

IBM Worklight V6.1.0 **入門**

マルチページ・アプリケーションの作成



商標

- IBM、IBM ロゴおよび [ibm.com](http://www.ibm.com) は、世界の多くの国で登録された International Business Machines Corporation の商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、<http://www.ibm.com/legal/copytrade.shtml> をご覧ください。
- Java およびすべての Java 関連の商標およびロゴは Oracle やその関連会社の米国およびその他の国における商標または登録商標です。
- この資料は、事前に IBM の書面による許可を得ずにその一部または全部を複製することは禁じられています。

IBM® について

- <http://www.ibm.com/ibm/us/en/> を参照してください。

アジェンダ

- マルチページ・アプリケーションの基礎
- 外部 HTML ファイルをロードする
- ヒストリーを持つページ・ナビゲーションを実装する
- 演習

マルチページ・アプリケーションの基礎

- IBM Worklight® ハイブリッド・アプリケーションは、シングル DOM モデルを使用しています。
- シングル DOM モデルとは、ハイパーリンクを使用したり、`window.location` プロパティを変更したりすることによって、さまざまな HTML ファイル間でナビゲートしてはならないことを意味します。
- その代わりに、外部の HTML ファイル・コンテンツをロードし、Ajax 要求を使用して既存の DOM に挿入することによって、マルチページ・インターフェースを実装する必要があります。
- これは、メインのアプリケーション HTML ファイルによって IBM Worklight クライアント・サイド JavaScript™ フレームワーク・ファイルがロードされるためであり、ブラウザで HTML ファイル間をナビゲートすると、JavaScript のコンテキストやロードされたスクリプトが失われるためです。
- 今日使用可能なほとんどの JavaScript UI フレームワーク (jQuery Mobile、Sencha Touch、Dojo Mobile など) では、必要なマルチページ・ナビゲーションを行うことができる幅広い API が提供されています。
- このモジュールでは、組み込み機能のみを使用してマルチページ IBM Worklight アプリケーションをビルドする方法について学習します。
- **重要:** JavaScript UI フレームワークを使用している場合は、このモジュールで説明されている組み込み機能を使用しないでください。代わりに、フレームワーク API を使用する必要があります。

マルチページ・アプリケーションの基礎

- 複数のページを含むアプリケーションをビルドするには、以下の2通りの方法があります。

すべてのアプリケーション
・ ページを含む単一の
HTML ファイル



アプリケーション・
ページごとに個別の
HTML ファイル



マルチページ・アプリケーションの基礎

- 単純なアプリケーションに推奨されるモデルは、単一の HTML ファイルです。
- 開発者は「現在の」ページ `<div>` を表示し、残りを非表示にする必要があります。
- ただし、大規模なアプリケーションの場合、以下の課題があります。
 - 開発者は、どの DIV を表示し、どの DIV を非表示にするかについて、どのような場合もすべての責任を負う必要があります。
 - 何らかの新しいコンテンツ (表など) をページに追加する場合、事前に準備されたテンプレートをロードすることはできず、テンプレートを手動で生成する必要があります。
 - 多数のページを含む大容量の単一 HTML ファイルの場合、ロードに時間がかかります。
 - 複数のページを含む単一の HTML ファイルの場合、簡単に「迷子」になります。個別のファイルの方が簡単に管理できます。

マルチページ・アプリケーションの基礎

- 動的なページのロードによって、複数のページを含む動的なリッチ・アプリケーションをより簡単にビルドすることができます。
- 組み込み jQuery API を使用して、アプリケーションで DOM エLEMENTの動的なロード、更新、および挿入を行うことができます。
- 必要に応じて、CSS や JavaScript を使用した HTML ページを挿入できます。
- ナビゲーション・ヒストリーを実装できます。
- ページをロードまたはアンロードする際に、JavaScript コードを実行できます。
- これ以降のスライドでは、単純なマルチページ・ナビゲーション・インターフェースを実装する方法について学習します。

