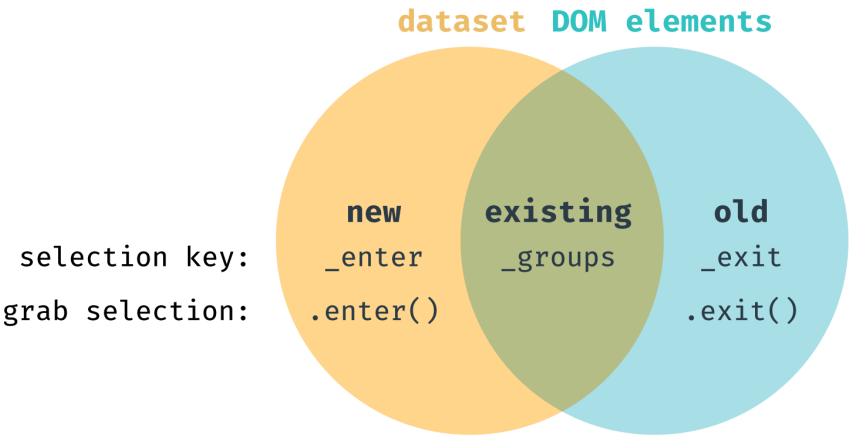
If we expand the `__data__` value, we will see one our data points.

Great! We can see that each value in `_enter` corresponds to a value in our dataset. This is what we would expect, since all of the data points need to be added to the DOM.

The `_exit` value is an empty array — if we were removing existing elements, we would see those listed out here.

In order to act on the **new** elements, we can create a d3 selection object containing just those elements with the `enter` method. There is a matching method (`exit`) for **old** elements that we'll need when we go over transitions in **Chapter 4**.



join schematic with selection calls

Let's get a better look at that new selection object:

```
const dots = bounds.selectAll("circle")
    .data(dataset)
  .enter()
console.log(dots)
```

This looks just like any d3 selection object we've manipulated before. Let's append one `<circle>` for each data point. We can use the same `.append()` method we've been using for single-node selection objects and d3 will create one element for each data point.