Marketing Cloud Personalization Builder and Einstein Recommendations Implementation Guide

Salesforce, Spring '24

@salesforcedocs

Last updated: November 9, 2023

© Copyright 2000–2024 Salesforce, Inc. All rights reserved. Salesforce is a registered trademark of Salesforce, Inc., as are other names and marks. Other marks appearing herein may be trademarks of their respective owners.

CONTENTS

Introduction 1

Collect Tracking Code 3

Anonymous Versus Known Visitors 4
Install Collect Tracking Code 4
Configure Collect Tracking Code for Personalization Builder 5
Streaming Catalog Updates Through Collect Tracking Code 5
Set User Info and Attributes in Collect Tracking Code 6
Do Not Track Script 7
Set Contact Attribute Affinity 7
Track Page View 8
Track Items Viewed 9
Track Category Viewed 9
Track In-Site Search 9
Track Items in Cart 10
Track Purchase Details 11
Validate Collect Tracking Code Implementation 13

Catalog 14

Einstein Recommendations Catalog Field Definitions 15
Reserved Attributes 18
Add Attributes 19
Configure Custom Attributes 19
Create a Catalog Field Mapping 20
Import a Catalog 21

Einstein Recommendations Status Console 23

Fix an Implementation Error 23

Einstein Recommendations Data Extensions 24

Einstein Recommendations Rule Manager 25

Common Rule Types 25
Configure a Rule 26

Einstein Recommendation Scenarios 27

Scenario Hierarchy 31
Waterfall Predictions 32

Einstein Email Recommendations 33

Create Einstein Email Recommendations 34

Einstein Web Recommendations 39
Create an Einstein Web Recommendation 40
Use Cases 45

INTRODUCTION

The Personalization Builder and Einstein Recommendations Implementation Guide is designed to lead you through the steps used by our own Marketing Cloud subject-matter experts to successfully complete large-scale global implementations.

About This Guide

Personalization Builder, powered by Einstein, delivers the next best product, content, or offer to every individual through product and content recommendations for email and web. Every customer interaction is an insight. With every click, download, view, and purchase, customers are giving data about their preferences. Personalization Builder combines user behavior with algorithms and your unique business rules to build a user profile of affinities. Use that profile to determine the most relevant content and products for each customer in real time.

Who This Guide Is For

This guide is for Marketing Cloud admins, partners, and marketers to implement Personalization Builder.

Before You Start

Personalization Builder must be purchased and provisioned. Contact your account representative for information. Collect and Catalog must be implemented correctly for each use case in order to successfully implement Personalization Builder. Review a list of prerequisites for Personalization Builder implementation.

Services Required

Personalization Builder is a protected-services product. Currently, protected status is limited to building the audience file for Behavioral Triggers.

Introduction

Personalization Builder is an application in Marketing Cloud that includes Web Recommendations and Email Recommendations.

The major components of Personalization Builder discussed in this guide include:

- **Catalog**: The catalog stores all the metadata for the data that's being captured from the website via a daily flat-file upload or JavaScript snippet called Streaming Updates. Every call captured from Collect must match a record in the catalog.
- **Collect**: Collect is a one-pixel JavaScript snippet used to capture data about known subscribers and unknown visitor behavior on your website. It tracks the products and content each visitor views, the products they add to the cart, and the products they purchase. Because Collect captures data at the ID level, it must be tied to a catalog.
- **Personalization Builder Data Extensions**: Personalization Builder data extensions are actionable data extensions that include views, purchases, profiles, and product or content information.

- **Einstein Web Recommendations**: Einstein Web Recommendations provide personalized recommendations based on user behavior. The recommendations are surfaced on websites and mobile apps through an API.
- **Einstein Email Recommendations**: Einstein Email Recommendations use user behavior and machine learning to produce recommendations that are added to the emails via an image and link pair.
- **Status Console**: Status Console monitors the Collect Tracking Code and catalog import for common implementation errors.

Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

COLLECT TRACKING CODE

Collect Tracking Code is a JavaScript snippet you can use to capture data about known contacts and unknown visitor behavior. You can use this data to enhance Contact profiles and optimize customer journeys across channels through predictive and personalized recommendations. You can also update product and content catalog details in real time or to track email-driven revenue.

When a visitor hits your website with the Collect code installed, a cookie is dropped with a unique ID and session ID. The cookie automatically adds an ID to all Collect Calls. The cookie tracks the user until it is removed or cleared.

Collect Tracking Code monitors the variables and events you select at the contact level including:

- Browser type
- User location
- Session time and length
- Page URL
- Page title
- Referrer URL

Configurations vary by product and use case.

Note: The Collect Tracking Code pixel identifies itself as an invisible image and does not affect the user experience of website visitors who use a screen reader.

