

更に上のクオリティ
更に上のサービス!

問題集

ITEXAMPASS

<https://www.itexampass.jp>



1年で無料進級することに提供する

Exam : 070-480J

**Title : Programming in HTML5 with
JavaScript and CSS3**

Version : DEMO

1. トピック 1、ボリューム A

ドラッグドロップ

JavaScript と正規表現を使用してユーザー入力を検証しています。

定義済みの正規表現のグループは、2つの入力フィールドを検証します。

- validateEmail という名前の関数の電子メールアドレス（たとえば、`firstname@contoso.com`）
- validateSSN という名前の関数（たとえば、`555555555` または `555-55-5555`）で 2 番目と 5 番目の文字の後にオプションのハイフンを使用できる 9 桁の数字。

正しい式を使用して入力を検証する必要があります。

どの関数を各関数に挿入しますか？（答えるには、適切な正規表現ステートメントを正しい場所にドラッグします。各正規表現ステートメントは、1 回、複数回、またはまったく使用されないことがあります）コンテンツを表示するには、分割バーをベインまたはドラッグしてドラッグする必要があります。

```
/^d{1,2}(\-|\/1\.)\d{1,2}\1\d{4}$/;
```

```
/^(0-9){3}\^(0-9){2}\^[0-9]{4}$/;
```

```
/^(0-9){0}\^[0-9]{2}\^[0-9]{4}$/;
```

```
/^[a-zA-Z0-9._-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,4}$/;
```

```
/^[0-9]{3}+@[a-zA-Z0-9._-]+\.[0-9]{2,4}$/;
```

Answer Area

```
function validateSSN(userinput) {
    var ssnPattern =
    return ssnPattern.test(userinput);
}
function validateEmail(userinput) {
    var emailPattern =
    return emailPattern.test(uerirput);
}
```

Answer:

```
/^d{1,2}(\-|\/|1\.)\d{1,2}\1\d(4)$/;
```

```
 /^(0-9){0}\^[0-9]{2}\^[0-9]{4}$/;
```

```
 /^[0-9]{3}+@[a-zA-Z0-9._-]+\.[0-9]{2,4}$/;
```

Answer Area

```
function validateSSN(userinput) {  
    var ssnPattern =  
        /^[0-9]{3}\^[0-9]{2}\^[0-9]{4}$/;  
    return ssnPattern.test(userinput);  
}  
function validateEmail(userinput) {  
    var emailPattern =  
        /^[a-zA-Z0-9._-]+@[a-zA-Z0-9._-]+\.[a-zA-Z]{2,4}$/;  
    return emailPattern.test(uerirput);  
}
```

2. 次の HTML を含む顧客 Web フォームを開発しています。

```
<input id="txtValue" type="text" />
```

顧客は、フォームを送信する前にテキストボックスに有効な年齢を入力する必要があります。

コントロールに検証を追加する必要があります。

どのコードセグメントを使用しますか？

- A.

```
function validate(){
  var value = $("txtValue").text();
  var regex = /^[d\,\.]*$/;
  if (!regex.test(value) || name == "")
    alert("please enter valid value");
  return;
}
```
- B.

```
function validate(){
  var name = $("#txtValue").val();
  if (name == null || name == "")
    alert("please enter valid value");
  return;
}
```
- C.

```
function validate(){
  var name = $("#txtValue").text();
  if (name == null || name == "")
    alert("please enter valid value");
  return;
}
```
- D.

```
function validate() (
  var value = $("#txtValue").val();
  var regex = /^[d\,\.]*$/;
  if (!regex.test(value) || value == "")
    alert("please enter valid value");
  return;
}
return emailPattern.test(userinput);
}
```

- A. オプション A
B. オプション B
C. オプション C
D. オプション D

Answer: D

3. あなたは会社のウェブサイトのページに表示される顧客連絡フォームを開発しています。このページは顧客に関する情報を収集します。

顧客がフォームを送信する前に値を入力する場合は、有効な電子メールアドレスでなければなりません。データ検証要件が満たされていることを確認する必要があります。

あなたは何を使うべきですか？

- A. `<input name="email" type="url"/>`
B. `<input name="email" type="text" required="required"/>`
C. `<input name="email" type="text"/>`
D. `<input name="email" type="email"/>`

Answer: D

4. ドラッグドロップ

HTML5 と jQuery を使用してユーザーの電子メールアドレスを取得するフォームを開発しています。フォームは電子メールアドレスを取得し、それをクエリ文字列パラメータとして返す必要があります。クエリ文字列パラメータは、電子メールアドレスで使用される@記号を表示する必要があります。この機能を実装する必要があります。

どのようにフォームを開発する必要がありますか？（答えを出すには、適切なコードセグメントを回答エリア内の正しいターゲットにドラッグします。各コードセグメントは、1回、複数回、またはまったく使用されません。分割バーをペインの間にドラッグするか、コンテンツを表示する。）

```
str = $("form").serialize();
```

```
str = decodeURIComponent(str);
```

```
str = $("form").contents.toString();
```

```
str = $("form").toLocalestring();
```

```
str = decodeURI(str);
```

```
<!DOCTYPE html>
<html>
<head>
  <script src=
    "http://code.jquery.com/jquery-latest.js">
</script>
</head>
<body>
  <form>
    <input type="text" name="email" />
    <input type="submit" value="Submit" />
  </form>
  <script>
    $("form").submit(function () {
      var str;
      
      
      return true;
    });
  </script>
</body>
</html>
```

Answer:

```
str = $("form").contents.toString();
```

```
str = $("form").toLocalestring();
```

```
str = decodeURI(str);
```

```
<!DOCTYPE html>
<html>
<head>
  <script src=
  "http://code.jquery.com/jquery-latest.js">
</script>
</head>
<body>
  <form>
    <input type="text" name="email" />
    <input type="submit" value="Submit" />
  </form>
  <script>
    $("form").submit(function () {
      var str;
      str = $("form").serialize();
      str = decodeURIComponent(str);
      return true;
    });
  </script>
</body>
</html>
```

5. Windows Communication Foundation(WCF)サービスを使用するアプリケーションを開発しています。アプリケーションは、次のコードを使用してサービスと対話します。（行番号は参照用にのみ記載されています）。

```
01 <script>
02   function getCountry(country) {
03     $.ajax({
04       type: "POST",
05       contentType: "application/json; charset=utf-8",
06       url: "http://contoso.com/Service.svc/GetCountry",
07       data: '{"Country":"' + country + "'",
08       datatype: "json"
09       success: OnSuccess,
10       error: OnError
11     });
12   }
13 }
14 function OnSuccess(data, status) {
15   ...
16 }
17 function OnError(request, status, error) {
18   ...
19 }
20 </script>
```

WCF サービスを認証する必要があります。

あなたは何をすべきか？

A. 11 行目に次のコード行を追加します。

, ユーザー名: あなたのユーザー名

, パスワード: あなたのパスワード

B. 11 行目に次のコード行を追加します。

, 資格情報: プロンプト

C. 06 行目でコードを次のコード行に置き換えます。

url: "http://contoso.com/Service.svc/GetCountry?"

ユーザー名=ユーザー名とパスワード=パスワード "

D. 11 行目に次のコード行を追加します。 ユーザー名とパスワードは XML ファイルに保存されます。

,資格情報: credentials.xml

Answer: C