

4565: Computing Foundations for a Digital Age

Unit 4

Understanding Data Formats: Tables, JSON, and Schema

Helping students (and teachers!) make sense of data structures

When Is a Table the Best Format?

Tables are ideal when:

- You have **many entries** with the **same structure**
- You want to **view all data at once** (like a spreadsheet)
- You plan to **sort, filter, or run calculations** (Excel/Sheets)

Great for:

- Grades
- Attendance
- Survey results

Example Table:

Name	Age	Grade
Ava	13	A
Ben	12	B

Why it works:

- Tables are **clean, organized**, and great for **pattern recognition** and quick comparisons.
-

JSON (JavaScript Object Notation) is best when:

- Your data is **nested, flexible**, or not every entry looks the same
- You're working between **apps, systems, or websites**
- You want a **structured but adaptable format**

Great for:

- Game profiles
- App settings
- Personalized content

Example JSON:

- ```
{
 "username": "avacoder",
 "age": 13,
 "badges": ["Explorer", "Builder"],
 "premiumUser": true
}
```

### Why it works:

JSON handles **complex** and **non-uniform** data beautifully. It's readable by computers and used behind the scenes in most apps and websites.

---

## When Should You Use a Schema?

**Schemas** are used to **define rules** for what kind of data is allowed. Think of it as the **blueprint or template**.

### Schemas are helpful when:

- You're building a **form, app, or database**
- You want to enforce **consistency**
- You need to validate input (e.g., a number must be positive)

### Great for:

- Form creation
- Database design
- API development

### Schema Example:

- This field expects a string.
- That field must be a number.
- This one can only be true/false.

### Why it works:

Schemas help keep data **predictable** and **safe** — especially when different people or systems are entering it.

---

### Quick Reference: What Format to Use When

| Situation                     | Best Format |
|-------------------------------|-------------|
| Lots of similar records       | ✓ Table     |
| Complex or nested data        | ✓ JSON      |
| Defining data structure/rules | ✓ Schema    |