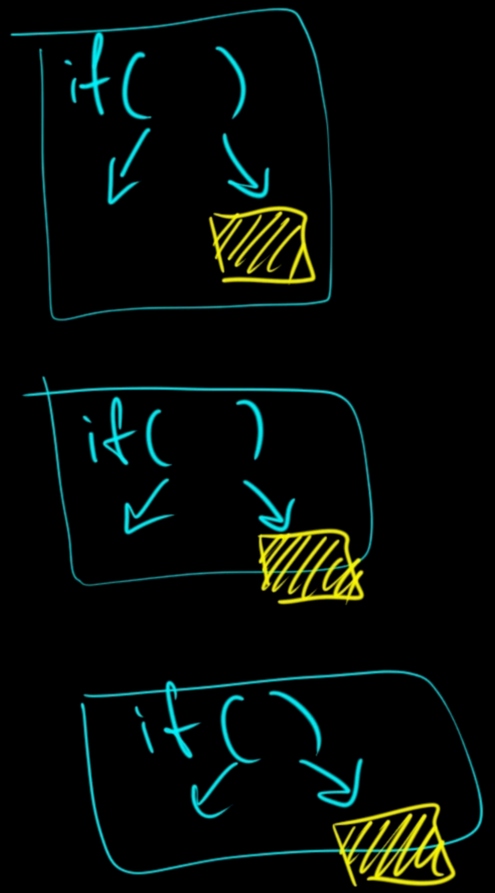
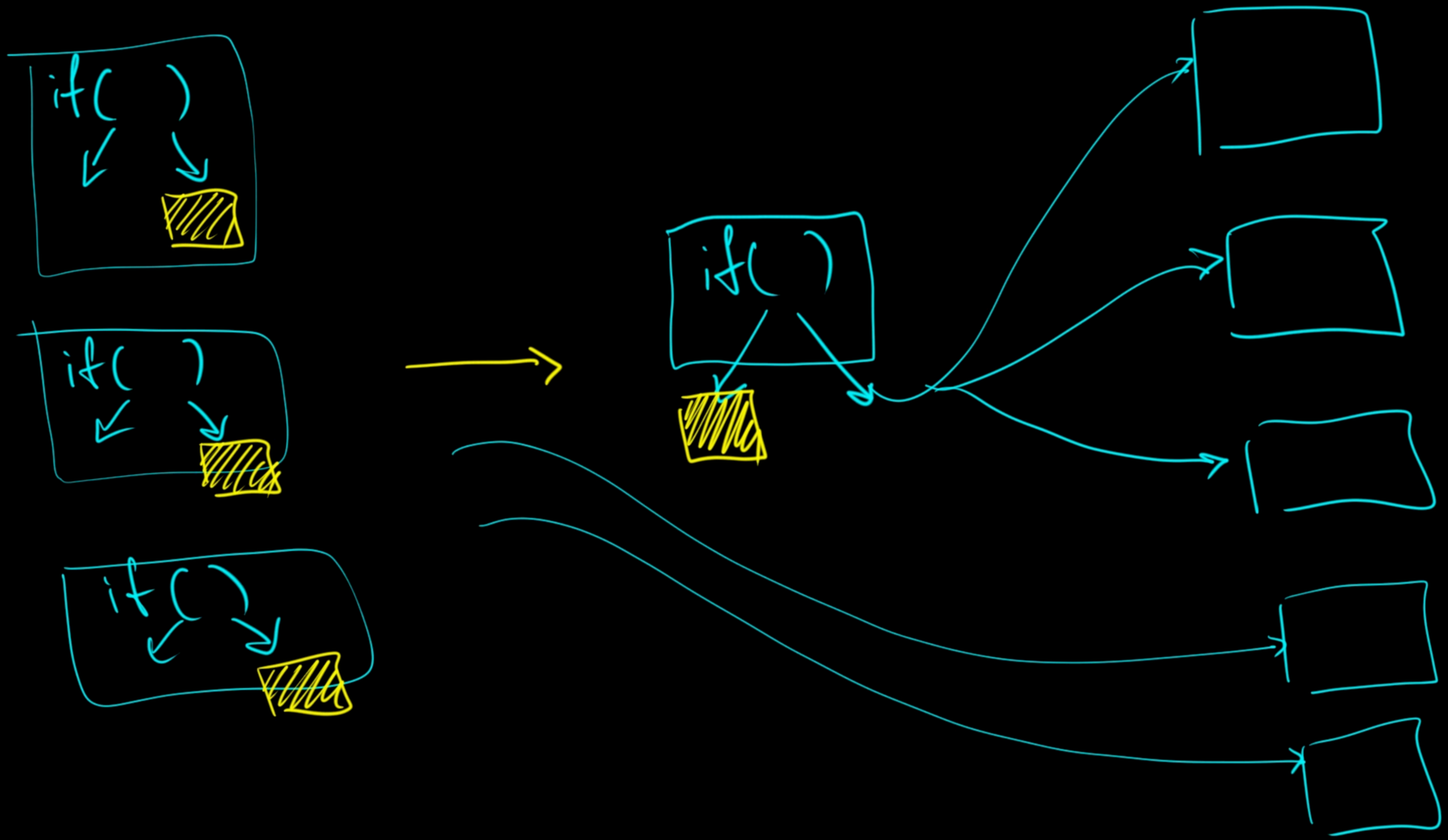
