1. Students often confuse `map` and `apply`. Explain the difference between them, using the following code snippet as an example:

   ```racket
   (map list '(1 2 3))
   (apply list '(1 2 3))
   ```

   If you have forgotten what the `list` function does, try it out in the interactive REPL.

2. Beside each expression below, write its value, or the (approximate) error raised when Racket attempts to evaluate it.

   ```racket
   (filter even? '(1 2 3 4 5))
   (map even? '(1 2 3 4 5))
   (apply and '(#t #f #t #f #t))
   ```

3. Students often get confused when passing in a function as an argument, especially if they haven’t seen this before. Here is an example of one common mistake students make:

   ```racket
   (filter (odd?) '(1 2 3 4 5))
   ```

   Explain what error Racket would raise, and why this happens.