



One Identity Manager 8.1.2

HTML5-Entwicklungshandbuch

Copyright 2020 One Identity LLC.

ALLE RECHTE VORBEHALTEN.

Diese Anleitung enthält urheberrechtlich geschützte Informationen. Die in dieser Anleitung beschriebene Software wird unter einer Softwarelizenz oder einer Geheimhaltungsvereinbarung bereitgestellt. Diese Software darf nur in Übereinstimmung mit den Bestimmungen der geltenden Vereinbarung verwendet oder kopiert werden. Kein Teil dieser Anleitung darf ohne die schriftliche Erlaubnis von One Identity LLC in irgendeiner Form oder mit irgendwelchen Mitteln, elektronisch oder mechanisch reproduziert oder übertragen werden, einschließlich Fotokopien und Aufzeichnungen für irgendeinen anderen Zweck als den persönlichen Gebrauch des Erwerbers.

Die Informationen in diesem Dokument werden in Verbindung mit One Identity Produkten bereitgestellt. Durch dieses Dokument oder im Zusammenhang mit dem Verkauf von One Identity LLC Produkten wird keine Lizenz, weder ausdrücklich oder stillschweigend, noch durch Duldung oder anderweitig, an jeglichem geistigen Eigentumsrecht eingeräumt. MIT AUSNAHME DER IN DER LIZENZVEREINBARUNG FÜR DIESES PRODUKT GENANNTEN BEDINGUNGEN ÜBERNIMMT ONE IDENTITY KEINERLEI HAFTUNG UND SCHLIESST JEGLICHE AUSDRÜCKLICHE, IMPLIZIERTE ODER GESETZLICHE GEWÄHRLEISTUNG ODER GARANTIE IN BEZUG AUF IHRE PRODUKTE AUS, EINSCHLIESSLICH, ABER NICHT BESCHRÄNKT AUF DIE IMPLIZITE GEWÄHRLEISTUNG DER ALLGEMEINEN GEBRAUCHSTAUGLICHKEIT, EIGNUNG FÜR EINEN BESTIMMTEN ZWECK ODER NICHTVERLETZUNG VON RECHTEN. IN KEINEM FALL HAFTET ONE IDENTITY FÜR JEGLICHE DIREKTE, INDIREKTE, FOLGE-, STÖRUNGS-, SPEZIELLE ODER ZUFÄLLIGE SCHÄDEN (EINSCHLIESSLICH, OHNE EINSCHRÄNKUNG, SCHÄDEN FÜR VERLUST VON GEWINNEN, GESCHÄFTSUNTERBRECHUNGEN ODER VERLUST VON INFORMATIONEN), DIE AUS DER NUTZUNG ODER UNMÖGLICHKEIT DER NUTZUNG DIESES DOKUMENTS RESULTIEREN, SELBST WENN ONE IDENTITY AUF DIE MÖGLICHKEIT SOLCHER SCHÄDEN HINGEWIESEN HAT. One Identity übernimmt keinerlei Zusicherungen oder Garantien hinsichtlich der Richtigkeit und Vollständigkeit des Inhalts dieses Dokuments und behält sich das Recht vor, Änderungen an Spezifikationen und Produktbeschreibungen jederzeit ohne vorherige Ankündigung vorzunehmen. One Identity verpflichtet sich nicht, die in diesem Dokument enthaltenen Informationen zu aktualisieren.

Wenn Sie Fragen zu Ihrer potenziellen Nutzung dieses Materials haben, wenden Sie sich bitte an:

One Identity LLC.
Attn: LEGAL Dept
4 Polaris Way
Aliso Viejo, CA 92656

Besuchen Sie unsere Website (<http://www.OneIdentity.com>) für regionale und internationale Büro-Adressen.


Patente


One Identity ist stolz auf seine fortschrittliche Technologie. Für dieses Produkt können Patente und anhängige Patente gelten. Für die aktuellsten Informationen über die geltenden Patente für dieses Produkt besuchen Sie bitte unsere Website unter <http://www.OneIdentity.com/legal/patents.aspx>.

Marken

One Identity und das One Identity Logo sind Marken und eingetragene Marken von One Identity LLC. in den USA und anderen Ländern. Für eine vollständige Liste der One Identity Marken besuchen Sie bitte unsere Website unter www.OneIdentity.com/legal. Alle anderen Marken sind Eigentum der jeweiligen Besitzer.

