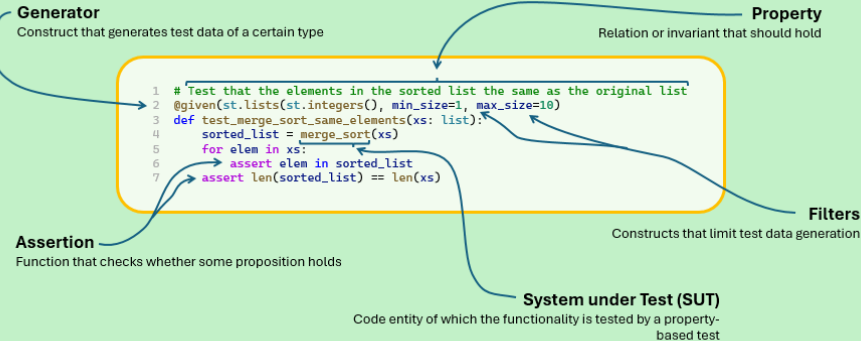


# How is property-based testing used in real-world Python projects?

## A case study of the Hypothesis framework

### What is Property-Based Testing (PBT)?

Testing functionality with large amounts of automatically generated data



### How did we Collect and Analyze Data?

#### Qualitative Analysis through Open Coding

```
test_base64_roundtrip
# Hypothesis
> hypothesis.given(
  binary_hypothesis_strategies.binary(),
  newline_hypothesis_strategies.booleans(),
  ...
)
def test_base64_roundtrip(self, binary, newline):
    converted = base64.b2a_base64(binary, newline=newline)
    restored = base64.a2b_base64(converted, newline=newline)
    assert converted == restored, 'base64 roundtrip failed'
```

#### Open Coding

Assigning labels to parts of the tests to identify emerging concepts.

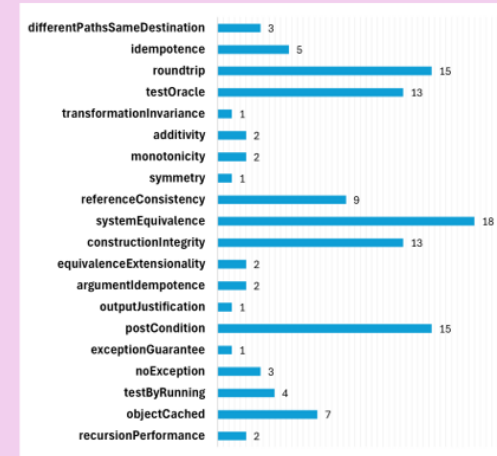
#### Quantitative Analysis in a Spreadsheet

Repository	Test name	#assertions	Custom generation	Category
cpython	test_count	1	FALSE	Test Oracle
streamlit	test_map_set_del	2	TRUE	Postcondition
...	...	...	...	...
Average/Total	-	4.4	34	-

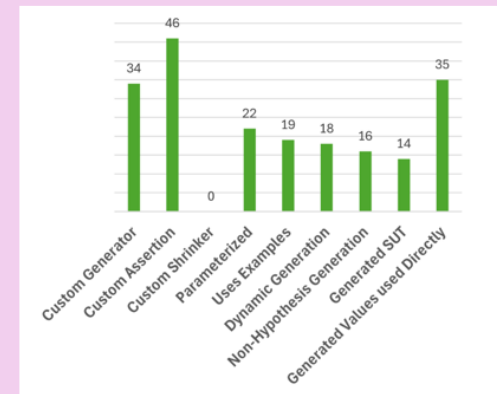
### What did we find?

Top Categories & Key Features

- **Roundtrip**: testing an operation and its reverse.
- **Post-condition**: ensuring a simple condition on the outcome of the SUT.
- **Oracle-based testing**: checking an SUT against
  - an equivalent system,
  - a reference implementation,
  - a test oracle.
- Sometimes used for non-functional properties.
- Often a generalization of unit tests.



Frequencies of categories identified in 87 property-based tests that use Hypothesis in Python.



Frequencies of key features identified in 87 property-based tests that use Hypothesis in Python.

### What does that mean?

Improving frameworks and education

1. Using property-based testing to **generalize unit tests**.
2. **Custom shrinking** possibly **unnecessary** – if implemented well by framework.
3. **Decomposing** property-based tests to simplify them
  - If multiple assertions: split on assertion,
  - If assertions conditionalized: split on condition,
  - If SUT generated by PBT framework: split on SUT.

### What more could be researched?

The future is in our hands...

- **Extend dataset** to include more repositories
- Different **selection** procedure for repositories
- More **cross-framework** analysis
- **Temporal analysis** using Git histories
- **Automated analysis** using machine learning