問1　Web システムのクロスサイトスクリプティング対策に関する次の記述を読んで，設問1，2に答えよ。

E 社は，提携店舗数 100 店の賃貸不動産仲介業者である。E 社では，3 年前に開設した Web システム（以下，E システムという）を用いて賃貸物件情報を提供している。E システムの運用はシステム部が担当し，E システムの保守とコンテンツの開発は E システムの開発を担当した B 社に委託している。

E 社は，E システムをインターネットに公開することを踏まえて，IPA が公表している“安全なウェブサイトの作り方”のチェックリストを参考にしてセキュリティ対策の実施項目を作成し，その実施項目に従って Java で開発するよう B 社に依頼していた。対策の実施項目のうち，クロスサイトスクリプティング（以下，XSS という）対策の実施項目を，表1に示す。

表1　XSS 対策の実施項目

項番	実施項目
①	HTML テキストの入力を許可しない仕様とする。
②	Web ページに出力する全ての要素に対して，エスケープ処理を施す。
③	URL を出力するときは，“http://”又は“https://”で始まる URL だけを許可する。
④	スクリプト要素の内容（“<script>…</script>”の“…”部分）を動的に生成しない。
⑤	スタイルシートを任意のサイトから取り込めるようにしない。
⑥	HTTP レスポンスヘッダの Content-Type フィールドに文字コード（charset）を指定する。

〔社内検査〕

E システム開発後，他社の Web サイトで XSS 対策漏れが原因の事故が多発したことから，E システムにも問題が潜在していることが懸念された。そこで容易に検査できる方法はないかと探していたところ，基本的な XSS 対策を実施しているかどうかを診断する“ウェブ健康診断仕様”（以下，Web 健康診断という）が，“安全なウェブサイトの作り方”の別冊として IPA から公表されているとの情報を得た。表2は Web 健康診断に基づいて作成した XSS 対策の判定基準である。システム部の C 部長は，表2の内容であれば，社内で検査できると考え，若手の D 君を担当に指名した。

表2の判定基準では，“対象画面”の“検査文字列を入力する場所”に“入力する検査文字列”を入力した場合に，“脆(ぜい)弱性ありと判定する基準”で示された基準で脆弱性ありと判定する。

表２　Web 健康診断に基づいて作成した XSS 対策の判定基準

検出パターン	対象画面	検査文字列を入力する場所	入力する検査文字列	脆弱性ありと判定する基準
1	・入力内容確認画面 ・エラー画面	GET パラメタ及び POST パラメタ	'>"><hr>	HTTP レスポンスボディに検査文字列がエスケープ処理などを行われずに出力される場合，脆弱性ありと判定
2			'>"><script>alert(document.cookie)</script>	
3		URL 中最後の“/”に続く文字列	<script>alert(document.cookie)</script>	同上。例えば，URL が“http://www.aaaa.jp/bbbb/index.html”で，“index.html”の部分に検査文字列をエンコードせずに挿入したときエスケープされずに出力される場合，脆弱性ありと判定
4		GET パラメタ及び POST パラメタ	javascript:alert(document.cookie);	HTTP レスポンスボディの特定の URI 属性（src, action, background, href, content）に検査文字列が出力される場合，脆弱性ありと判定

図１は E システムのうち，賃貸物件の選択結果を表示するプログラムである。D 君が表 2 の判定基準に基づいて検査をしたところ，検出パターン [a] の検査で脆弱性ありと判定された。その原因箇所は図 1 のプログラムの [b] 行目であった。図１に，他の脆弱性ありと判定されたものはなかった。

```
（省略）　〔package, import 宣言など〕
 1 public class SelectWin extends HttpServlet {
 2    public void doGet(HttpServletRequest request, HttpServletResponse response)
 3        throws IOException, ServletException
 4    {
 5        response.setContentType("text/html; charset=¥"UTF-8¥"");
 6        request.setCharacterEncoding("UTF-8");
   /* パラメタ name（物件名称），loc（地区名）及び url（URL 情報）を取得 */
 7        String rqName = request.getParameter("name");
 8        String rqLoc = request.getParameter("loc");
（省略）　〔rqName,  rqLoc の文字化け対策〕
 9        String rqUrl = request.getParameter("url");
   /* 画面の HTML を出力 */
10        PrintWriter out = response.getWriter();
11        out.println("<html>");
12        out.println("<head>");
13        out.println("<title>賃貸物件の選択結果</title>");
14        out.println("</head>");
15        out.println("<body>");
16        out.println("<h3>詳細表示を開く</h3>");
17        out.println("<p>");
   /* 選択された賃貸物件の詳細を開くリンクを作成 */
18        out.print  ("<a href=¥"" + escapeHTML(rqUrl) + "¥">");
19        out.print  (" 【" + escapeHTML(rqLoc) + "】 ");
20        out.println(escapeHTML(rqName) + "</a>");
21        out.println("</body>");
22        out.println("</html>");
23    }
（省略）　〔その他の必要なメソッドなど〕
```

