

## Chapter 06

- 1) In the \_\_\_\_\_ method of servlet connection is created.  
a) destroy() b)service() c) **init()** d) connect()
- 2) A servlet is an instance of \_\_\_\_\_  
a) **HTTPServlet class** b)Cookie c)HttpSessionBindingEvent d)HttpUtils
- 3) To work with servlet \_\_\_\_\_ is required  
a) **Java Development Kit** b)Java Server Kit c)Java Server Development Kit d)Java Servlet Development
- 4)Which method is called to process HTTP request?  
a) destroy() b) **service()** c) init() d)none of these
- 5)The server calls \_\_\_\_\_ method to relinquish any resources, such as files handles that are allocated for servlet.  
a)service() b)init() c)**destroy()** d) stop()
- 6) The life cycle of servlet is managed by \_\_\_\_\_  
a) servlet context b) **servlet container** c) supporting protocol(http or https) d) all of the above
- 7) The include() method of RequestDispatcher  
a) Sends a request to another resource b)**Include resource of file like servlet,jsp or html**  
c)Appends the request and response object to the current servlet d)None of the above
- 8) Servlet creates \_\_\_\_\_ thread for each request of client.  
a) **single** b)two c)multiple d) None of These
- 9)Which package contains the classes and interfaces required to build servlet?  
a)import javax.servlet.\*; b) import javax.servlet.http.\*; c)**both a&b** d) None of these
- 10)The \_\_\_\_\_ class provides functionality that makes it easy to handle requests and responses.  
a)**Generic Servlet** b)ServletInputStream c)ServletOutputStream d) None of these
- 11) The call to setContentType() establishes the MIME type of the HTTP response.

a) **True**    b) False

12) servletrunner utility tool listens on \_\_\_\_\_ port for incoming client requests.

a) 80    b)21    c) 53    **d)8080**

13)Which of the following interface declares the lifecycle method for servlet.

**a)Servlet**    b)Servlet Config    c) ServletContext    d) ServletResponse

14)\_\_\_\_\_ interface allows servlet to get initialization parameters.

a)SingleThreadModel    b)ServletRequest    **c)ServletConfig**    d)ServletContext

15) ServletContext interface enables servlets to log events and access information about their environment.

**a) True**    b) False

16) \_\_\_\_\_ interface is used to read data from a client request.

a)**ServletResponse**    b)ServletRequest    c) both a&b    d) None of these

17) \_\_\_\_\_ is responsible for managing execution of servlet

a) **Web Container**    b) Servlet Context    c) JVM    d) Server

18) GenericServlet class implements \_\_\_\_\_ and \_\_\_\_\_ interfaces.

a)ServletRequest and ServletResponse    b) ServletResponse and Servlet  
**c) Servlet and ServletConfig**    d)None of these

19) Which of the following class provides an input stream for reading requests from a client

a) ServletException    b)GenericServlet    c)ServletOutputStream    **d) ServletInputStream**

20) Which of the following class provides an output stream for writing response to the client

a) ServletException    b)GenericServlet    **c)ServletOutputStream**    d) ServletInputStream

21) \_\_\_\_\_ class indicates that a servlet error occurred.

**a) ServletException**    b) Unavailable Exception    c) GenericServlet    d) None of these

22) Which of the following interface allows a servlet to obtain configuration data when it is loaded

a) **ServletConfig** b) ServletContext c) ServletRequest d) None of these

23) ServletContextgetServletContext() returns the \_\_\_\_\_ for this servlet.

a) value b) **context** c) enumeration d) None of these

24) Which of the following method returns the value of the initialization parameter named param.

a) ServletContextgetServletContext() b) **String getInitParameter(String param)**

c) Enumeration getInitParameterNames() d)String getServerInfo()

25) Which of the following interface enables the servlet to obtain information about their environment?

a) **ServletContext** b) ServletConfig c)ServletRequest d) ServletResponse

26) Which of the following method returns the port number?

a)String getScheme() b)String getServerName() c) **int getServerPort()** d) String getRemoteHost()

27) \_\_\_\_\_ method returns the name of the server.

a)String getScheme() b)**String getServerName()** c) int getServerPort() d) String getRemoteHost()

28) \_\_\_\_\_ method returns the string equivalent of the client hostname.

a) **String getRemoteHost()** b)String getRemoteAddr() c) String getProtocol() d) None of these

29) Which method is used to specify before any lines that uses the PrintWriter?