Considerations:

- All Collect calls associated with items, such as Track Item Detail, Track Cart, and Track Purchase, must match a value in the Catalog. Items not matching in the Catalog is one of the top implementation problems. The Status tab in the Marketing Cloud Email or Web Recommendations application provides insight into whether Collect data matches catalog data.
- Collect calls cannot pass more data about the page that you are viewing, nor can the name-value pairs in the calls be changed. Values like item, unique_id, and email are fixed labels. Never replace fixed labels with custom labels.

Anonymous Versus Known Visitors

When a visitor hits your website with the Collect code installed, a cookie is dropped with a unique ID and session ID. Collect tracks the behavior of anonymous users as long as the cookie is active. Once a user becomes known through an identifier, such as email address, the identifier is added to the anonymous ID. The profile also merges to pull all anonymous data into the known profile.

Install Collect Tracking Code

Install Collect Tracking Code on your website.

Configure Collect Tracking Code for Personalization Builder

Configure Collect Tracking Code for use with Personalization Builder.

Streaming Catalog Updates Through Collect Tracking Code

To configure server-side streaming updates for single or multiple items in your catalog, use the API key or catalog import option. Streaming updates replaces the standard file upload process.

Set User Info and Attributes in Collect Tracking Code

Configure Marketing Cloud Collect Tracking Code to create and maintain a basic user record and add optional subscriber information to a contact profile.

Do Not Track Script

Use Do Not Track Script code to suppress Collect Tracking Code on your website when a subscriber opts out of tracking. Collect Tracking Code doesn't use browser settings to determine whether to track user activity.

Set Contact Attribute Affinity

Use Collect Tracking Code to build subscriber affinity models based on contact attributes.

Track Page View

The trackPageview call is the most common call and can capture three types of information: product or content views, categories views, or search views. No data outside of the item, category, and search name: value pair can be passed in the trackPageview call.

Track Items Viewed

Configure Collect Tracking Code to track items a contact views on your website.

Track Category Viewed

Configure Collect Tracking Code to store category data for the product a contact views on your website.

Track In-Site Search

Configure Collect Tracking Code to store data about a contact's search on your website.

Track Items in Cart

Configure Collect Tracking Code to track items added or removed from a contact's cart.

Track Purchase Details

Configure Collect Tracking Code to store details about a contact's purchases from your website.

Validate Collect Tracking Code Implementation

View the Installation Status console to confirm that your Collect Tracking Code implementation is successful.

Anonymous Versus Known Visitors

When a visitor hits your website with the Collect code installed, a cookie is dropped with a unique ID and session ID. Collect tracks the behavior of anonymous users as long as the cookie is active. Once a user becomes known through an identifier, such as email address, the identifier is added to the anonymous ID. The profile also merges to pull all anonymous data into the known profile.

Choose a contact ID to use with Einstein Recommendations. This ID can never change and currently cannot be updated. The best practice is to use the Marketing Cloud Subscriber Key if possible. This ID is also passed from the website through the Collect JavaScript snippet, so ensure that it is available through the website data layer. As a best practice, we recommend using the Subscriber Key for this reason.

When a known user opens a new device, the profile merges the data from the old cookie to the new cookie and all behavior is maintained. Therefore, anonymous data, and cross-device data, can be tied to one profile as long as they become known at some point through the setUserInfo call.

Currently, the identifier cannot be updated, so if users are sharing computers, they switch back and forth between the two accounts once they are known.

Install Collect Tracking Code

Install Collect Tracking Code on your website.

To begin configuration of Collect Tracking Code, select **Email Recommendations** or **Web Recommendations** under Personalization Builder in the app switcher. Select Collect and complete the steps. Then install Collect Tracking Code on your site.

Note: We recommend that you consult your web developer to complete the installation and configuration of Collect Tracking Code.

1. Copy and paste your Collect Script within the `<head>` or `<body>` in your website template. This step automatically installs the code on every page of the website. Replace MID with your business unit account ID.

   Collect Code Script

   ```
   <script type="text/javascript" src="https://MID.collect.igodigital.com/collect.js">
   </script>
   ```

2. Copy and paste the Set Org ID Script between the body tags of each page of your website. Replace MID with your business unit account ID.

   Set Org ID Script

   ```
   <script type="text/javascript">
   _etmc.push(["setOrgId", "MID"]);
   _etmc.push(["trackPageView"]);
   </script>
   ```

   Tip: Use Options and Configuration within Personalization Builder to modify predictive recommendations features. Changes made in Personalization Builder take effect immediately.

   Note: To enable subscribers to opt out of tracking on your website, replace the Set Org ID Script with Do Not Track Script on page 7.