注記　escapeHTML は HTML の特殊文字 5 文字（<, >, &, ", '）のエスケープ処理を行うメソッドとして別途定義されている。

図１　賃貸物件の選択結果を表示するプログラム

〔詳細な診断の受診〕

表 2 の判定基準に基づいた社内検査で脆弱性が検出されたことから，C 部長は詳細な診断を受ける必要があると判断し，セキュリティ専門会社の F 社に詳細な診断を依頼した。F 社では，インターネット経由でのアクセスによる E システムの診断とサーバプログラムのソースコードレビューを行った。F 社による診断結果を，図 2 に示す。

XSS 対策に関する診断結果

この脆弱性は，表 1 の実施項目①～⑥が全て正しく実施済みであれば発生しない。
確認した範囲で現状の実施状況をまとめると次のとおりである。

実施項目	実施状況	備考
①	合格	画面からの入力で HTML テキストを必要としているものも，許可しているものもない。
②	一部不合格	社内検査の検査対象に関してはエスケープ処理が施されている。しかし，社内検査の検査対象以外に，対策漏れがある。
③	不合格	URI 属性出力に対しては制限が必要である。
④	不合格	スクリプト要素の内容を動的に生成している。命令として解釈される可能性がある。
⑤	合格	E システム以外のスタイルシートは使用されておらず，任意のサイトから取り込めるようになっていない。
⑥	合格	文字コードは全て指定されている。

図 2　F 社による診断結果（抜粋）

〔プログラムの修正〕

図 3 の賃貸物件の検索画面を表示するプログラムでは，賃貸物件を検索するキーワードと地区名を設定するための画面を表示する。その次に呼び出される図 4 の賃貸物件の検索結果を表示するプログラムでは，そのキーワードと地区名を用いて賃貸物件を検索し，その結果をオプションメニューで表示する。オプションの選択が変更された際に，表示用に用意されているテキスト領域に，賃貸物件の情報を“【地区名】説明”の形式で表示する。選択ボタンが押下されると，図 1 のプログラムが呼び出される。

F 社による診断結果に基づき，図 3 と図 4 のプログラムを詳しく検討し，次のとおり問題点を整理して修正することとした。

(1)　図 3 の賃貸物件の検索画面を表示するプログラムの問題点と修正

問題点：［ c ］行目と［ d ］行目では，表 1 の実施項目［ e ］に該当する対策が 行われていなかった。社内検査では，検査対象を

[f] に限定しているのでD君はこの脆弱性を検出できなかった。

修正　：エスケープ処理が必要であり，定義されているメソッド [g] を適用すべきである。

```
(省略)     〔package, import 宣言など〕
 1 public class SelectKey extends HttpServlet {
 2    public void doGet(HttpServletRequest request, HttpServletResponse response)
 3        throws IOException, ServletException
 4    {
 5        String rsKey, rsLoc;
   /* rsKey にデータベースに登録されたキーワードで検索回数が一番多いものを取得
      rsLoc にデータベースに登録された選択候補地区名を取得，選択候補地区名の値は運用者が設定 */
 6        response.setContentType("text/html; charset=¥"UTF-8¥"");
 7        request.setCharacterEncoding("UTF-8");
   /* 画面の HTML を出力 */
 8        PrintWriter out = response.getWriter();
 9        out.println("<html>");
10        out.println("<head>");
11        out.println("<title>賃貸物件の検索</title>");
12        out.println("</head>");
13        out.println("<body>");
14        out.println("<h3>検索キーワードの設定</h3>");
(省略)     〔データベースに登録されたキーワードで検索回数が一番多いものを rsKey として取得〕
15        out.println("<br>");
16        out.println("<form action=¥"SelectURL¥" method=¥"GET¥">");
17        out.println("よく用いられるキーワード：" + rsKey);
18        out.println("<br>");
19        out.println("検索に用いるキーワードを入力してください：");
20        out.println("<input type=¥"text¥" size=¥"20¥" name=¥"key¥">");
21        out.println("<br>");
22        out.println("物件の地区名を選択してください：");
23        out.println("<select name=¥"loc¥">");
(省略)     〔データベースから選択候補地区名を検索
           検索結果から選択候補地区名を rsLoc として繰返し取得〕
24         out.println("<option value=¥"" + rsLoc + "¥">" + rsLoc + "</option>");
(省略)     〔繰返しはここまで〕
25        out.println("</select>");
26        out.println("<br>");
27        out.println("<input type=¥"submit¥" value=¥"選択¥">");
28        out.println("</form>");
29        out.println("</body>");
30        out.println("</html>");
31    }
(省略)     〔その他の必要なメソッドなど〕
```

