Working with Rational Numbers Using `fractions.Fraction`

🔷 Code Explanation

python Copy Edit

```
>>> from fractions import Fraction >>> myFra = Fraction(2, 7) >>> myFra Fraction(2, 7)
```

🔍 What's Happening Here:

- `Fraction(2, 7)` creates a rational number **2/7**.
- The `Fraction` class stores values **exactly as rational numbers** (numerator and denominator).
- No approximation occurs—unlike floats, this is **mathematically exact**.

✅ Advantages of Using `Fraction`

Feature	Description
Exact Math	100% accurate representation of rational numbers
Readable Format	Keeps the fraction form (like `3/4`)
Auto Simplification	`Fraction(4, 8)` → `Fraction(1, 2)`
Supports Operations	Can add, subtract, multiply, divide fractions

---

🔶 Example: Accurate Arithmetic

python Copy Edit

```
>>> Fraction(1, 3) + Fraction(1, 3) Fraction(2, 3) >>> Fraction(1, 10) * Fraction(2, 5)
Fraction(1, 25)
```

This level of **precision and clarity** is great for:

- Mathematical modeling
- Educational software
- Financial systems (if values are rational)

VS Comparison

Operation	`float` Result	`Fraction` Result
`1/3 + 1/3`	`0.666666666...`	`Fraction(2, 3)`
`0.1 + 0.2`	`0.30000000000000004`	`Fraction(3, 10)`