Legende

-  **WARNUNG:** Das Symbol **WARNUNG** weist auf ein potenzielles Risiko von Körperverletzungen oder Sachschäden hin, für das Sicherheitsvorkehrungen nach Industriestandard empfohlen werden. Dieses Symbol ist oft verbunden mit elektrischen Gefahren bezüglich Hardware.

-  **VORSICHT:** Das Symbol **VORSICHT** weist auf eine mögliche Beschädigung von Hardware oder den möglichen Verlust von Daten hin, wenn die Anweisungen nicht befolgt werden.

Inhalt

Über dieses Handbuch	4
Architektur der One Identity Manager-HTML-Anwendungen	5
HTML-Anwendungsentwicklung	6
One Identity Visual Studio Code-Erweiterung installieren	7
Arbeitsumgebung einrichten	7
Verbindung zur One Identity Manager-Datenbank herstellen	7
Entwicklungsordner einrichten	8
One Identity Visual Studio Code-Erweiterung verwenden	9
Kompilierungsbranche	9
Kompilierungsbranche auswählen	9
Kompilierungsbranche erstellen	10
Kompilierungsbranche löschen	10
API Client aktualisieren	10
HTML-Anwendungen automatisch einrichten	11
HTML-Anwendungen kompilieren und in Datenbank speichern	11
Übersetzungen prüfen	12
Anwendung erstellen	12
Über uns	16
Kontaktieren Sie uns	16
Technische Supportressourcen	16

Über dieses Handbuch

Dieses Handbuch zeigt Webentwicklern, wie sie die HTML-Anwendungen des One Identity Manager im Quelltext ansehen und deren interne Funktionsweise nachvollziehen können. Dazu stellt Ihnen One Identity eine Erweiterung für Visual Studio Code zur Verfügung.

Verfügbare Dokumentation

Die Online Version der One Identity Manager Dokumentation finden Sie im Support-Portal unter [Online-Dokumentation](#). Videos mit zusätzlichen Informationen finden Sie unter www.YouTube.com/OneIdentity.

Architektur der One Identity Manager-HTML-Anwendungen

Der One Identity Manager verwaltet eine Ordnerstruktur, in der die Quelldateien für alle HTML-Anwendungen liegen. Diese Ordnerstruktur ist im Datenmodell als Teil der automatischen Software-Aktualisierung abgelegt und dort der Maschinenrolle **HTML Development** zugewiesen.

Sie können diese Ordnerstruktur lokal in einem Entwicklungsordner ablegen. Dort liegen auf oberster Ebene die Ordner für Datenbankmodule. Darunter liegt für jede HTML-Anwendung ein eigener Ordner. Wie Sie einen solchen Entwicklungsordner einrichten, erfahren Sie im Kapitel [Entwicklungsordner einrichten](#) auf Seite 8.

Die HTML-Anwendungen sind als nodeJS-Anwendungen aufgebaut, die das Framework **Angular** einsetzen. Grundsätzlich werden alle HTML-Anwendungen unterstützt, die sich als nodeJS-Anwendungen kompilieren lassen.

Die HTML-Anwendungen nutzen für die Kommunikation mit der One Identity Manager-API den API Client. Der API Client ist eine npm-Bibliothek, die automatisch bei der Kompilierung der API erzeugt und in der Datenbank abgelegt wird. Der API Client regelt den kompletten Netzwerkzugriff auf den API Server.

Ein Großteil der Logik der HTML-Anwendungen ist in Form von Plugins realisiert, die unabhängig von einer konkreten HTML-Anwendung genutzt werden können. Damit können Sie diese Logik auch für Ihre eigenen Anwendungen nutzen.

HTML-Anwendungsentwicklung

In diesem Kapitel finden Sie Informationen über den Umgang mit der One Identity Visual Studio Code-Erweiterung.