a)setPageType() b) **setContentType()** c)setContextType() d)setResponseType()

30) In a web application, running in a web server, servlet is responsible for creating request and response object.

a) **True** b) False

31) Which of the following method Write s and stack the trace for e to the server log?

a) **void log(Exception e, String s)** b) void log( String s) c) void log() d) None of these

32) Which of the following method returns an enumeration with the name of servlets in the same namespace in the server?

a)String getInitParameter(String param) b) Enumeration getInitParameterNames()

( c) Enumeration getServletNames() d) None of these

33) \_\_\_\_\_ returns the real path that corresponds to the virtual path vpath.

a) String getServerInfo() b) String getMimeType(String File)

c) **String getRealPath(String vpath)** d) int getLength()

34) The \_\_\_\_\_ interface is used to indicate that only a single thread should execute the service() method of a servlet.

a) **SingleThreadModel** b) UnithreadModel c) ThreadModel d) None of These

35) If a servlet implements \_\_\_\_\_ interface, the server creates several instances of it.

a) ServletResponse Interface b) ServletRequest Interface c) **SingleThreadModel Interface** d) None of these

36) Which of the following class provides implementations of the basic life cycle methods for a servlet.

a) ServletInputStream Class b) **GenericServlet Class**

c) ServletException Class d) ServletOutputStream Class

37) void log(String s) method belongs to which of the following classes.

a) **GenericServlet Class** b) ServletOutputStream Class c) ServletInputStream Class d) None of These

38) Which of the following interface is a unique object per servlet?

a) **ServletConfig** b) ServletContext c) ServletRequest d) None of these

39) \_\_\_\_\_ is a unique object for complete application.

a) ServletConfig b) **ServletContext** c) ServletRequest d) None of these

40) We cannot set attributes in \_\_\_\_\_ interface that other servlets can use in their implementations.

A) **ServletConfig** b) ServletContext c) ServletRequest d) both a & b

41) We can set attributes in \_\_\_\_\_ that other servlets can use in their implementation.

ServletConfig b) **ServletContext** c) ServletRequest d) none of these

42) Which of the following interface is used to forward the request to another resource that can be HTML, JSP or another servlet in same application?

- a) **Request Dispatcher Interface**
- b) SingleThreadModel Interface
- c) ServletResponse Interface
- d) None of These

43) Which of the following method forwards the request from a servlet to another resource (servlet, JSP file, or HTML file) on the server?

- a) void include(ServletRequest request, ServletResponse response)
- b) **void forward(ServletRequest request, ServletResponse response)**
- c) void include(ServletRequest request)
- d) void forward(ServletRequest request)

44) PrintWriter is a character-stream class whereas ServletOutputStream\_\_\_\_\_ is a class.

- a) **byte-stream**
- b) bit-stream
- c) string-stream
- d) io-stream

45) \_\_\_\_\_ is protocol independent implementation of Servlet

- a) ServletInputStream
- b) ServletOutputStream
- c) **GenericServlet**
- d) None of These

46) Which of the following interface guarantees that no two threads will execute concurrently in the servlet's service method ?

- a) ServletResponse
- b) ServletRequest
- c) **SingleThreadModel**
- d) ServletConfig

47) Generic servlet is the super class of all servlets.

- a) **True**
- b) False

48) Servlet runs each request in a \_\_\_\_\_ ?

- a) OS shell
- b) JVM
- c) **Separate thread**
- d) JRE

49) GenericServlet class is encapsulated inside \_\_\_\_\_ package

- a) java.lang
- b) **javax.servlet**
- c) java.servlet
- d) javax.servlet.http



```

throws ServletException ,IOException{
Printwriter pw= response.getWriter();
Enumeration e= request.getParameterNames();
While(e.hasMoreElements()) {
String pname= (String) e.nextElement();
pw.print(pname + " = ");
Sting pvalue = request.getParameter(pname);
pw.println(pvalue);
}
pw.close();
}
}

```

a)import.java.util.\*; b) import javax.servlet.http.\*; c) import java.awt.\*; d)None of these

53) Write the missing statement in the below code

```

import javax.servlet.*;
public class WelcomeServlet extends GenericServlet{
public void service( ServletRequest request,ServletResponse response) throws ServletException
,IOException{
response.setContentType("text/html");
PrintWriter pw = response.getOutputStream();
pw.println("<b> Hello");
}
}

