

1. Which elements are allowed in the `<filter-mapping>` element of the deployment descriptor? (Select three)
  - a `<servlet-name>`
  - b `<filter-class>`
  - c `<dispatcher>`
  - d `<url-pattern>`
  - e `<filter-chain>`

*Answers: a, c, and d*

**Explanation**

Answer a is correct because you can map filters to named servlets, as well as URL patterns. Answer c will control under which dispatching mechanism the filter is invoked. Answer d allows you to map the filter to an URL pattern. Answer b is a legitimate element, but it belongs in the `<filter>` element. Answer e is a nonexistent element.

2. What is wrong with the following code?

```

public void doFilter(ServletRequest req, ServletResponse res,
FilterChain chain)
throws ServletException, IOException {

    chain.doFilter(req, res);
    HttpServletRequest request = (HttpServletRequest)req;
    HttpSession session = request.getSession();
    if (session.getAttribute("login") == null) {
session.setAttribute("login", new Login());
    }
}
}

```

- a The doFilter() method signature is incorrect; it should take HttpServletRequest and HttpServletResponse.
- b The doFilter() method should also throw ServletException.
- c The call to chain.doFilter(req, res) should be this.doFilter(req, res, chain).
- d Accessing the request after chain.doFilter() results in an IllegalStateException.
- e Nothing is wrong with this filter.

*Answer: e*

### **Explanation**

Answers a and b are wrong; the doFilter() method's signature and thrown exceptions are correct. Answer c, calling this.doFilter(req, res, chain), would result in unwanted recursion. Answer d is incorrect; no code here will throw an IllegalStateException.

3. Given these filter mapping declarations:

```

<filter-mapping>
    <filter-name>FilterOne</filter-name>
    <url-pattern>/admin/*</url-pattern>
    <dispatcher>FORWARD</dispatcher>
</filter-mapping>
<filter-mapping>
    <filter-name>FilterTwo</filter-name>
    <url-pattern>/users/*</url-pattern>
</filter-mapping>
<filter-mapping>
    <filter-name>FilterThree</filter-name>
    <url-pattern>/admin/*</url-pattern>
</filter-mapping>
<filter-mapping>
    <filter-name>FilterTwo</filter-name>
    <url-pattern>/*</url-pattern>
</filter-mapping>

```

in what order are the filters invoked for the following browser request?

/admin/index.jsp

- a FilterOne, FilterThree
- b FilterOne, FilterTwo, FilterThree

- d** FilterThree, FilterTwo
- e** FilterThree
- f** None of these filters are invoked.

*Answer: d*

***Explanation***

FilterOne is cannot be invoked by calling /admin/index.jsp from the browser but only through a request dispatcher forward, so answers a and b are incorrect. Answer c lists the correct filters but in the wrong order. Answer e names only the first filter that is invoked. Answer d names the correct filters, in the correct order.