cowabunga-v3.rkt</a> <a href="#">cowabunga-v4.rkt</a> <a href="#">cowabunga-v5.rkt</a> <a href="#">cowabunga-v6.rkt</a>	Compound-htdw

## Practice Problems

Module Kind #	Assignment	Duration	Difficulty	Code Files	Requires Lecture
Compound P1	Design a compound data definition to represent students, and a function to monitor their allergies.	25 min.	■	<a href="#">student-starter.rkt</a> <a href="#">student-solution.rkt</a>	Compound - dd
Compound P2	Design a compound data definition to represent trips, and a function to compare their lengths.	25 min.	■	<a href="#">trip-starter.rkt</a> <a href="#">trip-solution.rkt</a>	Compound - dd
Compound P3	Design a world to represent grass that grows and is replanted.	80 min.	◆	<a href="#">growing-grass-starter.rkt</a> <a href="#">growing-grass-solution.rkt</a>	Compound - htdw

Compound P5	Design a world where a lambda rolls back and forth across the screen.	100 min.	◆	<a href="#">rolling-lambda-starter.rkt</a> <a href="#">rolling-lambda-solution.rkt</a>	Compound - htdw
Compound P4	Design a world where the mouse position is displayed at the mouse cursor.	50 min.	■	<a href="#">tracker-starter.rkt</a> <a href="#">tracker-solution.rkt</a>	Compound - htdw

## Homework Problems

Module Kind #	Assignment	Duration	Difficulty	Code Files	Requires Lecture
Compound H1	Design a compound data definition to represent movies, and a function to compare their release dates.	25 min.	■	<a href="#">movie-starter.rkt</a> <a href="#">movie-solution.rkt</a>	Compound - dd
Compound H2	Design a world to represent a growing and rotating red box.	80 min.	◆	<a href="#">spinning-starter.rkt</a> <a href="#">spinning-solution.rkt</a>	Compound - htdw

## Module Quiz

Be sure to complete the homework problems before you do the module quiz. The quiz itself can be found on the [All Quizzes](#) page.

## Tips for Success

Be patient and take the time to be sure you work through several complete HtDW and Compound problems. They are longer, so it will take time, but you will need this material for the project and for next module's work, and it will also help you cement your understanding of prior modules.

This module more of you should begin to see that the design method is helping you organize and simplify your work on larger problems. Some of you may still feel you could do these problems 'in your head', but please follow the recipe anyways.

Created Thu 29 Aug 2013 6:40 PM EDT (UTC -0400)

Last Modified Mon 16 Sep 2013 3:25 PM EDT (UTC -0400)