```

**a)pw.close()** b) pw.stop() c) pw.destroy() d) none of these

54) Which statement is missing in the place marked as \*\*\*\*\*

```

import java.io.*;
import javax.servlet.*;
public class First extends GenericServlet{
public void service(ServletRequest req,ServletResponse res) throws IOException,ServletException{
res.setContentType("text/html");
out.print("<html><body>");

```

```
out.print("<b>hello generic servlet</b>");
out.print("</body></html>");
}
}
```

- a) **PrintWriter out=res.getWriter();**      b) `PrintWriter in = res.getWriter()`  
c) `PrintWriter out=res.putWriter();`      d) `PrintWriter in = res.putWriter()`

55) Which of the following are interfaces in javax.servlet.http package?

- a) `HttpServletRequest` b)`HttpServletResponse` c)`HttpSession` **d)All of the above**

56) Which of the following informs an object that it is bound to or unbound from a session?

- a)`HttpServletRequest` b)`HttpServlet` c)`HttpSession` **d)HttpSessionBindingListener**

57) ----- allows session data to be read and written

- a)`HttpServletRequest` b)`HttpServlet` c)**HttpSession** d)`HttpSessionBindingListener`

58) ----- class provides methods to handle HTTP requests and responses

- a)HttpServlet** b)`Cookie` c)`HttpSessionEvent` d)None of the above

59) ----- class encapsulates session-changed event

- a)`HttpServlet` b)`Cookie` **c)HttpSessionEvent** d)None of the above

60) Which of the following code can be used to send an error response to the client using the specified status code and error message?

- a) `request.sendError(statusCode,message)` **b) response.sendError(statusCode,message)**  
c) `header.sendError(statusCode,message)` d) None of the above



61) The sendRedirect() method of HttpServletResponse interface can be used to redirect response to another resource, it may be servlet, jsp or html file.

**a. True** b. False

62) Which methods are used to bind the objects on HttpSession instance and get the objects?

a) setAttribute b)getAttribute **c. Both a & b** d. None of the above

63) getAuthType() method returns authentication scheme

**a. True** b. False

64) getHeaderNames() returns an enumeration of the header names

**a.True** b.False

65) ----- returns part of the URL that identifies the servlet

**a)getServletPath()** b)getPathInfo() c)getPathTranslated() d)None of the above

66) In getSession(boolean new) method If new is true and no session exists,

creates and returns a session for this request. Otherwise, returns the existing session for this request.

a. **True** b.False

67) Which of the following is not a method of HttpServletRequest interface.

**a) isRequestedSessionIdFromCookie()** b)getSession() c)getRequestURI()  
**d)addCookie(Cookie cookie)**

68) SC\_NOT\_FOUND indicates that the requested resource is not available.

**a.True** b.False

69) Determines if the session ID must be encoded in the URL identified as url. If so, returns the modified version of url. Otherwise, returns url.

- a) encodeRedirectURL(String url)   **b) encodeURL(String url)**   c) encode(String url  
d) None of the above

69) boolean containsHeader(String field) Returns true if the HTTP response header contains a field named field.

- a) False   **b) True**

70) ----- Returns true if the server created the session and it has not yet been accessed by the client.

- a) invalidate()   **b) isNew()**   c) getLastAccessedTime()   d) None of the above

71) Invalidate() method is in HttpSessionBindingListener interface

- a) True   **b) False**

72) Cookies are stored at server side.

- a) True   **b) False**

73) Some of the information that is saved for each cookie includes the following:

- a) The name of the cookie                      b) The value of the cookie  
c) The expiration date of the cookie            **d) All of the above**

74) Which method adds cookie to the HTTP response

- a) void addCookie()   **b) void addCookie(Cookie cookie)**  
c) void addCookie(String cookie)   d) void addCookie(int i)

75) If an expiration date is not explicitly assigned to a cookie, it is deleted when the current browser session ends. Otherwise, the cookie is saved in a file on the user's machine.

a) **True**    b)False

76) Constructor for cookie is

a) **Cookie(String name, String value)**    b) Cookie(String name, int value)

c) Cookie(Char name, String value)    d)None Of These

77) ----- method Returns true if the cookie must be sent using only a secure protocol. Otherwise, returns false.

a) **getSecure()**    b) getName()    c)clone()    d)None of these

78) The HttpServlet class extends GenericServlet.

a)False    **b)True**

79) Which of the following is not a method of HttpServlet class?

a)doDelete()    b)doGet()    c)doHead()    **d)getValue()**

80) Which method is Called by the server when an HTTP request arrives for this servlet.

a)getLastModifies()    **b)service()**    c)doPut()    d)None of the above

81) HttpSessionEvent encapsulates EventObject

a) **True**    b)False

82) ----- obtains the session to which the listener is being bound or unbound

a)getName() **b)getSession()** c)getSessionName() d)None Of the above

83) A servlet developer overrides which of the following methods?

a)doDelete() b)doGet() c)doHead() **d)All of the above**

84) ----- method returns true if the cookie contains session id.Otherwise returns false.

