# Ratios, Proportions & Percentages

## Why This Matters

Ratios, proportions, and percentages are how we **compare** and **scale** quantities. They answer questions like:

- "How does this compare to that?"
- "If I double this, what happens to that?"
- "What's 15% of 200?"
- "Is this a good deal?"

As a developer, you encounter these constantly:

- API rate limits (requests per second)
- Image aspect ratios (16:9)
- CPU usage (75%)
- Progress bars (60% complete)
- Scaling layouts (responsive design)

Let's build intuition from scratch.

---

## The Big Picture: Comparison and Scaling

**Ratios**: Compare two quantities

```
Coffee to water: 1:16
(1 part coffee, 16 parts water)
```

**Proportions**: Maintain ratios when scaling

```
If 1:16 works for 1 cup, then 2:32 works for 2 cups
```

**Percentages**: Special ratio comparing to 100

```
75% = 75 out of 100 = 3 out of 4
```

All three concepts are deeply related.

---

## 1. Ratios: Comparing Quantities

### What They Are

A **ratio** compares two quantities by division.

**Notation**: `a : b` or `a/b` or "a to b"

Examples:

```
Pizza slices: 3 pepperoni : 5 cheese
Speed: 60 miles : 1 hour (60 mph)
Recipe: 2 eggs : 1 cup flour
```

### Mental Model: Parts of a Whole

If you have a ratio of 3:5 (red:blue):

```
Red:  ■■■
Blue: ■■■■■
Total: 8 parts (3 red + 5 blue)
```

### Interpreting Ratios

**3:5** means:

- For every 3 reds, there are 5 blues
- Red is 3/8 of the total
- Blue is 5/8 of the total
- The relationship is 3/5 or 0.6 (reds per blue)

### Programming Analogy: Aspect Ratios

```javascript
const aspectRatio = {
  width: 16,
  height: 9
};

// Maintain ratio when scaling
function scale(aspectRatio, newWidth) {
  const ratio = aspectRatio.width / aspectRatio.height;
  return {
    width: newWidth,
    height: newWidth / ratio
  };
}

scale(aspectRatio, 1920);  // { width: 1920, height: 1080 }
```

### Simplifying Ratios

Just like fractions, ratios can be simplified:

```
6:8  →  3:4  (divide both by 2)
10:15 → 2:3  (divide both by 5)
```

**Method**: Find the GCD (greatest common divisor) and divide both sides.

### Ratios with More Than Two Parts

```
Red:Green:Blue = 2:3:5

Total parts: 2 + 3 + 5 = 10

If you have 100 items:
Red:   20 (2/10 of 100)
```

```
Green: 30 (3/10 of 100)
Blue:  50 (5/10 of 100)
```

### Unit Ratios

A **unit ratio** has one side equal to 1:

```
3:5  →  1:1.67  (divide by 3)
     or  0.6:1  (divide by 5)
```

This is useful for "per unit" comparisons:

- Miles per gallon (mpg)
- Requests per second
- Price per item

---

## 2. Proportions: Keeping Ratios Constant

### What They Are

A **proportion** is an equation stating that two ratios are equal.

```
a     c
─  =  ─
b     d
```

Read as: "a is to b as c is to d"

### Example: Recipe Scaling

Original recipe (serves 4):

```
2 eggs : 4 servings
```

How many eggs for 10 servings?

```
2 eggs      x eggs
──────  =  ──────
4 servings 10 servings
```

**Cross-multiply** to solve:

```
2 × 10 = 4 × x
20 = 4x
x = 5 eggs
```

### Cross-Multiplication (Why It Works)

```
a     c
─  =  ─
b     d
```

```
Multiply both sides by bd:

a × d = b × c
```

This gives you a simple equation to solve.

