```
;;
;; FILE:     homework02.rkt
;; AUTHOR:   Nick Sanford
;; DATE:     1/29/19
;; COMMENT:  Provides templates for your solutions, plus a few tests.
;;
;; MODIFIED:
;; CHANGE:
;;

#lang racket
(require rackunit)      ; enables you to use rackunit tests

; ------------------
; -----   [1]   -----
; ------------------

(define price-per-ounce
  (lambda (units-per-pack ounces-per-unit pack-price)
    (/ pack-price (* units-per-pack ounces-per-unit))))                ; replace the 0 with
your code

(check-equal? (price-per-ounce 6 24   1.44) 0.01)
(check-equal? (price-per-ounce 6 16.9 1.44) 0.014201183431952664)

; ------------------
; -----   [2]   -----
; ------------------

(define ladder-height
  (lambda (ladder-length base-distance)
    (sqrt (- (expt ladder-length 2) (expt base-distance 2)))))                ; replace the 0
with your code

(check-equal? (ladder-height 10 6)   8)
(check-equal? (ladder-height 13 5)   12)
(check-equal? (ladder-height 20 3.5) 19.691368667515217)

; ------------------
; -----   [3]   -----
; ------------------

(define candy-temperature
  (lambda (temp elevation)
    (- temp (/ elevation 500.0))))               ; replace the 0 with your code

; write your own check-equal? tests for the three examples
(check-equal? (candy-temperature 244 5280)   233.44)
(check-equal? (candy-temperature 302 977.69) 300.04462)
(check-equal? (candy-temperature 302 -1401)  304.802)

; ------------------
; -----   [4]   -----
; ------------------

(define in-range?
  (lambda (actual desired epsilon)
    (<= (abs (- desired actual)) epsilon)))              ; replace the #f with your code
```