**a)Boolean isRequestedSessionIdFromCookie()**

b) Boolean isRequestedSessionId()

c) Boolean isSessionIdFromCookie()

d)None of the above

85) The servlet is invoked when a form on a Web page is submitted.

**a)True** b)False

86) Which method returns copy of this object?

a)getclone() **b)clone()** c)setclone() d)None of these

87) Which class allows state information to be stored on a client machine?

a) **Cookie** b)HttpServlet c)HttpSession d)None of these

88) Which method returns the URL?

a)getURL() b)URL() **c)getRequestURL()** d)None of the above

89) Method isRequestedSessionFromCookie() returns true if a cookie contains session id otherwise false

a)True b)False

90) Which method redirects the client to the URL?

a) **sendRedirect(String url)** b)Redirect(String url)

b) sendError(String url) d)None of the above

91) Which method adds field to the header with date value equal to msec?

a)**void setDateHeader(String field,long msec)** b) void setDateHeader(String field,int msec)

c) void setDateHeader(long msec) d) void setDate(String field,long msec)

92) Which method sets status code for this response to code

a)**void setStatus(int code)** b) void setStatus()

c) void Status(int code) d)None of the above

93) Which method returns the time when the client last made a request for this session

a)void getLastAccessedDate() **b)long getLastAccessedTime()**

c)getAccessedTime() d)None of the above

94) Which method performs an HTTP DELETE?

a) **void doDelete(HttpServletRequest req,HttpServletResponse res) throws IOException, ServletException**

b) void Delete(HttpServletRequest req,HttpServletResponse res) throws IOException, ServletException

c) void doDelete(HttpServletRequest req,HttpServletResponse res) throws IOException, ServletException

d) void doDelete(HttpServletRequest req,HttpServletResponse res) throws IOException, ServletException

95) Which method performs an HTTP GET?

a) void doGet(HttpServletRequest req, HttpServletResponse res) throws ServletException

**b) void doGet(HttpServletRequest req, HttpServletResponse res) throws IOException, ServletException**

c) void Get(HttpServletRequest req, HttpServletResponse res) throws IOException, ServletException

d) None of the above

96) Which method invalidates this session and removes it from the context?

a) **void invalidate()** b) void validate() c) void verify() d) void removeAttribute()

97) Which of the following is not a method of HttpSession interface?

a) getAttribute() b) setAttribute() **c) setHeader()** d) isNew()

98) Which of the following is a method of HttpServletResponse interface?

a) getAttribute() b) setAttribute() **c) setHeader()** d) isNew()

99) SC\_OK indicates that HTTP request succeeded.

a) **True** b) False

100) SC\_NOT\_OK indicates that requested resource is not available.

**a) True** b) False

101) Which of the following is true about cookies?

a) Cookies are stored on client b) Cookies contain state information

c) Cookies track user activities **d) All of the above**

102) Which of the following is not a method of cookie class?

- a) Clone() b) getMaxAge() **c) doGet()** d) getName()

103) Which one is not a constructor for cookie?

- a) Cookie() b) Cookie(String name) **c) a & b** d) Cookie(String name, String value)

104) Which of the following is not a method of HttpServlet Class?

- a) **setComment()** b) doDelete() c) doGet() d) doOptions()

105) HttpServlet class methods throw which exceptions?

- a) IOException b) ServletException c) IllegalStateException **d) a & b**

106) Which of the following is constructor for HttpSessionEvent Class?

- a) HttpSessionEvent() **b) HttpSessionEvent(HttpSession session)**

- c) HttpSessionEvent(String value) d) None of the above

107) Action parameter of tag specifies -----

- a) Location **b) URL** c) Path d) None of the above

108) ----- method returns any path information that is located after the servlet path and before a query string of the URL

- a) String getPathInfo()** b) String getPath()

- c) String getMethod() d) None of the above

109) ----- returns an array of the cookies in this request

- a) **Cookie[] getCookies()** b) Cookie[] getMaxCookies()  
c) Cookie[] getMinCookies() d) None of the above

110) Which method returns int equivalent of the header field named field?