Mit der Visual Studio Code-Erweiterung können Sie die Kommandos des ImxClient-Kommandozeilenprogramms ohne tiefere Kenntnis der Syntax direkt in Visual Studio Code aufrufen:

- [Verbinden](#) zur Datenbank
- [Einrichten](#) des Entwicklungsordners
- [Verwalten und verwenden](#) von Kompilierungszweigen
- [Aktualisieren](#) der HTML-Anwendungen mit gleichzeitiger Einrichtung
- [Kompilieren](#) einer HTML-Anwendung
- [Aktualisieren](#) der Basisbibliotheken (API-Client)
- [Prüfen](#) auf fehlende Übersetzungen

Die Visual Studio Code-Erweiterung liefert Ihnen zudem folgende hilfreiche Informationen:

- Status der aktuellen Datenbankverbindung
- Informationen über den verwendeten ImxClient
- Informationen über den verwendeten API-Client
- Zustand des Entwicklungsordners
- Anzeige des Kompilierungszweiges
- Informationen zur letzten Übersetzungsprüfung

Verwandte Themen

- [One Identity Visual Studio Code-Erweiterung installieren](#) auf Seite 7
- [Arbeitsumgebung einrichten](#) auf Seite 7
- [One Identity Visual Studio Code-Erweiterung verwenden](#) auf Seite 9
- [Anwendung erstellen](#) auf Seite 12

One Identity Visual Studio Code-Erweiterung installieren

Um die One Identity Visual Studio Code-Erweiterung zu installieren

1. Starten Sie Visual Studio Code.
2. In Visual Studio Code klicken Sie in der linken Navigation auf **Erweiterungen**.
3. Im Bereich **Erweiterungen** klicken Sie auf **... | Aus VSIX installieren**.
4. Im Datei-Browser wählen Sie im One Identity Manager-Installationsverzeichnis die Datei `vcode-extension.vsix` und klicken Sie **Installieren**.
5. Starten Sie Visual Studio Code neu.
6. In Visual Studio Code klicken Sie in der linken Navigation auf **Explorer**.
7. Im Bereich **Explorer** navigieren Sie zu **One Identity | Configuration**.
8. Neben **Imx client** klicken Sie **One Identity Manager: edit imx client path**.
9. Im Datei-Browser wählen Sie im One Identity Manager-Installationsverzeichnis die Datei `ImxClient.exe` und klicken Sie **Öffnen**.

Arbeitsumgebung einrichten

In diesem Kapitel erfahren Sie, wie Sie mithilfe der One Identity Visual Studio Code-Erweiterung Ihre Arbeitsumgebung einrichten, sodass Sie anschließend problemlos die Visual Studio Code-Erweiterung und ihre Funktionen verwenden können.

Verwandte Themen

- [Verbindung zur One Identity Manager-Datenbank herstellen](#) auf Seite 7
- [Entwicklungsordner einrichten](#) auf Seite 8

Verbindung zur One Identity Manager-Datenbank herstellen

Viele der Funktionen, die Ihnen die Visual Studio Code-Erweiterung bietet, benötigen eine Verbindung zur One Identity Manager-Datenbank.

Um eine Verbindung zur One Identity Manager-Datenbank herzustellen

1. In Visual Studio Code klicken Sie in der linken Navigation auf **Explorer**.
2. Im Bereich **Explorer** navigieren Sie zu **One Identity | Workspace**.
3. Neben **Database** klicken Sie **One Identity Manager: connect to database**.
Das Dialogfenster zum Auswählen der Datenbankverbindung öffnet sich.
4. Im Dialogfenster nehmen Sie eine der folgenden Aktionen vor:
 - Um eine bestehende Verbindung zur One Identity Manager-Datenbank zu verwenden, wählen Sie aus der Auswahlliste **Datenbankverbindung auswählen** die entsprechende Verbindung aus.
- ODER -
 - Um eine neue Verbindung zur One Identity Manager-Datenbank zu verwenden, klicken Sie **Neue Verbindung erstellen** und geben Sie eine neue Verbindung an.
5. Unter **Authentifizierungsverfahren** geben Sie das Verfahren und die Anmeldedaten an, mit denen Sie sich an der Datenbank anmelden möchten.
6. Klicken Sie **Anmelden**.

Entwicklungsordner einrichten

Wenn Sie einen Entwicklungsordner einrichten werden die folgenden Aktionen durchgeführt:

- Die Ordner für HTML-Anwendungen werden aus der Datenbank heruntergeladen und als Unterordner des Entwicklungsordners gespeichert.
- Die Bibliotheken werden aus der Datenbank heruntergeladen und im Unterordner `imx-modules` des Entwicklungsordners gespeichert.
- Die neuesten Bibliotheken werden in den Unterordner `imx-modules` des Entwicklungsordners heruntergeladen. Wenn keine Datenbankverbindung besteht, wird dieser Schritt übersprungen (siehe auch [Verbindung zur One Identity Manager-Datenbank herstellen](#) auf Seite 7).