Configure Collect Tracking Code for Personalization Builder

Configure Collect Tracking Code for use with Personalization Builder.

1. Go to Personalization Builder and select **Web Recommendations** or **Email Recommendations**.
2. Hover over Admin and select **Implementation** from the dropdown.
3. Select your catalog sync method: **Streaming Updates** or **Batch Upload**.
   - For streaming updates, provide a domain name and select the catalog type
   - For batch upload, provide information required for manual catalog upload and update
4. Select the attributes you want to track and copy and paste codes for all attributes to track.

Streaming Catalog Updates Through Collect Tracking Code

To configure server-side streaming updates for single or multiple items in your catalog, use the API key or catalog import option. Streaming updates replaces the standard file upload process.

Important: Streaming updates via Client-Side JavaScript is no longer supported.

To remove an item from the pool of recommendable items, use the API.

- Requests to the server-side streaming updates service via HTTP aren't accepted. Instead, use HTTPS in all requests to the streaming update service.

- API keys sent as URI query parameters aren't accepted. Instead, move the API key into the body of the request.

In these examples, replace MID with the account ID of the business unit used for Collect Tracking Code.

Updating a single item

```
https://MID.collect.igodigital.com/c2/MID/update_item_secure.json
{
  "api_key": "a1234567-cccc-1234-a123-12345a123456",
  "payload": {
    "item": "INSERT_ITEM",
    "unique_id": "INSERT_UNIQUE_ITEM_ID",
    "item_type": "INSERT_ITEM_TYPE",
    "name": "INSERT_ITEM_NAME_OR_TITLE",
    "url" : "INSERT_ITEM_URL",
    "INSERT_ATTRIBUTE_NAME": "INSERT_ATTRIBUTE_VALUE"
  }
}
```

Updating multiple items

```
https://MID.collect.igodigital.com/c2/MID/update_item_secure.json
{
  "api_key": "a1234567-cccc-1234-a123-12345a123456",
  "payload": {
"item": "INSERT_ITEM",
        "unique_id": "INSERT_UNIQUE_ITEM_ID",
        "name": "INSERT_ITEM_NAME_OR_TITLE",
        "url": "INSERT_ITEM_URL",
        "item_type": "INSERT_ITEM_TYPE",
        "INSERT_ATTRIBUTE_NAME": "INSERT_ATTRIBUTE_VALUE"
      },
      {
        "item": "INSERT_ITEM",
        "unique_id": "INSERT_UNIQUE_ITEM_ID",
        "name": "INSERT_ITEM_NAME_OR_TITLE",
        "url": "INSERT_ITEM_URL",
        "item_type": "INSERT_ITEM_TYPE",
        "INSERT_ATTRIBUTE_NAME": "INSERT_ATTRIBUTE_VALUE"

  }
}
```

Set User Info and Attributes in Collect Tracking Code

Configure Marketing Cloud Collect Tracking Code to create and maintain a basic user record and add optional subscriber information to a contact profile.

The setUserInfo push is an important addition to the call by passing the Contact ID. No matter what type of ID is chosen, use the email label for this field and add the value in the SetUserInfo snippet. You do not need to add it to every page because adding it once ties the known identifier to the Collect Tracking Code cookie.

Tip: Configure Collect Tracking Code to store subscriber information when a contact enters an email address on the website or clicks through a marketing email. In order to use the Personalization Builder Integrations, you must use the user's email address. For privacy reasons, we recommend using a non personally identifiable information (PII) identifier or hashing the email.

Set User Info

```
<script type="text/javascript">
   _etmc.push(["setOrgId", "MID"]);
   _etmc.push(["setUserInfo", {"email": "INSERT_EMAIL_OR_UNIQUE_ID"}]);
   _etmc.push(["trackPageView"]);
</script>
```

Tip: Add this call everywhere that a user logs in or becomes known. Some common areas to be watching for include:

- login pages or modals
- signup pages or modals
- login steps, check out process, and so forth
- newsletter signup

1. Copy and customize the script by replacing "INSERT_EMAIL_OR_UNIQUE_ID" with dynamic code to get the Subscriber ID. A Subscriber ID is typically an email address or number unique to that user.
2. Add the modified script in the header code on your pages.

Do Not Track Script

Use Do Not Track Script code to suppress Collect Tracking Code on your website when a subscriber opts out of tracking. Collect Tracking Code doesn't use browser settings to determine whether to track user activity.

This code suppresses all tracking activities, including:

- trackCart
- trackConversion
- trackEvent
- trackPageView
- trackRating

Use this code in place of the standard Set Org ID script:

```
<script type="text/javascript">
   _etmc.push(["setOrgId", "123456"]);
   _etmc.push(["doNotTrack"]);
   _etmc.push(["trackPageView"]);
   </script>
```

To manage privacy settings through API, visit Personalization Builder API developer documentation.