マルチページ・アプリケーションの基礎

- **注: Windows Phone 8 環境でマルチページ・ナビゲーションを実装する場合、jQuery load() 関数を使用するたびに URL を変更する必要があります。**
- URL パス・ストリングの先頭に、「www/default/」を追加します。
- 例:

```
$("#pagePort").load("pages/MainPage.html", function() {  
    currentPage.init();  
});
```

- URL パス・ストリングを次に変更する必要があります。

```
$("#pagePort").load("  
www/default/pages/MainPage.html", function() {  
    currentPage.init();  
});  
}
```

- このトレーニング・モジュールの例で、このタスクを実行する方法について詳しく説明しています。

マルチページ・アプリケーションの基礎

- **注: Windows Phone 8 環境でマルチページ・ナビゲーションを実装し、jQuery Mobile を `changePage()` 関数で使用すると、jQuery Mobile の問題が原因で失敗します。**
- この問題を解決するには、以下の Stackoverflow トピックで説明されている、jQuery Mobile の .js ファイルで行う必要がある変更について調べてください。
- <http://stackoverflow.com/questions/17965560/ibm-worklight-v-5-0-6-cant-navigate-multipages-on-windows-phone-7-5-environme>

アジェンダ

- マルチページ・アプリケーションの基礎
- 外部 HTML ファイルをロードする
- ヒストリーを持つページ・ナビゲーションを実装する
- 演習

外部 HTML ファイルをロードする

- 外部 HTML ファイルは、既存の DOM の任意の場所に挿入できる HTML コードのセグメントです。
- 単一の HTML ファイルに、複数の HTML エlement階層を含めることができます。
- `<script>` タグを使用して、JavaScript を含めることもできます。
- `<link>` タグを使用して、CSS ファイルを含めることもできます。
- 親Element (通常は `<div>`) に挿入することもできますが、必須ではありません。
- jQuery の `$.load()` API を使用して実装します。

外部 HTML ファイルをロードする

- HTML ファイルをロードするには、次の構文を使用します。


```
$(containerSelector).load(filePath, callbackFunction);
```

- `containerSelector` – ロードするコンテンツをホストする、エレメントの jQuery CSS セレクター。
- `filePath` – HTML ファイルへの相対パス。常に、メイン HTML ファイルを基準としたパスになります。
- `callbackFunction` – ロードが完了したときに実行される JavaScript 関数

外部 HTML ファイルをロードする

- 例

```
$("#pagePort").load("pages/MainPage.html", function(){  
    alert("loaded!");  
});
```



```
<div id="AppBody">  
    <!-- This is static header, it will be shown always -->  
    <div id="header">  
        <h1>Pages and Fragments</h1>  
    </div>  
  
    <!-- This is a placeholder for dynamic page content -->  
    <div id="pagePort"></div>  
</div>
```

- このコードは **MainPage.html** ファイルをロードし、そのコンテンツを pagePort <div> エレメントに挿入します。
- **MainPage.html** からの JavaScript 参照と CSS 参照が DOM にロードされます。
- `alert("loaded!")` は、**MainPage.html** のロードが完了すると実行されます。

アジェンダ

- マルチページ・アプリケーションの基礎
- 外部 HTML ファイルをロードする
- **ヒストリーを持つページ・ナビゲーションを実装する**
- 演習

履歴を持つページ・ナビゲーションを実装する

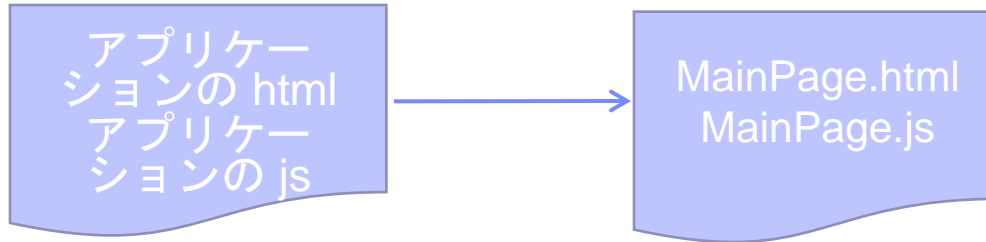
- 前に説明した手法を使用して外部 HTML ファイルをロードすることにより、履歴を持つナビゲーション・インターフェースを実装できます。



