

---

---

# Simian Army

— Singleton —

---

---

# Singleton Class

- MonkeyRunner

```
public enum MonkeyRunner {  
  
    /** The instance. */  
    INSTANCE;  
  
    /** The Constant LOGGER. */  
    private static final Logger LOGGER = LoggerFactory.getLogger(MonkeyRunner.class);  
  
    /**  
     * Gets the single instance of MonkeyRunner.  
     *  
     * @return single instance of MonkeyRunner  
     */  
    public static MonkeyRunner getInstance() {  
        return INSTANCE;  
    }  
}
```

# Client

- BasicMonkeyServer

```
public class BasicMonkeyServer extends HttpServlet {  
    private static final Logger LOGGER = LoggerFactory.getLogger(BasicMonkeyServer.class);  
  
    private static final MonkeyRunner RUNNER = MonkeyRunner.getInstance();
```

```
@Override  
public void init() throws ServletException {  
    super.init();  
    configureClient();  
    addMonkeysToRun();  
    RUNNER.start();  
}
```

# Role of MonkeyRunner

- In charge of Monkeys

```
/**
 * Start all the monkeys registered with addMonkey or replaceMonkey.
 */
public void start() {
    for (Monkey monkey : monkeys) {
        LOGGER.info("Starting " + monkey.type().name() + " Monkey");
        monkey.start();
    }
}

/**
 * Stop all of the registered monkeys.
 */
public void stop() {
    for (Monkey monkey : monkeys) {
        LOGGER.info("Stopping " + monkey.type().name() + " Monkey");
        monkey.stop();
    }
}
```

# Enum Singleton

- Easy to write
- Lazy loaded
- Thread-safe
- Automatically handle serialization