### Visual: Proportion as Scaling

```
Original:
■■ (eggs)
────────────
■■■■ (servings)

Scaled:
■■■■■ (eggs)
────────────────────────
■■■■■■■■■■ (servings)

The ratio stays the same: 1 egg per 2 servings
```

### Example: Map Scale

Map scale: 1 inch = 50 miles

If two cities are 3.5 inches apart on the map, what's the real distance?

```
1 inch      3.5 inches
────────  =  ──────────
50 miles      x miles

1 × x = 50 × 3.5
x = 175 miles
```

### Programming Analogy: Responsive Scaling

```
// Original dimensions
const original = { width: 800, height: 600 };

// Scale to new width, maintain proportion
function scaleProportionally(original, newWidth) {
  const ratio = original.height / original.width;
  return {
    width: newWidth,
    height: newWidth * ratio
  };
}

scaleProportionally(original, 1200);  // { width: 1200, height: 900 }
```

---

## 3. Percentages: Ratios Out of 100

### What They Are

**Percent** means "per hundred" (Latin: *per centum*).

```
75% = 75/100 = 0.75
```

Percentages are just a convenient way to express fractions and ratios.

### Mental Model: Parts per 100

Think of percentages as **standardized ratios** where the whole is always 100:

```
75% means:
■■■■■■■■■■  (10 rows)
■■■■■■■■■■
■■■■■■■■■■
■■■■■■■■■■
■■■■■■■■■■
■■■■■■■■■■
■■■■■■■■■■
■■■■■■■■○○  ← 75 filled out of 100
■■■■■○○○○○
○○○○○○○○○○
```

### Converting Between Forms

| Percent | Decimal | Fraction |
|---|---|---|
| 50% | 0.50 | 1/2 |
| 25% | 0.25 | 1/4 |
| 75% | 0.75 | 3/4 |
| 10% | 0.10 | 1/10 |
| 100% | 1.00 | 1/1 |
| 150% | 1.50 | 3/2 |

**Percent to Decimal**: Divide by 100

```
75% = 75/100 = 0.75
```

**Decimal to Percent**: Multiply by 100

```
0.75 = 0.75 × 100% = 75%
```

**Percent to Fraction**: Put over 100 and simplify

```
75% = 75/100 = 3/4
```

## Three Types of Percentage Problems

### Type 1: Find the Percentage of a Number

**"What is 20% of 150?"**

```
20% × 150 = 0.20 × 150 = 30
```

**Method**:

1. Convert percent to decimal
2. Multiply

**Programming**:

```
const percent = 20;
const number = 150;
const result = (percent / 100) * number;  // 30
```

### Type 2: Find What Percent One Number Is of Another

**"30 is what percent of 150?"**

```
30/150 = 0.2 = 20%
```

**Method**:

1. Divide the part by the whole
2. Convert to percent (multiply by 100)

**Programming**:

```
const part = 30;
const whole = 150;
const percent = (part / whole) * 100;  // 20
```

### Type 3: Find the Whole When Given a Percentage

**"30 is 20% of what number?"**

```
30 = 0.20 × x
x = 30 / 0.20 = 150
```

**Method**:

1. Set up equation: part = percent × whole
2. Solve for whole: whole = part / percent

**Programming**:

```
const part = 30;
const percent = 20;
const whole = part / (percent / 100);  // 150
```

---

## 4. Percentage Increase and Decrease

### Percentage Increase

**Formula**:

```
Percentage Increase = (New - Old) / Old × 100%
```

**Example**: Price went from $50 to $65

```
Increase = 65 - 50 = 15
Percentage = 15/50 × 100% = 30%
```

The price increased by 30%.

### Percentage Decrease

**Formula**:

```
Percentage Decrease = (Old - New) / Old × 100%
```

**Example**: Price dropped from $80 to $60

```
Decrease = 80 - 60 = 20
Percentage = 20/80 × 100% = 25%
```

The price decreased by 25%.