- a) int getHeader() b) int getIntHeader() **c) int getIntHeader(String field)** d) None of these

111) ----- method returns name of the user who issues this request.

- a) **String getRemoteUser()** b) String getUser() c) String getRemote() d) None of these

112) A servlet can write a cookie to a user's machine via the \_\_\_\_\_ method of the HttpServletResponse interface.

- a) **addCookie()** b) Cookie() c) CookieDo() d) updateCookie()

113) A servlet can write a cookie to a user's machine via the addCookie() method of the \_\_\_\_\_ interface

- a) ServletRequest b) HttpServletRequest c) **HttpServletResponse** d) ServletResponse

114) Correct signature of Cookies is:

- a) Cookie(String value, String name) b) Cookie(int value, String name)  
c) **Cookie(String name, String value)** d) Cookie(int value, int name)

115) Information that is saved for each cookie includes the following:

- a) The name of the cookie, the value of the cookie b) The expiration date of the cookie  
c) The domain and path of the cookie d) **All of above**



116) Following is the correct syntax for creating Cookies Object c:

- a)Cookie c = new Cookie(MyCookie, data);
- b)**Cookie c = new Cookie("MyCookie", data);**
- c)cookie c = new cookie("MyCookie", data);
- d)Cookie c = new Cookie(data,"MyCookie");

117) Following method is used for Sets the maximum age of the cookie in seconds.

- a)public void setmaxage(int expiry)
- b)public void Setmaxage(String expiry)
- c)public void setMaxAge(String expiry)
- d)**public void setMaxAge(int expiry)**

118) Following is the method of HttpServletRequest interface is used to return all the cookies from the browser.

- a)Cookie[] getcookies()
- b)private Cookie[] getCookies()
- c) **public Cookie[] getCookies()**
- d)public Cookie[] setCookies()

119) Which one is following method indicates if secure protocol to be used while sending this cookie?

- a)**public void setSecure(boolean secure)**
- b) public void setsecure(int secure)
- c) private void setSecure(boolean secure)
- d)public boolean setSecure(boolean secure)

120) Following method is used for setting up comments in the cookie.

- a)public void setComment(int purpose)
- b)**public void setComment(String purpose)**
- c)private void setpath(String path)
- d)public void setPath(String path)

121) Following method is used to Specifies a path for the cookie to which the client should return the cookie.

- a)**public void setPath(String path)**
- b)public void setpath(String value)
- c)public void setPath(int path)
- d)private void setpath(String path)

122) Following method is returns the cookie protocol version.

- a) **int getVersion()**   b)int setversion()   c)int GetVersion()   d)String getVersion()

123) Following method is used to Sets the domain in which this cookie is visible.

- a)**public void setDomain(String pattern)**   b)public void setDomain(int pattern)  
c)public void getDomain(String pattern)   d)private void setdomain(String pattern)

124) \_\_\_\_\_ is a way to maintain state (data) of an user.

- a)**Session Tracking**   b)Cookie tracking   c)HttpServletState   d)Session

125) \_\_\_\_\_ encapsulates session events.

- a) httpsessionevent   b)**HttpSessionEvent**   c)HttpSessionTrackingEvent  
d)HttpSessionEventObject

126) Following method returns the session in which the event occurred.

- a)HttpSession setSession()   b)httpsession getsession()  
c)**HttpSession getSession()**   d)HttpSession setSessionEvent()

127) HTTP is a \_\_\_\_\_ protocol.

- a) **Stateless**   b)state oriented   c)stateful   d)datagram

128) A session can be created via the getSession() method of \_\_\_\_\_.

- a)HttpServletResponse   b)**HttpServletRequest**   c)HttpRequest   d)HttpResponse

129) In some applications, it is necessary to \_\_\_\_\_ so that information can be collected from several interactions between a browser and a server.

- a) save date and time information      b) save creation of session
- b) **save state information**              d) save objects

130) A session can be created via the \_\_\_\_\_ method of HttpServletRequest

- a) getSessionCreate( )    b) SetSession( )    c) setsession( )    d) **getSession( )**

131) The \_\_\_\_\_ method is overridden to process any HTTP POST requests that are sent to servlet.