Um einen One Identity Manager-Entwicklungsordner einzurichten

1. In Visual Studio Code klicken Sie in der Menüleiste **Datei | Ordner öffnen**.
2. Im Datei-Browser wählen Sie den Ordner, den Sie als Entwicklungsordner verwenden möchten.
3. In Visual Studio Code klicken Sie in der linken Navigation auf **Explorer**.
4. Im Bereich **Explorer** navigieren Sie zu **One Identity | Workspace**.
5. Neben **Development folder** klicken Sie **One Identity Manager: set up development folder**.
6. Bestätigen Sie die Abfrage mit **Continue setup**.

One Identity Visual Studio Code-Erweiterung verwenden

In diesem Kapitel erfahren Sie, wie Sie die One Identity Visual Studio Code-Erweiterung verwenden.

Verwandte Themen

- [Kompilierungsbranche](#) auf Seite 9
- [API Client aktualisieren](#) auf Seite 10
- [HTML-Anwendungen automatisch einrichten](#) auf Seite 11
- [HTML-Anwendungen kompilieren und in Datenbank speichern](#) auf Seite 11
- [Übersetzungen prüfen](#) auf Seite 12

Kompilierungsbranche

Um verschiedene Versionen Ihres kompilierten Projektes zu verwalten und in der Datenbank zu hinterlegen, verwenden Sie Kompilierungsbranche. Sie können den zu verwendenden Kompilierungsbranch [auswählen](#), die vorhandenen Kompilierungsbranch anzeigen, Kompilierungsbranch [erstellen](#) und Kompilierungsbranch [löschen](#).

Verwandte Themen

- [Kompilierungsbranch auswählen](#) auf Seite 9
- [Kompilierungsbranch erstellen](#) auf Seite 10
- [Kompilierungsbranch löschen](#) auf Seite 10

Kompilierungsbranch auswählen

Um einen Kompilierungsbranch auszuwählen und zu verwenden

1. In Visual Studio Code klicken Sie in der linken Navigation auf **Explorer**.
2. Im Bereich **Explorer** navigieren Sie zu **One Identity | Configuration**.
3. Neben **Branch ID**, klicken Sie **One Identity Manager: edit branch id**.

TIPP: Falls noch keine Verbindung zur One Identity Manager-Datenbank besteht, müssen Sie diese jetzt herstellen (siehe [Verbindung zur One Identity Manager-Datenbank herstellen](#) auf Seite 7).

Eine Auswahlliste mit den verfügbaren Kompilierungsbranchen öffnet sich.

4. In der Auswahlliste klicken Sie den Kompilierungsweig, den Sie verwenden möchten.

Kompilierungsweige erstellen

Um einen Kompilierungsweig zu erstellen

1. In Visual Studio Code klicken Sie in der linken Navigation auf **Explorer**.
2. Im Bereich **Explorer** navigieren Sie zu **One Identity | Configuration**.
3. Neben **Branch ID**, klicken Sie **Add branch**.
Ein Eingabefeld öffnet sich.
4. Im Eingabefeld geben Sie einen Namen für den Kompilierungsweig ein und drücken Sie **Enter**.

TIPP: Falls noch keine Verbindung zur One Identity Manager-Datenbank besteht, müssen Sie diese jetzt herstellen (siehe [Verbindung zur One Identity Manager-Datenbank herstellen](#) auf Seite 7).

Kompilierungsweige löschen

Um einen Kompilierungsweig zu löschen

1. In Visual Studio Code klicken Sie in der linken Navigation auf **Explorer**.
2. Im Bereich **Explorer** navigieren Sie zu **One Identity | Configuration**.
3. Neben **Branch ID**, klicken Sie **One Identity Manager: edit branch id**.
TIPP: Falls noch keine Verbindung zur One Identity Manager-Datenbank besteht, müssen Sie diese jetzt herstellen (siehe [Verbindung zur One Identity Manager-Datenbank herstellen](#) auf Seite 7).
Eine Auswahlliste mit den verfügbaren Kompilierungsweigen öffnet sich.
4. In der Auswahlliste klicken Sie den Kompilierungsweig, den Sie löschen möchten.
5. Im Bereich **Explorer** neben **Branch ID**, klicken Sie **Delete branch**.
6. Bestätigen Sie die Abfrage mit **OK**.

