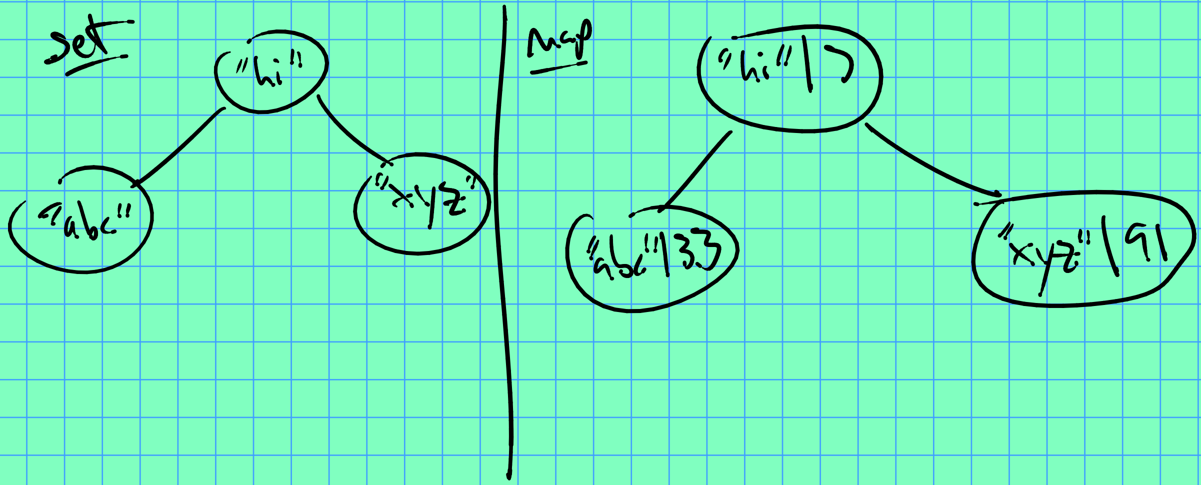
Maps

Similar to set, but stores key/value pairs (instead of just keys like set).



Think of it as a partially defined function. We explicitly prescribe the input/output behavior for some subset of the domain.

Very standard example: frequency table:
Read a list of strings from stdin, + print how many times each one occurred.

```
echo "hi hi loool" | ./freq

    hi: 2
    loool: 1
```

How to do this w/o maps? 2 vectors?

strings	counts
hi	2
loool	1

Now with maps:

```
map<string, int> F;
string s; // store input
while (cin>> s)
      F[s]++;  <——  Note: the int
                    F[s] will default
                       to 0.
// Now just print!!
for( iterator... )
     cout << (*i).first   <<":"<<(*i).second  ;"
                   ///                  |||
                    s                  F[s]
```

Another view: kind of like vector, but you can index with non-integer types.

map<int, int> ≋ vector<int>

but they are different!

Exercise: compare & contrast.