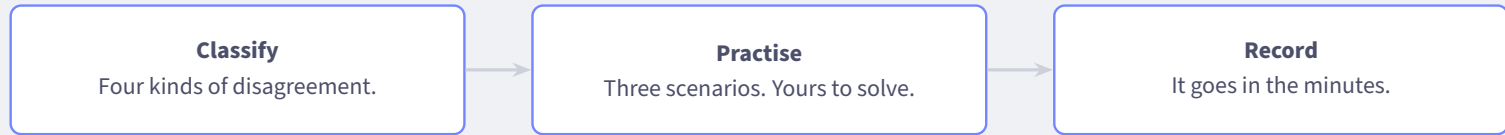


Group dynamics

Disagreement is normal. Handling it is a skill.



Disagreement is normal.

A group that never disagrees is either dominated by one person, or not engaging with the work.

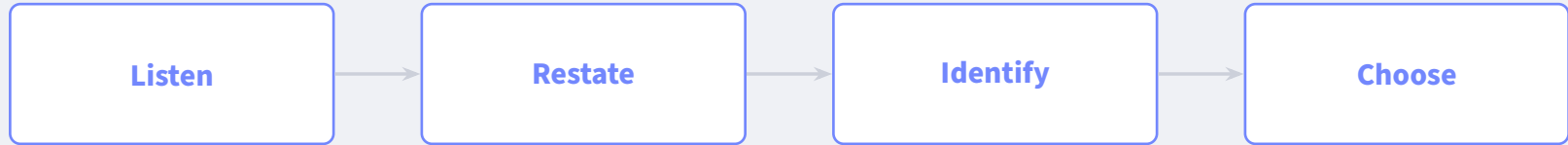
Fact

Approach

Priority

Effort

How would you settle each one?



Restate is the one everybody skips.

B wants the tests written before the functions.

C wants the library first, then the tests.

Neither will move. Fifteen minutes gone.

Which kind? What happens next?



Into the minutes:

A: “it runs without an error, so it is correct”.

D: “it is wrong”.

Ten minutes. Both getting irritated.

Which kind? What do they do right now?



How long should this have taken?

D's function is still not written.

Second week running.

Two members are messaging each other privately about it.

Which kind? What is wrong with the messages?



“The function that was due last Tuesday is not yet written.”

“D is lazy and does not care.”

In a meeting. Not outside one.

1. **Round-robin** the discussion. Anyone may pass.
2. **Check each decision** one member at a time. A nod is not enough.

Artificial the first time. Ordinary by week three.

Take a **break**. **Try both** for a week.

Defer to the chair. **Write it down**, one sentence each.

It goes in the minutes either way: resolved, or still open.

Object oriented programming

Bundling data and the things
you do to it.

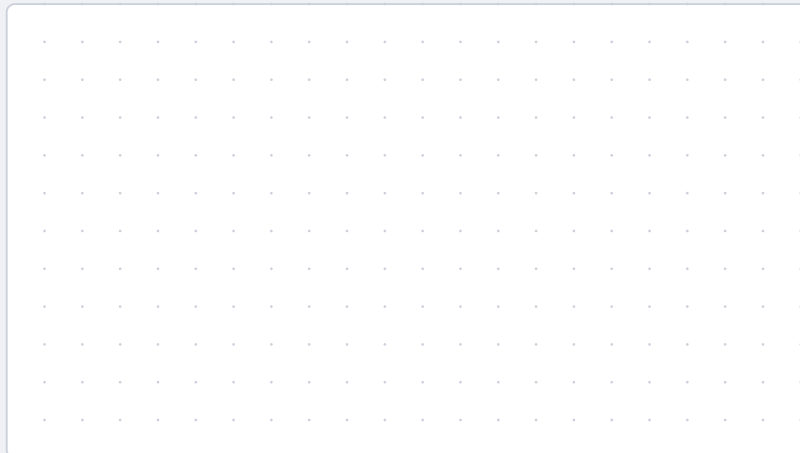
A terminal and an editor

Writing Python in a file, and
running it without a notebook.

You read these **on your own**. The next three slides are the parts that catch people.

```
class Polynomial:
    def __init__(self, coefficients):
        self.coefficients = coefficients
    def degree(self):
        return len(self.coefficients) - 1
```

degree takes no argument when you call it. So what is self?

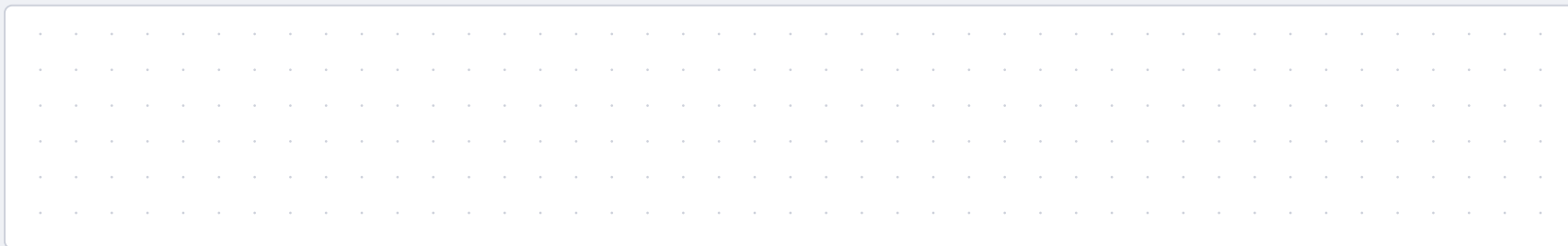


The instance it was called on. Python passes it for you, which is why it is there in the definition and not in the call.

No.

A library of plain functions is a perfectly good library, and it is marked the same way.

So when is a class worth it?



```
cd project
ls
python main.py
```

You hand in a .py file, not a notebook. Why?

`cd` moves, `ls` looks, and `python main.py` runs the file. A notebook remembers what you ran earlier; a file does not, which is the point.

Your minutes have a box for this

Disagreements and decisions

The disagreement, which of the four kinds, and the resolution or the next step

Rotate the roles

A new **chair** and a new **secretary**

Everyone does each job

Meet this week. Bring the Week 5 minutes to the next session. The second of the five that *I* is counted from.