図3　賃貸物件の検索画面を表示するプログラム

(2)　図4の賃貸物件の検索結果を表示するプログラムの問題点と修正

問題点：[h] 行目では，表1の実施項目 [i] に該当する対策が不十分であった。

修正　：[h] 行目を [j] に修正すべきである。

```
(省略)    〔package, import 宣言など〕
1 public class SelectURL extends HttpServlet {
2    public void doGet(HttpServletRequest request, HttpServletResponse response)
3        throws IOException, ServletException
4    {
5        response.setContentType("text/html; charset=¥"UTF-8¥"");
6        request.setCharacterEncoding("UTF-8");
  /* パラメタ key（キーワード）と loc（地区名）を取得 */
7        String rqKey = request.getParameter("key");
8        String rqLoc = request.getParameter("loc");
(省略)    〔rqKey, rqLoc の文字化け対策〕
  /* データベースから物件名称（rsName), URL 情報（rsUrl）及び説明（rsText）を取得
     これらの値は運用者が設定 */
9        String rsName, rsUrl, rsText;
(省略)    〔データベースから rqLoc と rqKey の値を用いて賃貸物件を検索〕
  /* 画面の HTML を出力 */
10       PrintWriter out = response.getWriter();
11       out.println("<html>");
12       out.println("<head>");
13       out.println("<title>賃貸物件の検索結果</title>");
14       out.println("<script type=¥"text/javascript¥">");
15       out.println("<!--");
  /* 選択ボタン押下時に選ばれている select オプションの値を変数 url と name の値に設定 */
16       out.println("function setValue() {");
17       out.println("  var index = document.form1.menu1.selectedIndex;");
18       out.print  ("  document.form1.url.value = ");
19       out.println("document.form1.menu1.options[index].value;");
20       out.print  ("  document.form1.name.value = ");
21       out.println("document.form1.menu1.options[index].label;");
22       out.println("}");
  /* オプションの選択が変更された際に msg1 のテキスト領域に“【地区名】説明”として表示 */
23       out.println("function onSet() {");
24       out.println("  var index = document.form1.menu1.selectedIndex;");
25       out.print  ("  document.form1.msg1.value = ¥"【¥" + ");
26       out.print  ("¥"" + escapeHTML(rqLoc) + "¥"");
27       out.println(" + ¥"】¥" +  document.form1.menu1.options[index].text;");
28       out.println("}");
29       out.println("// -->");
30       out.println("</script>");
31       out.println("</head>");
32       out.println("<body>");
33       out.println("<h3>詳細を表示する物件の選択</h3>");
34       out.println("<br>");
35       out.println("キーワード【" + escapeHTML(rqKey) + "】");
36       out.println("<br>");
37       out.println("<form name=¥"form1¥" action=¥"SelectWin¥" method=¥"GET¥">");
38       out.println("<input type=¥"text¥" size=¥"50¥" name=¥"msg1¥" readonly>");
39       out.println("<br>");
40       out.println("<select name=¥"menu1¥" onChange=¥"onSet()¥">");
(省略)    〔検索結果から物件名称（rsName), URL 情報（rsUrl）及び説明（rsText）を繰返し取得〕
41         out.print  ("<option label=¥"" + escapeHTML(rsName) + "¥" value=¥"");
42         out.println(escapeHTML(rsUrl) + "¥">" + escapeHTML(rsText) + "</option>");
(省略)    〔繰返しはここまで〕
43       out.println("</select>");
44       out.println("<br>");
45       out.println("<input type=¥"hidden¥" name=¥"url¥" value=¥"¥">");
46       out.println("<input type=¥"hidden¥" name=¥"name¥" value=¥"¥">");
47       out.print  ("<input type=¥"hidden¥" name=¥"loc¥" value=¥"");
48       out.println(escapeHTML(rqLoc) + "¥">");
49       out.println("<input type=¥"submit¥" value=¥"選択¥" onclick=¥"setValue()¥">");
```

図４　賃貸物件の検索結果を表示するプログラム

```
50        out.println("</form>");
51        out.println("</body>");
52        out.println("</html>");
（省略）     〔検索に使用したキーワード rqKey の値をデータベースに登録〕
53    }
（省略）     〔その他の必要なメソッドなど〕
```

図 4　賃貸物件の検索結果を表示するプログラム（続き）

E 社は，今回の XSS 対策以外の対策も含めて E システムの修正を B 社に依頼し，修正後，再度 F 社の診断を受診し，問題が解決していることを確認した。

設問 1　〔社内検査〕について，(1)～(4)に答えよ。

(1)　表 2 の検出パターン 1～4 は，それぞれ表 1 の②～⑥のどの実施項目の不備を検出できるものか。それぞれに該当する最も適切な項番を答えよ。

(2)　本文中の [a] に入れる適切な検出パターンを 1～4 から選び答えよ。

(3)　本文中の [b] に入れる適切な行番号を答えよ。

(4)　図 1 のプログラムの [b] 行目に対して実施する XSS 対策を 40 字以内で述べよ。

設問 2　〔プログラムの修正〕について，(1)～(7)に答えよ。

(1)　本文中の [c]，[d] に入れる適切な行番号を答えよ。

(2)　本文中の [e] に入れる最も適切な実施項目を表 1 の②～⑥から選び答えよ。

(3)　本文中の [f] に入れる適切な字句を，表 2 中の字句を用いて 20 字以内で述べよ。

(4)　本文中の [g] に入れる適切なメソッド名を答えよ。

(5)　本文中の [h] に入れる適切な行番号を答えよ。

(6)　本文中の [i] に入れる最も適切な実施項目を表 1 の②～⑥から選び答えよ。

(7)　本文中の [j] に入れる適切な修正後の 1 行のソースコードを答えよ。