Set Contact Attribute Affinity

Use Collect Tracking Code to build subscriber affinity models based on contact attributes.

Match a contact attribute to a tagged catalog field to increase the subscriber's affinity for the value of that contact attribute. The amount of increase is less than what results from a purchase but more than the increase from a view.

```
<script type="text/javascript">
   _etmc.push(["setOrgId", "MID"]);
   _etmc.push(["setUserInfo", {"email": "INSERT_EMAIL_OR_UNIQUE_ID",
```

```
        "details":    {
        "gender: "female"
                        }
}]);
   _etmc.push(["trackPageView"]);
</script>
```

1. Copy the script and replace `gender` with the contact attribute you want to match.
2. Add the modified script in the header code on the detail pages of your website.

Example: Lisa's marketing director wants to increase the user affinity for catalog items where the gender field value is female for customers who identify as female. Lisa uses the Set Contact Attribute Affinity code to permanently boost any item that has the value female in the gender field.

User Affinity Tag Scenario

Configure a tag scenario to return the most popular items with a particular catalog attribute value. Set the recommendation engine to return items with attribute values that match the top affinities of a contact. This feature can power recommendations like Brands You Love or Items in your Favorite Colors.

1. Highlight the catalog attribute to use.
2. Select **Same as User Affinity** as the value.

Example: Lisa's marketing director wants to increase the user affinity for catalog items where the gender field value is female for customers who identify as female. Lisa uses the Set Contact Attribute Affinity code to permanently boost any item that has the value female in the gender field.

The SetUserInfo call can also include customer attributes specific to the contact on the site. The label can be any value and pulls into the Contact Profile exactly as passed. The value for each pair overwrites any value that is on the profile for the active user.

To use the affinity seeding functionality, the label must match the field in the catalog and the value passed must be assigned to a product on the catalog.

Customer Attributes cannot be exported and currently do not get passed into Marketing Cloud data extensions. The primary use cases for these values are to seed Contact Attribute Affinity and to use these values in rules in influence recommendations. For example, a financial institution adds AgeGroup as Minor for some users, and then creates a rule that prohibits the recommendation engine from recommending items where`OfferType=CreditCard when AgeGroup = Minor`.

Track Page View

The trackPageview call is the most common call and can capture three types of information: product or content views, categories views, or search views. No data outside of the item, category, and search name: value pair can be passed in the trackPageview call.

```
<script type="text/javascript">
   _etmc.push(["setOrgId", "MID"]);
   _etmc.push(["trackPageView", { "item" : "INSERT_PRODUCT_CODE","search" : "XXXXX" }]);

</script>
```

Note: To have the search field populated in IGO_VIEWS, include the search parameter in the tracking call. Use the item parameter to create a record in IGO_VIEWS.

1. Copy the script and replace INSERT_PRODUCT_CODE with the attribute you want to match.

2. Paste the modified script in the header code on your detail pages.

Track Items Viewed

Configure Collect Tracking Code to track items a contact views on your website.

The value for the item must match to the field mapped to the ProductCode or ContentID. If the item ID does not match a record in the catalog, no affinity is built for the item. Mismatched item and field in the catalog is one of the most common mistakes during setup.

```
<script type="text/javascript">
   _etmc.push(["setOrgId", "MID"]);
   _etmc.push(["trackPageView", { "item" : "INSERT_PRODUCT_CODE" }]);
</script>
```

Note: Match the value sent for "INSERT_ITEM" to the Product Code catalog field if you include a catalog file with your Personalization Builder implementation.

1. Copy and customize the script by replacing "INSERT_ITEM" with the dynamic code for the unique identifier of each item.
2. Paste the modified script in the header code on your detail pages.

Track Category Viewed

Configure Collect Tracking Code to store category data for the product a contact views on your website.

The `trackpageview`call is also used on category pages to capture the category viewed by a visitor. This call uses the category name:value pair and the label is reserved.

The category value is flexible but we recommend that you use a consistent type of data. Our best practice is to include values that match tag fields in the catalog. Use the same structure for all type page categorizations, from categories to subcategories to brands.

Only one category is added to the Collect call at a time, so choose the most relevant term for the page being viewed, which is usually the most specific category. If viewing a category page with a breadcrumb of Men's > Apparel > Shoes > Tennis Shoes, the category would often be "Tennis Shoes" and would tie to a "Tennis Shoes" category in the product catalog.

Tip: Consistency is the key to the category call. The key is not required for a successful implementation. If you have category IDs or category names, it needs to match a tagged value in the catalog.

```
<script type="text/javascript">
   _etmc.push(["setOrgId", "MID"]);
   _etmc.push(["trackPageView", { "category" : "INSERT_CATEGORY" }]);
</script>
```

1. Copy and customize the code by replacing "INSERT_CATEGORY" with dynamic code for the category of items viewed.
2. Add the modified script in the header code on your category pages.