- ナビゲーション・履歴を配列オブジェクトのスタックとして保持する必要があります。
- また、JavaScript オブジェクト変数を使用して、現在ロードされているページ機能への参照を保持する必要があります。

```
var pagesHistory = [];  
var currentPage = {};
```

履歴を持つページ・ナビゲーションを実装する - ステップ1



```

$("#pagePort").load("pages/MainPage.html", function(){
  currentPage.init();
});
  
```

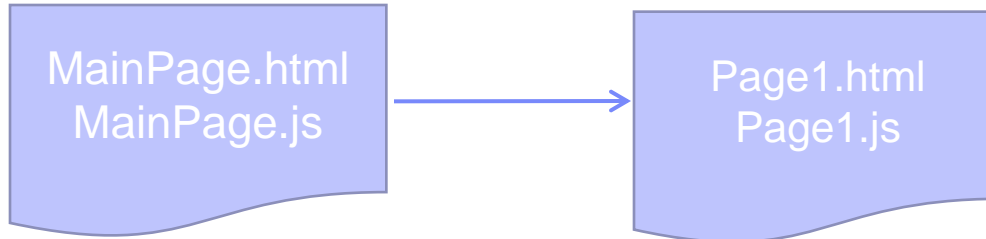
```

currentPage = {};
currentPage.init = function(){
  WL.Logger.debug("MainPage :: init");
};
  
```

1. アプリケーションの開始時に、**MainPage.html** がアプリケーション・コードからロードされて #pagePort div に挿入されます。
5. **MainPage.html** のロードが完了すると、`currentPage.init` メソッドが呼び出されます。

2. **MainPage.js** は、**MainPage.html** の一部としてロードされます。
3. `currentPage` オブジェクトが宣言されます。
4. `currentPage.init` 関数が宣言されます。

ヒストリーを持つページ・ナビゲーションを実装する – ステップ2



```

currentPage.loadPage = function(pageIndex){
  WL.Logger.debug("MainPage :: loadPage :: pageIndex: " + pageIndex);
  pagesHistory.push("pages/MainPage.html");
  $("#pagePort").load("pages/Page" + pageIndex + ".html", function(){
    currentPage.init();
  });
};
  
```

```

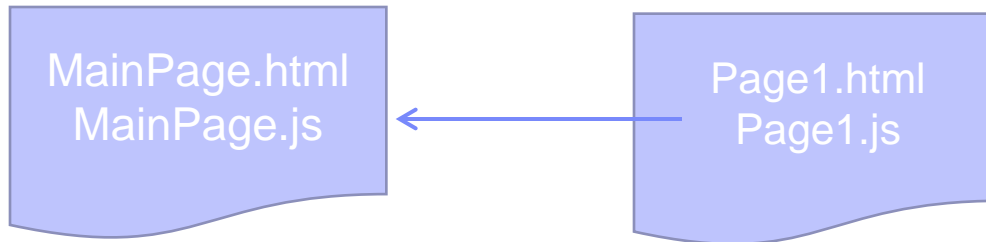
currentPage={};

currentPage.init = function() {
  WL.Logger.debug("Page1 :: init");
};
  
```

1. **MainPage.html** が **pagesHistory** スタックにプッシュされます。
2. **Page1.html** がロードされ、**#pagePort** div に挿入されます。
5. **Page1.html** のロードが完了すると、新しい **currentPage.init** メソッドが呼び出されます。

2. **Page1.js** は、**Page1.html** の一部としてロードされます。
3. **currentPage** オブジェクトが宣言され、古いオブジェクトがオーバーライドされます。
4. **currentPage.init** 関数が宣言されます。

