CSS

**🧾 Comparison of CSS Methodologies: OOCSS vs BEM vs Atomic CSS**

| **Feature** | **OOCSS (Object-Oriented CSS)** | **BEM (Block Element Modifier)** | **Atomic CSS (Utility-first)** |
| --- | --- | --- | --- |
| 🔤 Full Form | Object-Oriented CSS | Block Element Modifier | Atomic CSS |
| 🎯 Goal | Reusable + maintainable CSS components | Clear naming and structured components | Fast UI building using utility classes |
| 🧱 Structure Style | Structure vs Skin | block\_\_element--modifier | Each class = one style |
| 🧠 Philosophy | Reuse visual patterns | Strict naming rules for CSS architecture | Apply styling directly in HTML |
| 🛠 CSS Needed? | Yes, reusable classes in CSS | Yes, structured CSS with clear names | Minimal/No CSS writing needed |
| ⚙️ Reusability | High | High | Very High |
| 📦 Example Classes | .card, .card-blue, .card-title | .card\_\_title, .card\_\_title--big | bg-blue-500, text-white, p-4 |
| 🧪 Example HTML | <div class="card card-blue"> | <div class="card\_\_title card\_\_title--big"> | <div class="bg-blue-500 p-4 text-white"> |
| 🚫 Drawbacks | Can become bloated | Verbose class names | Messy HTML, hard to read at scale |
| 💡 Used In | Large component-based systems | Enterprise design systems | Tailwind CSS, Fast Prototyping |