

## Danger and Target Movement

**Directions:** Use the Design Recipe to write a function `update-danger`, which takes in the danger's x-coordinate and produces the next x-coordinate.

## Contract and Purpose Statement

*Every contract has three parts...*

```
function name : domain -> range ;  
what does the function do?
```

## Examples

Write some examples, then circle and label what changes...

( EXAMPLE (                      )                      )

*function name*                  *input(s)*                      *what the function produces*

( EXAMPLE (                      )                      )

*function name*                  *input(s)*                      *what the function produces*

## Definition

Write the definition, giving variable names to all your input values...

```
(define ( function name variable(s) )
  what the function does with those variable(s)
)
```

**Directions:** Use the Design Recipe to write a function `update-target`, which takes in the danger's x-coordinate and produces the next x-coordinate.

## Contract and Purpose Statement

Every contract has three parts...

```

; _____ : _____ -> _____
   function name          domain          range
;
_____
               what does the function do?

```

## Examples

Write some examples, then circle and label what changes...

( EXAMPLE (                      )                      )

*function name*                      *input(s)*                      *what the function produces*

(EXAMPLE (                      )                      )

<i>function name</i>	<i>input(s)</i>	<i>what the function produces</i>
----------------------	-----------------	-----------------------------------

## Definition

Write the definition, giving variable names to all your input values...

```
(define ( function name variable(s) )
  what the function does with those variable(s)
)
```