API Client aktualisieren

Sie können die Basisbibliotheken und damit den API Client jederzeit aktualisieren.

Um den API-Client zu aktualisieren

1. In Visual Studio Code klicken Sie in der linken Navigation auf **Explorer**.
2. Im Bereich **Explorer** navigieren Sie zu **One Identity | Workspace**.
3. Neben **Api client**, klicken Sie **One Identity Manager: update api client**.

TIPP: Falls noch keine Verbindung zur One Identity Manager-Datenbank besteht, müssen Sie diese jetzt herstellen (siehe [Verbindung zur One Identity Manager-Datenbank herstellen](#) auf Seite 7).

HTML-Anwendungen automatisch einrichten

Um HTML-Anwendungen automatisch zu erkennen und einzurichten, führen Sie eine Überprüfung der vorhandenen Ordner in Ihrem Entwicklungsordner durch. Dabei werden folgende Schritte durchgeführt:

- Der Unterordner `assets` des Arbeitsverzeichnisses wird mit dem Unterordner `src/assets` der Anwendung verknüpft.
- Die entsprechenden Plugins werden verknüpft und integriert.

HINWEIS: HTML-Anwendungen werden ebenso vor jeder Kompilierung automatisch eingerichtet.

Um HTML-Anwendungen automatisch einzurichten

1. In Visual Studio Code klicken Sie in der linken Navigation auf **Explorer**.
2. Im Bereich **Explorer** navigieren Sie zu **One Identity | Workspace | Development folder**.
3. Neben **Apps** klicken Sie **reload existing apps**.

TIPP: Falls noch keine Verbindung zur One Identity Manager-Datenbank besteht, müssen Sie diese jetzt herstellen (siehe [Verbindung zur One Identity Manager-Datenbank herstellen](#) auf Seite 7).

HTML-Anwendungen kompilieren und in Datenbank speichern

Sie können Ihre HTML-Anwendungen kompilieren und das Ergebnis automatisch in der One Identity Manager-Datenbank speichern.

Um die HTML-Anwendungen zu kompilieren und in der Datenbank zu speichern

1. In Visual Studio Code klicken Sie in der linken Navigation auf **Explorer**.
2. Im Bereich **Explorer** navigieren Sie zu **One Identity | Workspace |**

Development folder | Apps.

3. Neben der HTML-Anwendung, die Sie kompilieren und in der Datenbank speichern möchten, klicken Sie **Compile app**.

TIPP: Falls noch keine Verbindung zur One Identity Manager-Datenbank besteht, müssen Sie diese jetzt herstellen (siehe [Verbindung zur One Identity Manager-Datenbank herstellen](#) auf Seite 7).

Die Debug-Kompilierung wird durchgeführt und die Änderungen werden im aktiven **Kompilierungszeitpunkt** gespeichert.

Übersetzungen prüfen

Sie können den Entwicklungsordner auf fehlende Übersetzungen in Ihrer HTML-Anwendung überprüfen.

Um den Entwicklungsordner auf fehlende Übersetzungen zu überprüfen

1. In Visual Studio Code klicken Sie in der linken Navigation auf **Explorer**.
2. Im Bereich **Explorer** navigieren Sie zu **One Identity | Workspace | Development folder**.
3. Neben **Translations** klicken Sie **Check translations**.

Ein Bericht öffnet sich. Dieser Bericht zeigt Ihnen alle Dateien, in denen Texte gefunden wurden, die noch nicht oder nur teilweise übersetzt wurden.

Anwendung erstellen

Sie können mithilfe der One Identity Visual Studio Code-Erweiterung Ihre eigene HTML-Anwendung erstellen und in die bestehende Umgebung einbinden.

Das Erstellen einer neuen HTML-Anwendung führen Sie in drei Schritten durch:

1. **Erstellen** Sie die neue HTML-Anwendung.
2. **Importieren** Sie die Quelldateien in die Datenbank.
3. **Legen** Sie die HTML-Anwendung in der Datenbank an.