ヒストリーを持つページ・ナビゲーションを実装する – ステップ3



```

currentPage = {};
currentPage.init = function(){
    WL.Logger.debug("MainPage :: init");
};
  
```

```

currentPage.back = function(){
    WL.Logger.debug("Page1 :: back");
    $("#pagePort").load(pagesHistory.pop(), function(){
        currentPage.init();
    });
};
  
```

3. **MainPage.js** は、**MainPage.html** の一部としてロードされます。
4. `currentPage` オブジェクトが宣言され、古いオブジェクトがオーバーライドされます。
5. `currentPage.init` 関数が宣言されます。

1. 以前の HTML ファイル名が、**PagesHistory** スタックからポップされます (**MainPage.html**)。
2. このファイルがロードされ、`#pagePort` div に挿入されます。
6. **MainPage.html** のロードが完了すると、`currentPage.init` メソッドが呼び出されます。

アジェンダ

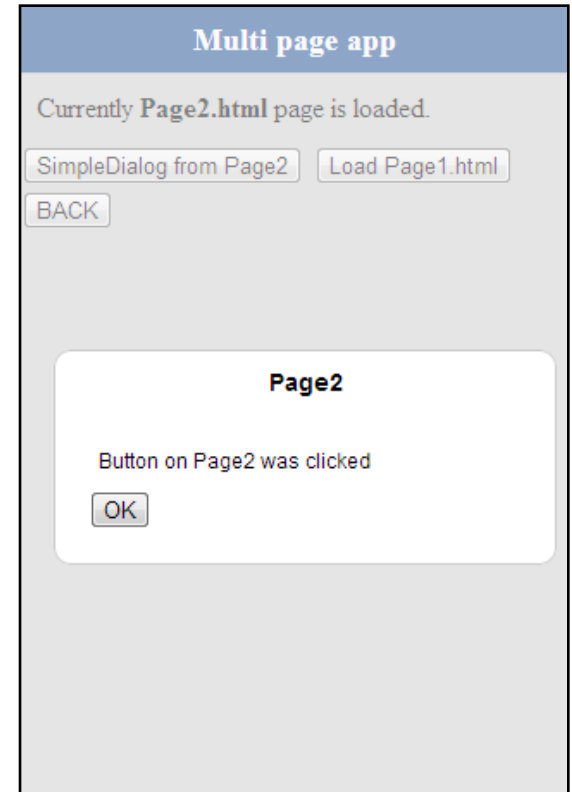
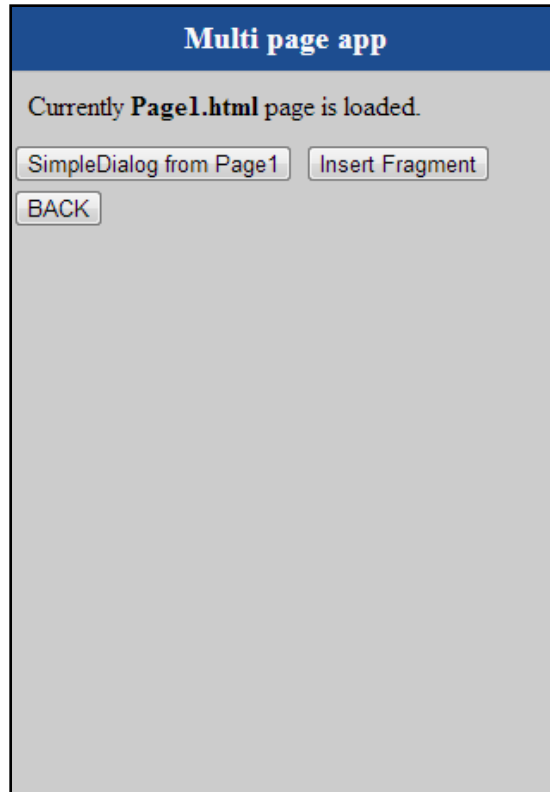
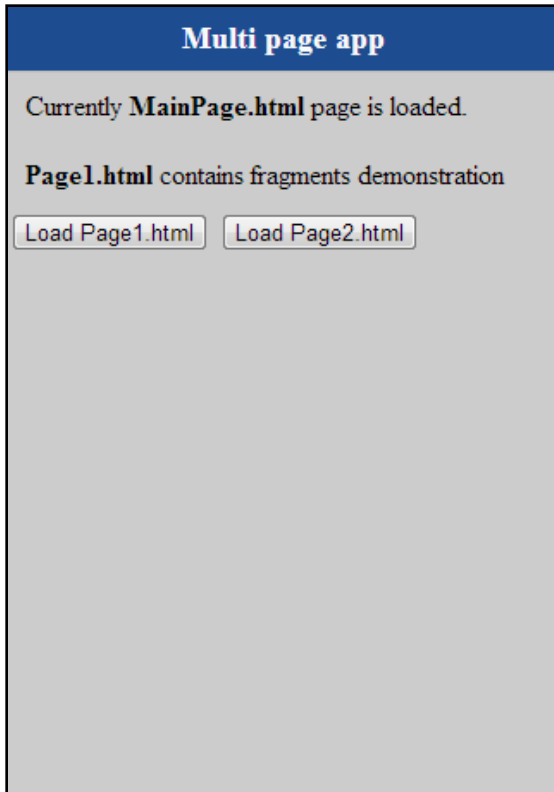
- マルチページ・アプリケーションの基礎
- 外部 HTML ファイルをロードする
- ヒストリーを持つページ・ナビゲーションを実装する
- 演習