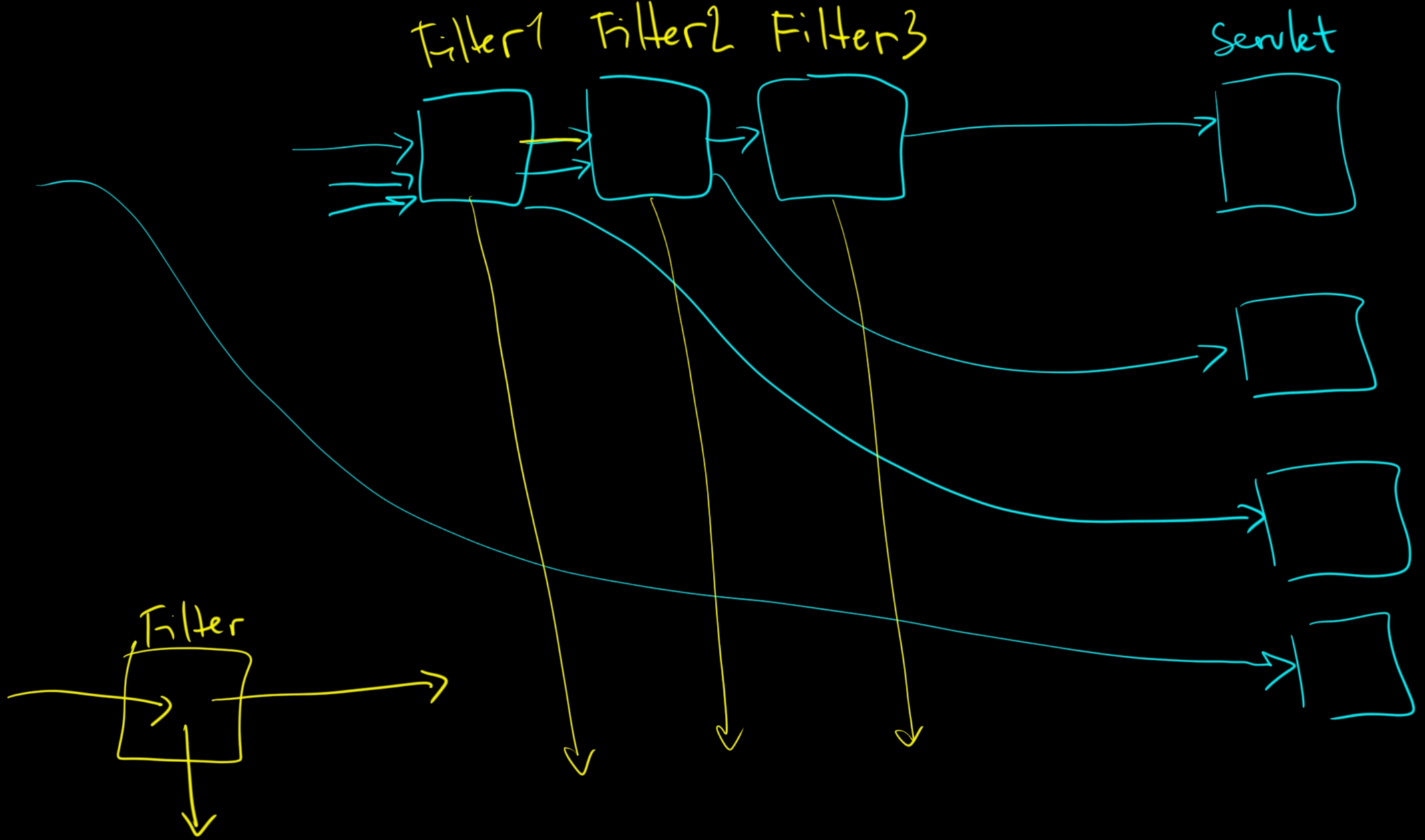