Um eine neue HTML-Anwendung zu erstellen

1. Installieren Sie die One Identity Visual Studio Code-Erweiterung (siehe [One Identity Visual Studio Code-Erweiterung installieren](#) auf Seite 7).
2. Richten Sie mithilfe der One Identity Visual Studio Code-Erweiterung die Arbeitsumgebung ein (siehe [Arbeitsumgebung einrichten](#) auf Seite 7).
3. In Ihrer Arbeitsumgebung legen Sie einen neuen Ordner CCC an.
4. Stellen Sie sicher, dass Sie die korrekte Angular-Version verwenden (6.2.9):

- a. Öffnen Sie ein Kommandozeilenprogramm.
- b. Im Ordner CCC führen Sie den Befehl **ng version** aus.
- c. Überprüfen Sie die ausgegebene Versionsnummer. Die verwendete Version muss 6.2.9 sein.

TIPP: Wenn keine Versionsnummer ausgegeben wird, dann ist Angular weder lokal noch global installiert.

Um Angular global zu installieren, führen Sie den Befehl **npm install -g @angular/cli@6.2.9** aus.

Um Angular lokal zu installieren, führen Sie den Befehl **npm install @angular/cli@6.2.9** aus.

5. Im Ordner CCC führen Sie den Befehl **ng new <Name Ihrer HTML-Anwendung>** aus.
6. Im Ordner QBM/OpsWeb kopieren Sie die Datei `imx-plugin-config.json` und fügen Sie sie in den Ordner `CCC/<Name Ihrer HTML-Anwendung>` ein.
7. Im Ordner `CCC/<Name Ihrer HTML-Anwendung>` bearbeiten Sie die (gerade kopierte) Datei `imx-plugin-config.json`:
 - Entfernen Sie alle Kategorien außer **common** und **shared**.
8. Im Ordner `CCC/<Name Ihrer HTML-Anwendung>` bearbeiten Sie die Datei `package.json`:
 - a. Unter `scripts` fügen Sie `"build:debug"` hinzu.
 - b. Unter `dependencies` fügen Sie die Abhängigkeiten zu `imx-api`, `imx-qbm-components` und `imx-qbm-dbts` hinzu.
 - c. Unter `dependencies | rxjs` ändern Sie die Version auf `^6.3.3`.
9. Im Ordner `CCC/<Name Ihrer HTML-Anwendung>` bearbeiten Sie die Datei `angular.json`:
 - Im Bereich `projects | <Name Ihrer HTML-Anwendung> | architect | build | options` fügen Sie `"outputPath": "dist"` hinzu.
 - Im Bereich `projects | <Name Ihrer HTML-Anwendung> | architect | build | options` fügen Sie `"preserveSymlinks": true` hinzu.
10. Im Ordner `CCC/<Name Ihrer HTML-Anwendung>/src` bearbeiten Sie die Datei `index.html`:
 - Im Bereich `head` fügen Sie `<base href=".>` hinzu.
11. Im Ordner `CCC/<Name Ihrer HTML-Anwendung>/src/environments` bearbeiten Sie die Dateien `environment.prod.ts` und `environment.ts`:
 - Fügen Sie jeweils einen Eintrag für die Client-URL hinzu:
 Für `environment.<Name Ihrer Domäne>.ts`: `clientUrl: ''`
 Für `environment.ts`: `clientUrl: 'http://localhost:8182'`
12. Im Ordner `CCC/<Name Ihrer HTML-Anwendung>` bearbeiten Sie die Datei `tsconfig.json`:
 - Im Bereich `compilerOptions | paths` fügen Sie `"@shared/*": ["src/imx-plugins/QBM/shared/*"]` hinzu.
13. Stellen Sie mithilfe der One Identity Visual Studio Code-Erweiterung sicher, dass Sie

keinen Kompilierungszeitpunkt verwenden (siehe [Kompilierungszeitpunkte auswählen](#) auf Seite 9).

14. Kompilieren Sie die HTML-Anwendung mithilfe der One Identity Visual Studio Code-Erweiterung (siehe [HTML-Anwendungen kompilieren und in Datenbank speichern](#) auf Seite 11).
15. Rufen Sie die fertig kompilierte HTML-Anwendung über die URL: **\$<URL des App-Servers>/html/<Name Ihrer HTML-Anwendung>/** auf.

HINWEIS: Verwenden Sie hier den Namen der HTML-Anwendung, so wie er in der Datei `package.json` hinterlegt ist.