演習

- 3つのページを含む単純なアプリケーションを作成します。
 - ロードしたページのコンテナーとしてアプリケーションHTMLファイルに含める `<div>` エレメントは、1つのみである必要があります。
 - **MainPage.html** は、アプリケーションの開始時に自動的にロードされて表示されるファイルです。
 - **MainPage.html** には、「ページ1のロード (load page1)」と「ページ2のロード (load page2)」という2つのボタンが必要です。
 - 各ボタンをクリックすると、対応するページがロードされます。
 - **Page1.html** と **Page2.html** の両方に、`WL.SimpleDialog` を表示するボタンが含まれている必要があります。
 - **Page1.html** と **Page2.html** の両方に、「戻る」ボタンが必要です。
 - **Page1.html** には、動的にHTMLフラグメントをロードし、事前定義プレースホルダー (`<div>`) に追加するボタンが必要です。
 - **Page2.html** には、**Page1.html** をロードするボタンが必要です。

演習

このトレーニング・モジュールのサンプルは、IBM Worklight 文書 Web サイト (<http://www.ibm.com/mobile-docs>) の「入門」ページにあります。



特記事項

- これらの資料は、以下のご使用条件に同意していただける場合に限りご使用いただけます。
- 本書は米国 IBM が提供する製品およびサービスについて作成したものです。
- 本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、または サービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。
- IBM は、本書に記載されている内容に関して特許権（特許出願中のものを含む）を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。
 - 〒103-8510
東京都中央区日本橋箱崎町19番21号
日本アイ・ビー・エム株式会社
法務・知的財産
知的財産権ライセンス渉外
- 以下の保証は、国または地域の法律に沿わない場合は、適用されません。IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。
- この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。
- 本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。
- IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。
- 本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム（本プログラムを含む）との間で情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。
 - IBM Corporation
Dept F6, Bldg 1
294 Route 100
Somers NY 10589-3216
USA

- 本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。
- 本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。
- IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者にお問い合わせください。
- 著作権使用許諾:**
 - 本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。
 - それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、次のように、著作権表示を入れていただく必要があります。
 - © (お客様の会社名) (西暦年) このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。 © Copyright IBM Corp. _年を入れる_ All rights reserved.