Track In-Site Search

Configure Collect Tracking Code to store data about a contact's search on your website.

Note: If not explicitly needed for a particular use-case, the Track In-Site Search call is not required for successful implementation and is oftentimes not implemented.

```
<script type="text/javascript">
   _etmc.push(["setOrgId", "MID"]);
   _etmc.push(["trackPageView", { "search" : "INSERT_SEARCH_TERM" }]);
</script>
```

1. Copy the code and replace "INSERT_SEARCH_TERM" with dynamic code that pulls in the search term.
2. Paste the modified script in the header code on your detail pages.

Track Items in Cart

Configure Collect Tracking Code to track items added or removed from a contact's cart.

Use `trackCart` anytime a shopping cart changes, such as when a visitor adds or removes a product to the cart, or updates the quantity of a product in the cart.

This code overwrites items in the contact's cart with the newest information, including the quantity and unit price for each item. If a cart has multiple products, each product must be added to the call each time it is captured. The active cart must be added to the call every time. This call is critical for Abandon Cart triggers, which uses this call to determine which products to show were abandoned.

The call is made up of 4 name: value pairs, item, quantity, price, and unique_id. Item and unique_id are required.

- The quantity is the number of items added for the particular SKU.
- The price is the price at the time of adding an item to the cart.
- The item, just like in all calls, matches the field mapped to ProductCode in the catalog.
- The unique_id matches the field mapped to the SKUId in the catalog. When these items match it ensures the exact record in the catalog, including all specific attributes like color and size, is tied to the cart rather than just the ProductCode.

Use the `trackCart` each time a product is added to the cart or the cart is updated. This activity often occurs on multiple places on the site including:

- Product Pages
- Category Pages
- Quick Cart
- Cart Page
- Search Results
- Landing Pages
- Home page

Our Best Practice is to include `setUserId`with `trackCart` . This practice helps to ensure that the correct Contact ID is always tied to the cart. It is not a requirement since `setUserInfo` only needs to be set once. However, we highly recommend adding it on all cart calls to ensure that the correct Contact ID is tied to the cart.

Track Cart script:

```
<script type="text/javascript">
   _etmc.push(["setOrgId", "MID"]);
   _etmc.push(["trackCart", { "cart": [
 {"item" : "INSERT_ITEM", "quantity":  "INSERT_QUANTITY" , "price" : "INSERT_PRICE" ,
           "unique_id" : "INSERT_UNIQUE_ID" } ,
 {"item" : "INSERT_ITEM", "quantity":  "INSERT_QUANTITY" , "price" : "INSERT_PRICE" ,
```

```
            "unique_id" : "INSERT_UNIQUE_ID" }
   ]}]);
</script>
```

1. Copy and customize the script.
2. Add the modified script in the header code on your detail pages.

Clear Cart

Use this code to clear all items from a contact's cart when the contact manually removes all items from the cart.

This call is vital for implementations with Abandon Cart triggers, as contact receive unwanted abandon cart emails for products they cleared from their cart.

```
<script type="text/javascript">
   _etmc.push(["setOrgId", "MID"]);
   _etmc.push(["trackCart", { "clear_cart": true } ]);
</script>
```

Note: The trackConversion method clears the contact cart automatically.

Track Purchase Details

Configure Collect Tracking Code to store details about a contact's purchases from your website.

There are two methods for tracking purchase details available in the Marketing Cloud. Track Conversion through Collect Tracking Code is most often used. Alternatively, use an AJAX Reconnector Call to track the same information.

Track Conversion

```
<script type="text/javascript">
   _etmc.push(["setOrgId", "MID"]);
   _etmc.push(["trackConversion", { "cart": [
 {"item" : "INSERT_ITEM", "quantity":  "INSERT_QUANTITY" , "price" : "INSERT_PRICE" ,
            "unique_id" : "INSERT_UNIQUE_ID" } ,
 {"item" : "INSERT_ITEM", "quantity":  "INSERT_QUANTITY" , "price" : "INSERT_PRICE" ,
            "unique_id" : "INSERT_UNIQUE_ID" }],
   // OPTIONAL PARAMETERS
   "order_number" : "INSERT_ORDER_NUMBER",
   "discount" : "INSERT_DISCOUNT",
   "shipping" : "INSERT_SHIPPING",
   "details" : { "AttributeName" : "Value" }
   // END OPTIONAL PARAMETERS
   }]);
</script>
```

1. Copy and customize the Track Conversion script to include the attributes to track.
2. Paste the modified script in the header code on your detail pages. This method requires the same structure as the Track Cart script.
3. Optionally, add order level information including:
   - Order number