- a) doGet( )    b) **doPost( )**    c) DoPost( )    d) DOGET( )

132) \_\_\_\_\_ interface enables a servlet to obtain information about a client request.

- a) **HttpServletRequest**    b) HttpServletResponse    c) httpServletRequest    d) Http Request

133) Identify the correct sequence of creating cookies.

a)

- 1) **Create a Cookie object.**
- 2) **Set the maximum Age.**
- 3) **Place the Cookie in HTTP response header.**

b)

- 1) Set the maximum Age.
- 2) Place the      Cookie in HTTP response header.
- 3) Create a Cookie object.

c)

- 1) Place the Cookie in HTTP response header.

2) Set the maximum Age.

3) Create a Cookie object.

d)

1) Set the maximum Age.

2) Create a Cookie object.

3) Place the Cookie in HTTP response header.

134) Following are the methods in connection with HttpSession interface.

a) String getId( ) b) void invalidate( ) c) long getLastAccessedTime( ) d) **All of above**

135) The HttpSession interface is implemented by the \_\_\_\_\_.

a) Session b) Cookies c) client d) **server**

136) A cookie is stored on a \_\_\_\_\_ and contains state information.

a) Session b) Cookies c) **client** d) server

137) Which class encapsulates a session-changed event?

a) Cookie b) HttpServlet c) **HttpSessionEvent** d) HttpSessionBindingEvent

138) Following method returns the time (in milliseconds since midnight, January 1, 1970, GMT) when this session was created.

a) int getcreationtime( ) b) long CreationTime( ) c) **long getCreationTime( )** d) long getCreation( )

139) Methods of HttpSession interface throw an \_\_\_\_\_ if the session has already been invalidated.

a) `IllegalState`   b) `IllegalException`   c) `LegalStateException`   d) **`IllegalStateException`**

140) Following method returns true if the requested session ID is valid in the current session context.

a) `boolean RequestedSessionIdValid()`   b) `boolean isRequestedSessionId()`  
c) `boolean isRequestedValid()`   d) **`boolean isRequestedSessionIdValid()`**

141) The Java \_\_\_\_\_ specification defines an application programming interface for communication between the Web server and the application program.

a) **Servlet**  
b) Server  
c) Program  
d) Randomize

142) The `doGet()` method in the example extracts values of the parameter's type and number by using \_\_\_\_\_

a) **`request.getParameter()`**  
b) `request.setParameter()`  
c) `responce.getParameter()`  
d) `responce.getAttribute()`

143) Dynamic interception of request and responses to transform the information is done by

a) `Servlet Container`   b) `Servlet config`   c) `Servlet context`   d) **`Servlet filter`**

144) State true or false.

i) `init()` is called after `start()` in applet  
ii) applets are used for networking  
iii) inheritance is a part of Java Foundation Classes  
iv) `final` does not prevent inheritance

a) i-true, ii-true, iii-false, iv-true  
b) **i-false, ii-false, iii-false, iv-false**

- c) i-true, ii-true, iii-true, iv-true
- d) i-true, ii-false, iii-false, iv-false

145) Identify Error in the following servlet code

```
import java.io.*;

import javax.servlet.*;

public class ColorGetServlet extends HttpServlet {

    public void doGet(HttpServletRequest request,HttpServletResponse response) throws
ServletException, IOException {

String color = request.getParameter("color"); response.setContentType("text/html"); PrintWriter
pw = response.getWriter(); pw.println("<B>The selected color is: "); pw.println(color);

pw.close();

    }

}
```

Answer: Package javax.servlet.http.\* not imported

146) Find error in the following code

```
import java.io.*;

import javax.servlet.*;

import javax.servlet.http.*;

public class ColorPostServlet extends HttpServlet

{

public void doPost(HttpServletRequest request,HttpServletResponse response) throws
ServletException, IOException

{

String color = request.getParameter("color");

PrintWriter pw = response.getWriter();

pw.println("<B>The selected color is: ");
```

```
pw.println(color);  
pw.close();  
}  
}
```

Answer:

Error: response.setContentType("text/html"); missing

145) Which is the missing statement in following code to get proper output:

```
import java.io.*;  
import java.util.*;  
public class HelloWorld extends HttpServlet  
{  
    public void doGet(HttpServletRequest request, HttpServletResponse response)  
        throws IOException, ServletException  
    {  
        response.setContentType("text/html");  
        PrintWriter out = response.getWriter();  
        HttpSession hp=request.getSession(true);  
        Date create=new Date(hp.getCreationTime());  
        Date access=new Date(hp.getLastAccessedTime());  
        out.println("<h1><br>Session is created at "+create);  
        out.println("<h1><br>Last access time is "+access);  
        out.println("<h1><br>Session id is "+hp.getId());  
    }  
}
```

a)Missing ;

b)Missing()

**c)Missing Package Statement**

d)Missing {

146) What is the missing statement in the following program to get the proper output

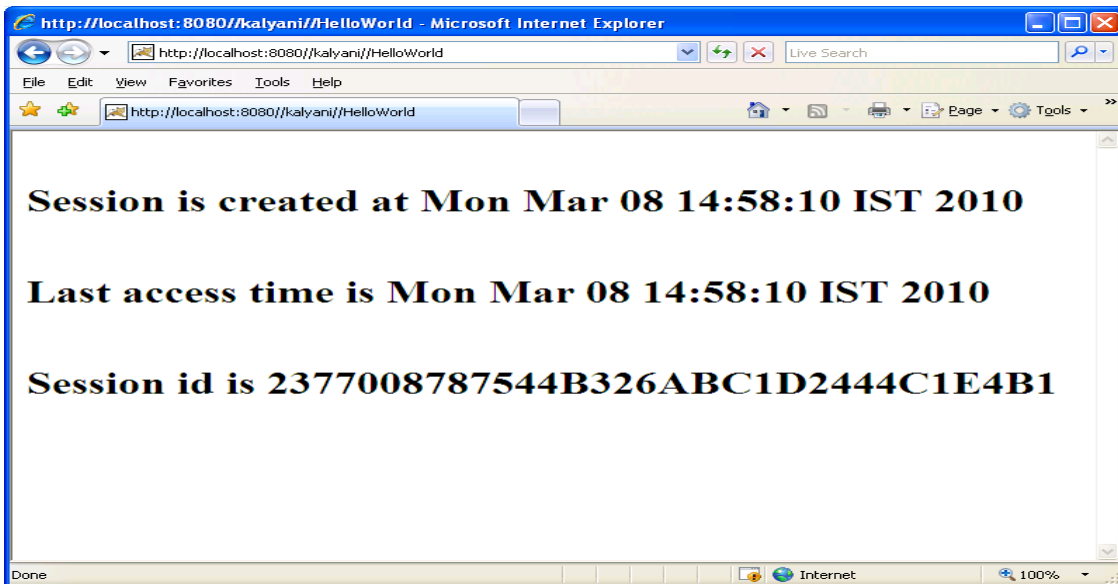
```
import java.io.*;  
import java.util.*;  
public class HelloWorld extends HttpServlet
```

```

{
public void doGet(HttpServletRequest request, HttpServletResponse response)
throws IOException, ServletException
{
response.setContentType("text/html");
PrintWriter out = response.getWriter();
HttpSession hp=request.getSession(true);
Date create=new Date(hp.getCreationTime());
Date access=new Date(hp.getLastAccessedTime());
out.println("<h1><br>Session is created at "+create);
out.println("<h1><br>Last access time is "+access);
out.println("<h1><br>Session id is "+hp.getId());
}
}

```

/\*OUTPUT\*/



- a)Missing semicolon
- b)missing {}
- c)missing Package statement
- d)none of the above

147) What is at the place of \_\_\_\_\_ to get output

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

```



```

public class example extends _____
{
public void doGet(HttpServletRequest request, HttpServletResponse response)throws
IOException, ServletException
{
response.setContentType("text/html");
PrintWriter out = response.getWriter();
// print out cookies
String name = request.getParameter("t1");
String value = request.getParameter("t2");
Cookie c = new Cookie(name, value);
response.addCookie(c);
String n1 = c.getName();
String v1 = c.getValue();
out.println("<h3> The Name Of Cookie is:</h3>" +n1);
out.println("<h3> The Value Of Cookie is:</h3>" +v1);
}
}

```

- a) **HttpServlet**  
b) **HttpServletResponse**  
c) **HttpServletRequest**  
d) All of the above

148) What we should write at the place of \_\_\_\_\_ to get the proper output.

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class password extends HttpServlet
{
public void doGet(HttpServletRequest request, HttpServletResponse response)throws
IOException, ServletException
{
response.setContentType("text/html");
PrintWriter out = _____
out.println("<html>");
out.println("<head>");
out.println("<title>Hello World!</title>");
out.println("</head>");
out.println("<body>");
}
}