Um die Quelldateien in die Datenbank zu importieren

1. Wechseln Sie in Ihre Arbeitsumgebung.
2. Starten Sie das Programm Software Loader.
3. Klicken Sie auf **In Datenbank importieren**.
4. Klicken Sie **Weiter**.
5. Auf der Seite **Verbindung zur Datenbank herstellen** wählen Sie die gewünschte Datenbank aus und geben Sie die Benutzerdaten an.
6. Klicken Sie **Weiter**.
7. Wählen Sie Ihre Arbeitsumgebung aus.
8. Markieren Sie alle Dateien unterhalb des Ordners `CCC/<Name Ihrer HTML-Anwendung>`. Schließen Sie dabei die Dateien der folgenden Ordner aus:
 - `node_modules`
 - `dist`
 - `.git`
 - `src/assets`
 - `src/imx-modules`
9. Klicken Sie **Weiter**.
10. Bestätigen Sie die Abfrage mit **Ja**.
11. Auf der Seite **Maschinenrollen zuordnen** markieren Sie alle Dateien und aktivieren Sie die Maschinenrolle **HTML Development**.
12. Klicken Sie **Weiter**.
13. Auf der Seite **Änderungskennzeichen wählen** führen Sie eine der folgenden Aktionen aus:
 - Um kein Änderungskennzeichen zu verwenden, klicken Sie **Die Dateien werden keinem Änderungskennzeichen zugeordnet**.
 - Um ein Änderungskennzeichen zu verwenden, klicken Sie **Die Dateien sollen folgendem Änderungskennzeichen zugeordnet werden**. Klicken Sie anschließend **...** und wählen Sie das gewünschte Änderungskennzeichen.
14. Klicken Sie **Weiter**.

15. Nachdem die Dateien auf der Seite **Übertragen der Dateien** erfolgreich in die Datenbank übertragen wurden, klicken Sie **Weiter**.
16. Auf der Seite **Beenden des Assistenten** klicken Sie **Fertig**.

Um die HTML-Anwendung in der Datenbank anzulegen

1. Starten Sie das Programm Designer.
2. Navigieren Sie zu **Basisdaten | Sicherheitseinstellungen | HTML Anwendungen**.
3. Legen Sie einen neuen Eintrag an, indem Sie in der Menüleiste **Objekt | Neu** klicken.
4. Vergeben Sie einen Anzeigennamen und geben Sie den Pfad **CCC/<Name Ihrer HTML-Anwendung>** an.

Verwandte Themen

- [One Identity Visual Studio Code-Erweiterung installieren](#)
- [Kompilierungsbranche auswählen](#)
- [HTML-Anwendungen kompilieren und in Datenbank speichern](#)

One Identity Lösungen eliminieren die Komplexität und die zeitaufwendigen Prozesse, die häufig bei der Identity Governance, der Verwaltung privilegierter Konten und dem Zugriffsmanagement aufkommen. Unsere Lösungen fördern die Geschäftssagilität und bieten durch lokale, hybride und Cloud-Umgebungen eine Möglichkeit zur Bewältigung Ihrer Herausforderungen beim Identitäts- und Zugriffsmanagement.

Kontaktieren Sie uns

Bei Fragen zum Kauf oder anderen Anfragen, wie Lizenzierungen, Support oder Support-Erneuerungen, besuchen Sie <https://www.oneidentity.com/company/contact-us.aspx>.

Technische Supportressourcen

Technische Unterstützung steht für One Identity Kunden mit einem gültigen Wartungsvertrag und Kunden mit Testversionen zur Verfügung. Sie können auf das Support Portal unter <https://support.oneidentity.com/> zugreifen.

Das Support Portal bietet Selbsthilfe-Tools, die Sie verwenden können, um Probleme schnell und unabhängig zu lösen, 24 Stunden am Tag, 365 Tage im Jahr. Das Support Portal ermöglicht Ihnen:

- Senden und Verwalten von Serviceanfragen
- Anzeigen von Knowledge Base Artikeln
- Anmeldung für Produktbenachrichtigungen
- Herunterladen von Software und technischer Dokumentation
- Anzeigen von Videos unter www.YouTube.com/OneIdentity
- Engagement in der One Identity Community
- Chat mit Support-Ingenieuren
- Anzeigen von Diensten, die Sie bei Ihrem Produkt unterstützen