プライバシー・ポリシーの考慮事項

- サービス・ソリューションとしてのソフトウェアも含めた IBM ソフトウェア製品（「ソフトウェア・オファリング」）では、製品の使用に関する情報の収集、エンド・ユーザーの使用感の向上、エンド・ユーザーとの対話またはその他の目的のために、Cookie ははじめさまざまなテクノロジーを使用することがあります。多くの場合、ソフトウェア・オファリングにより個人情報が収集されることはありません。IBM の「ソフトウェア・オファリング」の一部には、個人情報を収集できる機能を持つものがあります。ご使用の「ソフトウェア・オファリング」が、これらの Cookie およびそれに類するテクノロジーを通じてお客様による個人情報の収集を可能にする場合、以下の具体的事項を確認ください。
- このソフトウェア・オファリングは、展開される構成に応じて、(アプリケーション・サーバーが生成する) セッション情報を収集するセッションごとの Cookie を使用する場合があります。これらの Cookie は個人情報を含まず、セッション管理のために要求されるものです。加えて、匿名ユーザーの認識および管理のために持続的な Cookie が無作為に生成される場合があります。これらの Cookie も個人情報を含まず、要求されるものです。
- この「ソフトウェア・オファリング」が Cookie およびさまざまなテクノロジーを使用してエンド・ユーザーから個人を特定できる情報を収集する機能を提供する場合、お客様は、このような情報を収集するにあたって適用される法律、ガイドライン等を遵守する必要があります。これには、エンドユーザーへの通知や同意の要求も含まれますがそれには限られません。このような目的での Cookie を含む様々なテクノロジーの使用の詳細については、IBM の『IBM オンラインでのプライバシー・ステートメント』(<http://www.ibm.com/privacy/details/jp/ja/>) の「クッキー、ウェブ・ビーコン、その他のテクノロジー」および『IBM Software Products and Software-as-a-Service Privacy Statement』(<http://www.ibm.com/software/info/product-privacy/>) を参照してください。

サポートおよびコメント

- IBM Worklight の一連の文書、トレーニング資料、および質問をポストできるオンライン・フォーラムはすべて、次の IBM Web サイトからご覧になれます。
 - <http://www.ibm.com/mobile-docs>
- サポート
 - ソフトウェア・サブスクリプション & サポート (ソフトウェア・メンテナンスと呼ばれる場合もあります) は、パスポート・アドバンテージおよびパスポート・アドバンテージ・エクスプレスから購入されたライセンスに含まれています。International Passport Advantage Agreement および IBM International Passport Advantage Express Agreement の追加情報については、次のパスポート・アドバンテージ Web サイトを参照してください。
 - <http://www.ibm.com/software/passportadvantage>
 - ソフトウェア・サブスクリプション & サポートが有効になっている場合、IBM は、インストールおよび使用法 (ハウツー) に関する短期間の FAQ に対するサポートや、コード関連の質問に対するサポートを提供します。詳しくは、次の IBM ソフトウェア・サポート・ハンドブックを参照してください。
 - <http://www.ibm.com/support/handbook>
- ご意見
 - 本資料に関するご意見をお寄せください。本資料の具体的な誤りや欠落、正確性、編成、題材、または完成度に関するご意見をお寄せください。お寄せいただくご意見は、本マニュアルまたは製品の情報、およびその情報の提示方法に関するもののみとしてください。
 - 製品の技術的な質問および情報、および価格については、担当の IBM 営業所、IBM ビジネス・パートナー、または認定リマーカーターにお問い合わせください。
 - IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。IBM またはいかなる組織も、お客様から提示された問題についてご連絡を差し上げる場合にのみ、お客様が提供する個人情報を使用するものとします。
 - どうぞよろしくお願いいたします。
 - 次の IBM Worklight Developer Edition サポート・コミュニティにご意見をお寄せください。
 - <https://www.ibm.com/developerworks/mobile/worklight/connect.html>
 - IBM からの回答を希望される場合は、以下の情報をご連絡ください。
 - 氏名
 - 住所
 - 企業または組織
 - 電話番号
 - Eメール・アドレス

ありがとうございました

