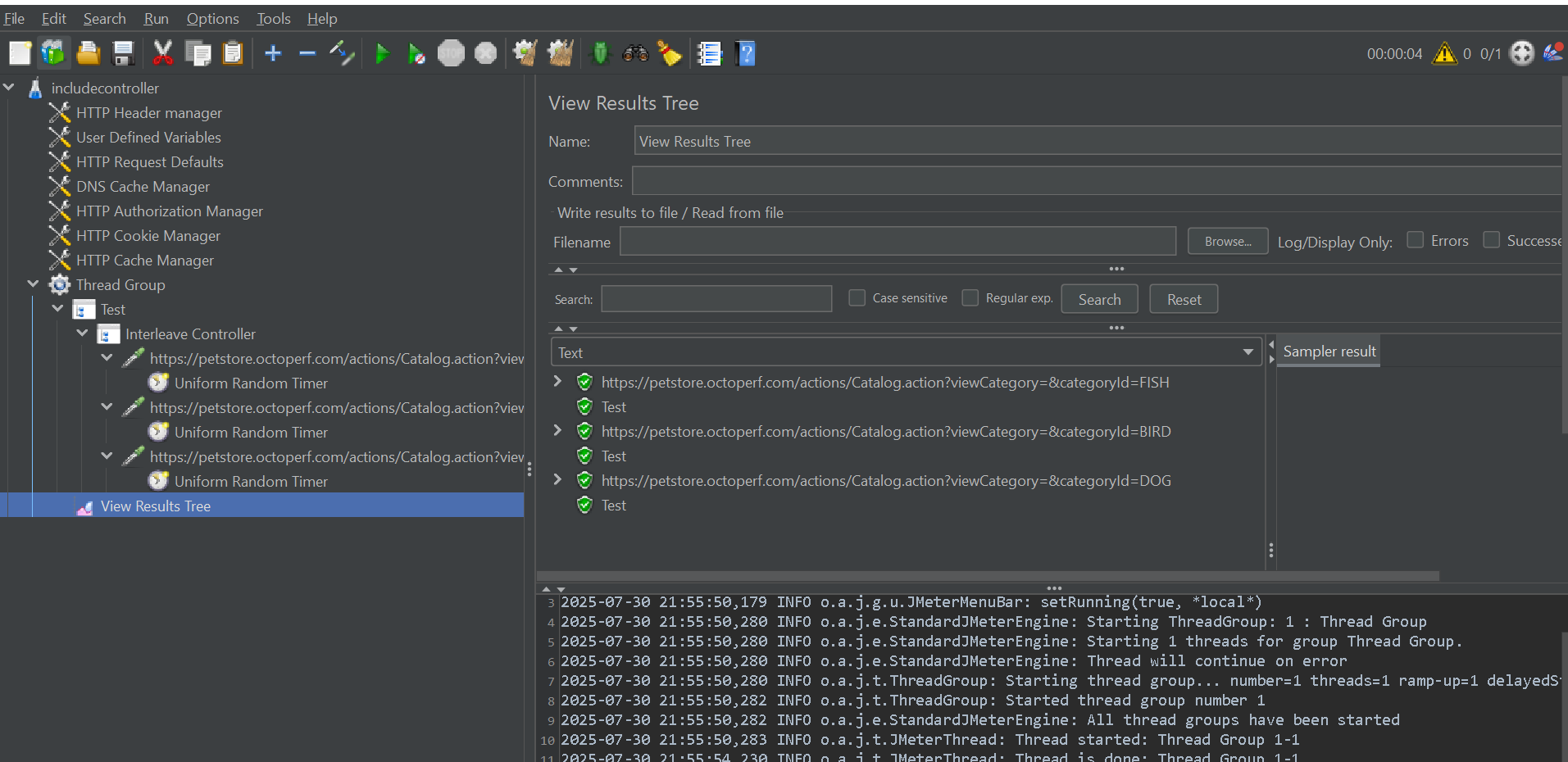
🔁 Interleave Controller





The **Interleave Controller** is a **Logic Controller** that executes **one child element per parent iteration**, following a **sequential round-robin** order. It's particularly useful when simulating alternating user actions or cycling through a series of requests across iterations.

**🔧 How It Works**

* For **each iteration** of the parent (e.g., Thread Group), it:
  + Executes **only one** of its direct children.
  + Moves to the **next child** in the next iteration.
  + **Cycles back** to the first child after the last one.

🔄 Think of it like a rotating wheel: each loop of the parent thread picks the next "spoke" (child element) in order.

**📌 Key Characteristics**

| **Feature** | **Behavior** |
| --- | --- |
| Execution per iteration | Executes **one child per iteration** |
| Order | **Sequential (not random)** |
| Repetition | **Loops back** to the beginning after the last |
| Scope | Operates within the parent controller's iteration |
| Children can be: | Samplers or other Logic Controllers |
| Optional setting: | **Ignore Sub-Controller Blocks** (see below) |

**📋 Use Cases**

| **Scenario** | **Description** |
| --- | --- |
| Alternating page views | Rotate visits to /home, /about, /contact in order |
| Controlled A/B testing | Alternate between Feature A and Feature B in a loop |
| Data lifecycle testing | Iterate over Create → Process → Cleanup flows |
| User journey simulation | One thread simulates a different user action per visit/iteration |

**📥 How to Add & Configure**

**➕ Add It:**

1. **Right-click** on your Thread Group or parent controller.
2. Select: Add > Logic Controller > Interleave Controller.

**➕ Add Children:**

* Add HTTP Requests or other Logic Controllers under the Interleave Controller.
* The order in which you add matters.

**🧪 Example: Website Navigation Simulation**

**🏗️ Test Plan Structure:**

text

CopyEdit

Test Plan

└─ Thread Group (1 thread, Loop Count: 5)

└─ Interleave Controller (Name: User Navigation)

├─ HTTP Request - Home Page (/home)

├─ HTTP Request - About Page (/about)

└─ HTTP Request - Contact Page (/contact)

**🧾 Execution Flow:**

| **Iteration** | **Request Executed** |
| --- | --- |
| 1 | Home Page |
| 2 | About Page |
| 3 | Contact Page |
| 4 | Home Page (again) |
| 5 | About Page (again) |

**⚙️ Ignore Sub-Controller Blocks Option**

| **Setting** | **Effect** |
| --- | --- |
| ❌ Unchecked (default) | Treats each child controller as a whole unit |
| ✅ Checked | Interleaves **inner samplers** even if nested inside child controllers |

📝 Use unchecked unless you specifically want to dive into nested controllers.

**🔄 Comparison Table: Interleave vs. Random vs. Random Order**

| **Controller** | **Executes** | **Selection** | **Repeats within iteration?** |
| --- | --- | --- | --- |
| **Interleave** | 1 child per iteration | Sequential | ❌ |
| **Random Controller** | 1 child per iteration | Random | ❌ |
| **Random Order** | All children | Random order | ✅ |

**🧠 Pro Tips**

* Combine with a **Loop Controller** for finer-grained control over repetition.
* Useful in **data-driven testing** with alternating CSV rows when paired with CSV Data Set Config.
* Keep child elements **stateless** (or isolate variable scopes) to avoid side effects between iterations.
* When testing **feature toggles or experiments**, this controller helps you simulate feature A/B behavior deterministically.