### Adding/Subtracting Percentages

**Increase by 20%**:

```
New Value = Old Value × 1.20

If old = 100:
New = 100 × 1.20 = 120
```

**Decrease by 20%**:

```
New Value = Old Value × 0.80

If old = 100:
New = 100 × 0.80 = 80
```

**General Formula**:

```
Increase by x%: multiply by (1 + x/100)
Decrease by x%: multiply by (1 - x/100)
```

### Common Mistake: Percentages Don't Add Symmetrically

If you increase by 50% then decrease by 50%, you don't get back to the original:

```
Start: 100
Increase by 50%: 100 × 1.5 = 150
Decrease by 50%: 150 × 0.5 = 75  (NOT 100!)
```

**Why?** The 50% decrease is calculated from the new base (150), not the original (100).

### Programming: Applying Percentage Changes

```
function applyPercentChange(value, percentChange) {
  return value * (1 + percentChange / 100);
}

applyPercentChange(100, 20);   // 120 (20% increase)
applyPercentChange(100, -20);  // 80 (20% decrease)
```

---

## 5. Real-World Applications

### Sales and Discounts

**Original price: $120, 25% off**

```
Discount = 120 × 0.25 = $30
Sale price = 120 - 30 = $90

Or directly:
Sale price = 120 × 0.75 = $90
```

### Sales Tax

**Item costs $50, 8% tax**

```
Tax = 50 × 0.08 = $4
Total = 50 + 4 = $54

Or directly:
Total = 50 × 1.08 = $54
```

### Tip Calculation

**Bill is $80, 18% tip**

```
Tip = 80 × 0.18 = $14.40
Total = 80 + 14.40 = $94.40
```

### Interest (Simple)

**$1000 invested at 5% annual interest for 3 years**

```
Interest per year = 1000 × 0.05 = $50
Total interest = 50 × 3 = $150
```

```
Final amount = 1000 + 150 = $1150
```

### CPU/Memory Usage

```
const totalMemory = 16000;  // MB
const usedMemory = 12000;   // MB
const percentUsed = (usedMemory / totalMemory) * 100;  // 75%
```

### Progress Bars

```
function updateProgress(completed, total) {
  const percent = (completed / total) * 100;
  progressBar.style.width = `${percent}%`;
  label.textContent = `${Math.round(percent)}%`;
}
```

### Image Scaling (Aspect Ratio)

```
// Maintain 16:9 aspect ratio
function calculateHeight(width) {
  return width * (9 / 16);
}

calculateHeight(1920);  // 1080
```

### API Rate Limits

```
Rate limit: 100 requests per minute
Used: 73 requests
Percentage used: 73/100 = 73%
Remaining: 27%
```

---

## 6. Proportional Reasoning in Programming

### Scaling Coordinates

```
// Scale canvas from 800x600 to 1600x1200 (2x)
function scalePoint(point, scaleFactor) {
  return {
    x: point.x * scaleFactor,
    y: point.y * scaleFactor
  };
}

scalePoint({ x: 100, y: 150 }, 2);  // { x: 200, y: 300 }
```

### Normalized Values (0 to 1)

```
// Normalize value to 0-1 range
function normalize(value, min, max) {
  return (value - min) / (max - min);
}

normalize(50, 0, 100);   // 0.5 (50%)
normalize(75, 0, 100);   // 0.75 (75%)

// Convert back
function denormalize(normalized, min, max) {
  return normalized * (max - min) + min;
}

denormalize(0.5, 0, 100);  // 50
```

### Animation Easing (Proportional Speed)

```
// Linear interpolation (lerp)
function lerp(start, end, t) {
  // t is a percentage (0 to 1)
  return start + (end - start) * t;
}

lerp(0, 100, 0.5);   // 50 (halfway)
lerp(0, 100, 0.25);  // 25 (quarter way)
```

### Resource Allocation

```
// Distribute resources proportionally
function allocate(resources, ratios) {
  const total = ratios.reduce((sum, r) => sum + r, 0);
  return ratios.map(r => (r / total) * resources);
}

allocate(100, [1, 2, 3]);  // [16.67, 33.33, 50]
```

---

## Common Mistakes & Misconceptions

### ❌ "50% of X plus 50% more is 100% of X"

No! 50% + 50% of 50% = 75%

```
Start: 100
Add 50%: 100 × 1.5 = 150
This is 50% more, not 100% more
```

### ❌ "Percentages can't be over 100%"

They absolutely can. 200% means twice as much.

```
100% = the whole thing
200% = twice the whole thing
50% = half the whole thing
```

### ❌ "Ratios and fractions are different"

They're the same thing:

```
Ratio 3:4 = Fraction 3/4 = Decimal 0.75 = Percent 75%
```

### ❌ "You can compare percentages without knowing the base"

"20% increase" means nothing without knowing what it's 20% *of*:

```
20% of 100 = 20
20% of 1000 = 200
```

### ❌ "0.5 is 5%"

No! 0.5 is 50%. To convert decimal to percent, multiply by 100.