- Shipping charges
- Discounts applied
- Custom details

Example: **Example**

If Lisa sets conversion tracking as follows:

```
_etmc.push(["trackConversion", { "cart": [ {"item" : "123", "quantity": "2" ,
"price" : "10.00" , "unique_id" : "123" } , {"item" : "234", "quantity": "1" ,
"price" : "5.00" , "unique_id" : "234" } ],
```

// OPTIONAL PARAMETERS

"order_number": "123456",

"discount": "2.00",

"shipping": "5.00"

// END OPTIONAL PARAMETERS

}]);

The resulting values stored in the data extension are:

SKU price quantity order_number

123 11.2 2 123456

234 5.6 1 123456

The formula used to calculate price is:

price + ((shipping - discount) * $ percentage of order/quantity)

For SKU 123, which accounts for 80% of the $25 order, it looks like: 10 + ((5-2) *.8/2) = 10 + (3 * .4) = 11.20

AJAX Redirector Call

Use an AJAX call to track order detail information not available on your confirmation screen.

Example:

```
<html><a href="http://ntoutfitters.com/product/4064100"
onclick="igoClickTracker('http://stack9.collect.igodigital.com/redirect/v2eyJuIVsbGVyc1J1Z2lvbiIsInUiOiI0MDkzMzQyNiIsInIiOjUxNDUsInB...=')">Crestone
 75 Pack</a>
<script type="text/javascript">
function igoClickTracker(click_url) {
 var xmlhttp;
 if (window.XMLHttpRequest) {
     // code for IE7+, Firefox, Chrome, Opera, Safari
     xmlhttp = new XMLHttpRequest();
 } else {
     // code for IE6, IE5
     xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
 }

 xmlhttp.open("POST", click_url , true);
```

```
 xmlhttp.send();
}
</script>
```

Validate Collect Tracking Code Implementation

View the Installation Status console to confirm that your Collect Tracking Code implementation is successful.

To access the Installation Status console, click the **Status** tab inside Personalization Builder. Current statistics from this tracking system are used to optimize performance on the site.

Tip: Go to your site in the same browser to view a product, add it to a cart, and make a purchase before you attempt to validate the code. If your site has an email intake field, enter an email address.

After configuring a catalog, and deploying Collect Tracking Code to your website, proper installation of Collect Tracking Code is essential for all Personalization Builder use cases.

1. Browse your website, ensuring that you visit each class of page on which you've installed Collect Tracking Code. For example, visit the home page, navigate to a product detail page, add it to your cart, and purchase it.
2. Log in. Select Email Recommendations or Web Recommendations under Personalization Builder.
3. Click the Status Tab.
4. The Status Console provides a lens into today's traffic on each class of page. Ensure that the counts for each type of page have incremented as you expect them.
5. The Status Console also shows common implementation problems and provides details on how to fix them.
6. You can subscribe to notifications to receive an email if data collection stops on a particular class of page. The notification alerts you when Collect Tracking Code is broken on a page.

CATALOG

A catalog includes all fields that contain attributes used to make Einstein Recommendations. Einstein Recommendations filters assets based on these attributes. For instance, you can define assets in the recommendation pool based on a product's color or size, or an article's author and subject matter.

Note: Einstein Recommendations requires a minimum number of attributes, which vary depending on the application or specifications.

The catalog functionality is used for multiple purposes within Einstein Recommendations. Here are the major areas to consider when consulting on what fields to add to a catalog:

- Support Collect: The catalog data passed via Collect ties to the catalog in the data via the Product Code (item) and SkuID (unique_id). These fields must match the Collect data to ensure proper function.
- Build User Profiles: When collecting user-data, the catalog takes the view ID and uses the tagged fields to build the profile.
- Support Recommendation Generation: Any data from the catalog can be used to build the recommendation. The catalog is the only access to the metadata used to produce recommendations.
- Locale and Language Support: If any attributes vary by a locale or language, add the field to the catalog.
- Support Attribute-Based Rules: Data that is in the catalog can support rule generation. For example, if you want only Men's clothes shown in a particular recommendation, a field like "gender" could support this rule. If a data type is not in the catalog, rules cannot be generated for it.

Marketing Cloud accepts three types of catalogs:

- Product: Products available for purchase
- Content: Articles, blogs posts, videos, or other types of content available for view
- Banner: Image files such as hero graphics, calls to action, or offers

You can import one file for each catalog type.

[1]

Einstein Recommendations Catalog Field Definitions

Catalog fields store your catalog's asset attributes for Marketing Cloud Einstein Recommendations. These attributes are both required and recommended catalog fields. You can also add custom attributes. Catalog field values are limited to 1024 single-byte characters or 512 double-byte characters.