Not Good!

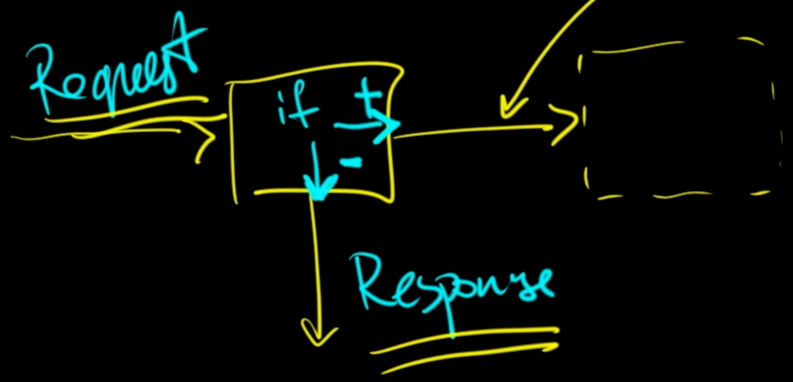

if( )
if( )
if( )
if( )

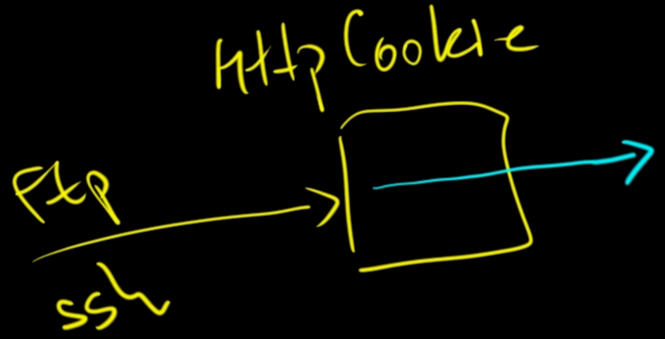

Filter1
Filter2
Filter3
Servlet
Filter


```
void doGet(HttpServletRequest req, HttpServletResponse resp)
```

```
public void doFilter(ServletRequest rq, ServletResponse rs, FilterChain chain)

}
```




Http Cookie
ftp
ssh

```java
public class MyCookieFilter implements Filter {
  @Override
  public void init(FilterConfig filterConfig) throws ServletException {
  }

  boolean isHttp(ServletRequest rq0, ServletResponse rs0) {
    return rq0 instanceof HttpServletRequest && rs0 instanceof HttpServletResponse
  }

  @Override
  public void doFilter(ServletRequest rq0, ServletResponse rs0, FilterChain chain)
    if (!isHttp(rq0, rs0)) {
      chain.doFilter(rq0, rs0);
    } else {
      HttpServletRequest rq = (HttpServletRequest) rq0;
      HttpServletResponse rs = (HttpServletResponse) rs0;

      if (Auth.getCookie(rq).isPresent()) {
        // OK
        chain.doFilter(rq0, rs0);
      } else {
        // ERROR
        rs.sendRedirect( s: "/login");
      }
    }
  }

  @Override
  public void destroy() {
  }
}
```

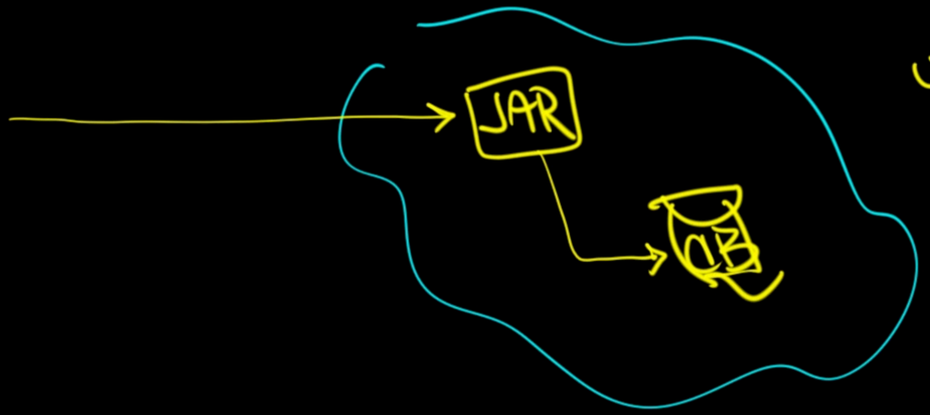
f: HttpServletRequest → Boolean

g: HttpServletRespons → void

```java
public class MyCookieFilter1 implements HttpFilter{
  @Override
  public boolean checkLogic(HttpServletRequest rq) {
    return Auth.getCookie(rq).isPresent();
  }

  @Override
  public void ifNotPassed(HttpServletResponse rs) throws IOException {
    rs.sendRedirect( s: "/login");
  }

}
```



url
username
passw

Environment

1) jar

2) docker