---

## Mental Math Tricks

### Common Percentages

**10%**: Divide by 10 (move decimal left)

```
10% of 340 = 34
```

**5%**: Half of 10%

```
5% of 340 = 17
```

**20%**: Double 10%

```
20% of 340 = 68
```

**25%**: Divide by 4

```
25% of 340 = 85
```

**50%**: Divide by 2

```
50% of 340 = 170
```

**1%**: Divide by 100

```
1% of 340 = 3.4
```

### Tip Calculation Shortcut

**15% tip**:

```
10% = move decimal left one place
5% = half of 10%
15% = 10% + 5%

Bill: $80
10% = $8
5% = $4
15% = $12
```

---

## Visual Examples

### Ratio Visualization

```
Ratio 2:3 (apples to oranges)

Apples:  🍎🍎
Oranges: 🍊🍊🍊

Total: 5 fruits
Apples: 2/5 = 40%
Oranges: 3/5 = 60%
```

### Percentage Bar

```
0%           25%          50%          75%         100%
├─────────────┼────────────┼────────────┼────────────┤
│■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■│

75% filled:
├─────────────┼────────────┼────────────┼────────────┤
│■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■░░░░░░░░░░░░░│
```

### Proportion Scaling

```
Original (4:3 aspect ratio):
┌────────────┐
│            │
│    4:3     │
│            │
└────────────┘

Scaled 2x (maintains ratio):
┌────────────────────────┐
│                        │
│                        │
│         4:3            │
```

```
│                              │
│                              │
└──────────────────────────────┘
```

---

## Tiny Practice

1. **Simplify the ratio**: 12:18
2. **Solve the proportion**: 3/5 = x/20
3. **Convert to percent**: 0.85
4. **Convert to decimal**: 42%
5. **What is 30% of 250?**
6. **15 is what percent of 60?**
7. **45 is 15% of what number?**
8. **Increase 80 by 25%**
9. **Decrease 120 by 40%**
10. **If a recipe calls for 3 cups flour for 12 cookies, how many cups for 20 cookies?**

▶ Answers

---

## Summary Cheat Sheet

### Ratios

```
a:b  =  a/b  =  "a to b"

Simplify: divide both by GCD
Compare: convert to decimals or unit ratios
```

### Proportions

```
a/b = c/d  ⟹  a×d = b×c  (cross-multiply)

Use for: scaling, unit conversion, recipe adjustment
```

### Percentages

```
Percent = (Part / Whole) × 100

To decimal: divide by 100
To fraction: put over 100, simplify
```

### Percentage Operations

| Operation | Formula | Example |
| --- | --- | --- |
| X% of Y | (X/100) × Y | 20% of 150 = 30 |
| X is what % of Y | (X/Y) × 100 | 30 is 20% of 150 |
| X is Y% of what | X / (Y/100) | 30 is 20% of 150 |

| | | |
|---|---|---|
| Increase by X% | Value × (1 + X/100) | 100 increased by 20% = 120 |
| Decrease by X% | Value × (1 - X/100) | 100 decreased by 20% = 80 |

### Programming Patterns

```
// Ratio as object
const ratio = { a: 16, b: 9 };

// Proportion solving
const x = (b * c) / a;

// Percentage calculation
const percent = (part / whole) * 100;

// Apply percentage change
const newValue = value * (1 + percent / 100);

// Normalize to 0-1
const normalized = (value - min) / (max - min);
```

---

## Next Steps

You now understand how to compare, scale, and work with ratios and percentages. These concepts are fundamental for understanding growth, change, and relationships between quantities.

Next, we'll explore **Powers, Roots, and Exponents**—how repeated multiplication works and why it's useful.

**Continue to**: [03-powers-roots-exponents.md](03-powers-roots-exponents.md)