Reserved Attributes

See a list of attributes that are reserved for specific use in Einstein Recommendations. You cannot use these attributes to store other values.

Add Attributes

To add an attribute in Einstein Recommendations, create a field at the end of the default field list.

Configure Custom Attributes

Configure custom attributes to use for Einstein Recommendations.

[1] Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

Create a Catalog Field Mapping

Create a catalog field mapping in Einstein Recommendations to ensure that catalog heading names are mapped to the required application field names.

Import a Catalog

Import a catalog to use in Einstein Recommendations as a pipe-delimited value (PDV) flat-file format. The first row of data must be the column headings and each line contains non-breaking spaces. The catalog import file contains at least two rows of catalog data in addition to the headings. We recommend daily catalog updates to reflect the changes on the site by the next business morning. You can arrange an alternative schedule. For improved reliability, image size is limited to 2 MB when using the CSV catalog import option.

Einstein Recommendations Catalog Field Definitions

Catalog fields store your catalog's asset attributes for Marketing Cloud Einstein Recommendations. These attributes are both required and recommended catalog fields. You can also add custom attributes. Catalog field values are limited to 1024 single-byte characters or 512 double-byte characters.

Catalog Requirements

- Leave the catalog field blank rather than populating it with a Null value.
- Text fields can't include carriage returns, new lines, or HTML.
- All characters in the catalog must either be UTF-8 encoded or converted to valid HTML, except for the pipe and tilde reserved characters.
- Items with a price of $0 are recommended and displayed with the price of $0.
- The final catalog file must be automated and uploaded daily with a unique static file name.

Product Catalog

Field Name	Required	Attribute Description
SkuID	No	Unique identifier for each product, specific to the customer. This ID must match the product SKU on the website and be passed via Collect Tracking Code.
ProductCode	Yes	Used to identify a group of related products. For example, a piece of apparel appears in the catalog multiple times with a unique SkuID for each size the product is available in. All these size variations are sent with the same ProductCode. The ProductCode value cannot contain spaces or special characters.
ProductName	Yes	Used to identify a product in the title of product detail pages.

Field Name	Required	Attribute Description
ProductType	Yes	Product category is open to retailers. The application maps your product types to our internal catalog system.
ProductLink	Yes	URL to reach the product detail page.
ImageLink	No	URL to the product image displayed in the recommendation.
RegularPrice	No	Regular price for the product. You can display this value in strikethrough text for a recommendation when a sale price is available.
SalePrice	No	Sale price for the product. It displays if a product is on sale.
OnlineAvailability	Yes	Indicates if a product is available for recommendation. This field only accepts two possible values, Y for available or N for not available. If the field is not sent with the feed, the default value is Y. If it is sent as N, the product is not returned in recommendations.

Content Catalog

Field Name	Required	Attribute Description
ContentID	Yes	Unique identifier for each piece of content, specific to the customer. This ID matches the content ID on the website and is passed via Collect Tracking Code.
Title	No	Title of content included in recommendations.
Link	Yes	URL to reach the content detail page.
ImageLink	No	URL of the image displayed in the recommendation.
Author	No	Author of specific content used to influence recommendations or to be displayed within the recommendations.
Category	No	Content category or categories.
Availability	Yes	Indicates if content is available for recommendation. This field only accepts two possible values, Y for available or N for

Field Name	Required	Attribute Description
		not available. If the field is not sent with the feed, the default value is Y. If it is sent as N, the content is not returned in recommendations.
Publication Date	No	Date of publication to the site. Use data format of YYYY-MM-DD.
Keywords	No	Keywords that describe the content.

Banner Catalog

Field Name	Required	Attribute Description
BannerID	Yes	Unique identifier for each banner or offer, specific to the customer. This ID matches the banner ID on the website passed via Collect code.
GroupID	Yes	Used to identify a group of related banners. For example, the same banner could be available in multiple sizes. Each size of the banner must be sent with a separate BannerID, and is linked together with a matching GroupID.
Name	No	Used for easy identification and content management.
Category	No	Offer category, aligned to the catalog values sent for product and or content categories.
Link	Yes	URL to reach the banner or offer detail page.
ImageLink	Yes	URL to the banner image displayed in the recommendation.
LocationSize	Yes	Size of the banner location. This value can be a size value, such as 600x400, or a name value, such as HeroHome.
PPC	No	Price Per Click used to calculate Key Performance Indicators. Round to the nearest 2 decimal places and send without any currency symbols.
Availability	Yes	Indicates if the banner available for recommendation. This field only accepts two possible values, Y for available or N for not available. If the field is not sent with the

Field Name	Required	Attribute Description
		feed, the default value is Y. If it is sent as N, the banner is not returned in recommendations.
StartDate	No	Banners are automatically excluded from the recommendation pool before the start date. Use a suggested date format of YYYY-MM-DD.
EndDate	No	Banners are automatically excluded from the recommendation pool on the end date. Use a suggested date format of YYYY-MM-DD.
Keywords	No	Used to describe the banner or offer.