```

```
String a= request.getParameter("t1");
String b= request.getParameter("t2");
int a1 =a.length();
int b1= b.length();
if(b.length()<=6)
{
out.println("<h3> Welcome To Home Page<h3>");
}
else
{
out.println("<h3>Password Should not more than 6 Character<h3>");
}
out.println("</body>");
out.println("</html>");
}
}
a)request.getWriter();
b)response.getWriter();
c)response.printWriter();
d)none of the above
```

149) What should we write at the place of \_\_\_\_\_ to get the proper output.

```
import java.io.*;
import javax.servlet.*;
public class GenericServletDemo extends GenericServlet
{
public void service(ServletRequest request , _____ response) throws
ServletException,IOException
{
response.setContentType("text/html");
PrintWriter out=response.getWriter();
out.println("<HTML>");
out.println("<HEAD><TITLE>hello page</TITLE></HEAD>");
out.println("<BODY>");
String name=(String) request.getParameter("user1");
String password=(String) request.getParameter("password1");
String phone=(String) request.getParameter("phone1");
out.println("Name="+name+"<br>");
out.println("Passsword="+password+"<br>");
out.println("Phone="+phone+"<br>");
out.println("</BODY></HTML>");
}
}
```

```
}
```

- a)HttpReuest
- b) ServletResponse**
- c)HttpResponse
- d)none of the above

150) Choose correct package for following code from given options.

```
import java.io.*;
import javax.servlet.*;
public class HelloServlet extends ***** {
public void service(ServletRequest request,
ServletResponse response)
throws ServletException, IOException
{
response.setContentType("text/html");
PrintWriter pw = response.getWriter();
pw.println("<B>Hello!");
pw.close();
}
}
```

- a) GenericServlet**
- b) genericServlet
- c) HttpServlet
- d) httpServle

151) In following Java program fill statement showing \*\*\*\*\*..Select any one option from given options

```
import java.io.*;
import java.util.*;
import javax.servlet.*;

public class PostParametersServlet
extends GenericServlet {
public void service(ServletRequest request,
ServletResponse response)
throws ServletException, IOException {
// Get print writer.
PrintWriter pw = response.getWriter();
```

```
// Get enumeration of parameter names
```

```

Enumeration e = request.getParameterNames();
;
// Display parameter names and values.
while(e.hasMoreElements()) {
String pname = (String)e.nextElement();
pw.print(pname + " = ");
String pvalue = *****;
pw.println(pvalue);
}
pw.close();
}
}

```

- a) **request.getParameterName(pname);**
- b) request.Getparametername(pname);
- c) Request.Getparametername(pname);
- d) request.getParametername(pname);

152) Consider the following program and identify the missing method attribute \*\*\*\*\* if you want to implement HTTP POST.

```

<html>
<body>
<center>
<form name="Form1"
method=*****
action="http://localhost:8080/examples/servlet/ColorPostServlet">
<B>Color:</B>
<select name="color" size="1">
<option value="Red">Red</option>
<option value="Green">Green</option>

<option value="Blue">Blue</option>
</select>
<br><br>
<input type="submit" value="Submit">
</form>
</body>
</html>

```

- a) HTTP POST
- b) HTTP GET

- c)post
- d)get

153) Consider the following program. Identify the missing word \*\*\*\*.

```
<html>
<body>
<center>
<form name="Form1"
action="http://****/examples/servlet/ColorGetServlet">
<B>Color:</B>
<select name="color" size="1">
<option value="Red">Red</option>
<option value="Green">Green</option>
<option value="Blue">Blue</option>
</select>
<br><br>
<input type="submit" value="Submit">
</form>
</body>
</html>
```

- a)localhost:8080
- b) localhost:0808
- c) locallhost:8080
- d) localhost:80800

154) Consider the following program and identify the missing method name \*\*\*\*\* if you want to implement HTTP POST.

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class ColorGetServlet extends HttpServlet {
public void *****(HttpServletRequest request,
HttpServletResponse response)
throws ServletException, IOException {
String color = request.getParameter("color");
response.setContentType("text/html");
PrintWriter pw = response.getWriter();
pw.println("<B>The selected color is: ");
pw.println(color);
pw.close();
}
}
```

- a)doGet()
- b)doPost()
- c)HTTP GET()
- d)HTTP POST()