1. Are these primitive expressions or call expressions?

primitive	call	`3`
primitive	call	`add(2, 3)`
primitive	call	`print(5)`
primitive	call	`3 + 4`

   Solution: primitive, call, call, call

2. When do we make a new frame in an environment diagram?

   Solution: After evaluating the operator and then the operands. What does it mean to evaluate? We don't execute the body of the function yet. Instead, ask yourself: What is the **value** of the operator? If the operator is a name, then look at the environment diagram, and see what value name points to. Sometimes, evaluating the operator is tricker, and involves more function calls. Try to write an example of a call expression where evaluating the operator involves another function call!

3. **Challenge:** Fill in the blanks below so that the output is as follows.

```
>>> def ne(mo):
...     def do(ry):
...         return mo("Just keep "+ ry)
...     return do
>>> def mar(lin):
...     print(lin())
>>> mar(ne(_____)(_____))
Just keep swimming
```

   Solution:

```
lambda x: lambda x:
'swimming'
```