Reserved Attributes

See a list of attributes that are reserved for specific use in Einstein Recommendations. You cannot use these attributes to store other values.

Attribute Name	Reserved Use
ReleaseDate	The first date an item is available for recommendations. This field is required for new arrivals recommendations. The recommended format for this field is YYYY-MM-DD.
Rating	The user rating displayed for an item. This field is used to create ratings-based recommendations.
NumReviews	The number of reviews an item received. This field is used to create ratings-based recommendations based on items customers are talking about.
Keywords	Keywords are used to match attributes to the customer's interests.
Description	This field contains the full description of an item included in recommendations. It must be plain text.
ProductFamily	This field is used to group multiple ProductCodes; for example, the ProductFamilyID can include 5 colors of a specific style of t-shirt. These 5 colors are sent in the catalog as 5 unique ProductCodes, with 6 SkuID's each.
Accessories	A list of SKU IDs or ContentIDs defined by the manufacturer or content team, used to create business rules. For example, these products can be excluded from recommendations on the detail pages if they are already being served to the user on this page. Use a tilde (~) format for this list.

Add Attributes

To add an attribute in Einstein Recommendations, create a field at the end of the default field list.

Attributes are data fields containing product or content characteristics such as color, dimension, brand name, subject, or author.

1. Enter the *Attribute Name* at the end of the column heading. Do not include spaces.
2. Specify the attribute value at the end of the attribute value list.

   If an attribute has multiple values, separate them with a tilde (~). Values are limited to 1024 single-byte characters or 512 double byte characters.

Product Record

Example: `SkuID|ProductCode|ProductName|ProductType|ProductLink|ImageLink|RegularPrice|SalePrice|OnlineAvailability|ProductCost|ProductFamilyID|ProductDescription|BrandName|Manufacturer|Color|Material|Rating|NumReviews|ReleaseDate| Keywords`
`12345|XCode12345|Brand X Product Whiz 9000|Electronics|http://www.companyx.com/product|http://www.companyx.com/product/image.jpg|99.99||Y|59.99|YCode12345|An amazing product.|Brand X|Company X|Blue~Red|Metal|4|25|2012-08-27|Technology~Music`

The current way to delete a product is to change the product's Online Availability attribute within the Catalog prior to upload. In the example above, change the **Y** in the field that preceeds 59.99 to **N**. After you upload the catalog file, the N value will prevent the product from showing up in product recommendations.

Content Record

Example:
`ContentID|Title|Link|ImageLink|Author|Category|PublicationDate|Keywords|Availability`
`12345|Crock Pot Lasagna Soup|http://www.companyx.com/lasagna_soup|http://www.companyx.com/soup/image.jpg||Recipes|2014-01-01| Italian~Soup~Slow Cooker|Y`

Banner Record

Example: `BannerID|GroupID|Name|BannerType|Link|ImageLink|LocationSize|Category|PPC|StartDate|EndDate|Keywords|Availability 12345|GRPABC|Free Shipping on 3+ Cookbooks|Shipping|http://www.companyx.com/offer|http://www.companyx.com/offer/image.jpg|600x400|Cookbooks~Slow Cooker|0.0025|2014-01-01|2014-03-31||Y`

Configure Custom Attributes

Configure custom attributes to use for Einstein Recommendations.

Unmapped fields are automatically used by the Marketing Cloud to make recommendations. For Streaming Updates, Custom Attributes Configuration is available after Marketing Cloud starts receiving data from your site.

Consider using these types of fields tags:

- Categories and Sub-Categories - . Mens, Womens, Kids, Coats, Shoes
- Brands - . Nike, Adidas, Under Armour
- Seasons - . Fall, Spring
- Topics - . Politics, Sports, Lifestyle
- Interests - Hiking, Golf, Cosmetics

Avoid these tags:

- Binary Values - . Binary values, such as Men or Women, could be good tags. We recommend reserving this option for the most impactful values.
- Sizes or Colors and Other SKU Level Attributes- The recommendation engine aggregates behavior at the Product Code level. These attributes on the SKU level could slant the results.

1. In Einstein Recommendations, hover over Admin and select **Implementation**.
2. Scroll to Custom Attributes.
3. Click **Custom Attribute Configuration**.
4. Select all fields included in the catalog.
5. **Save**.

Note: You receive an email when this configuration is enabled.

2

Create a Catalog Field Mapping

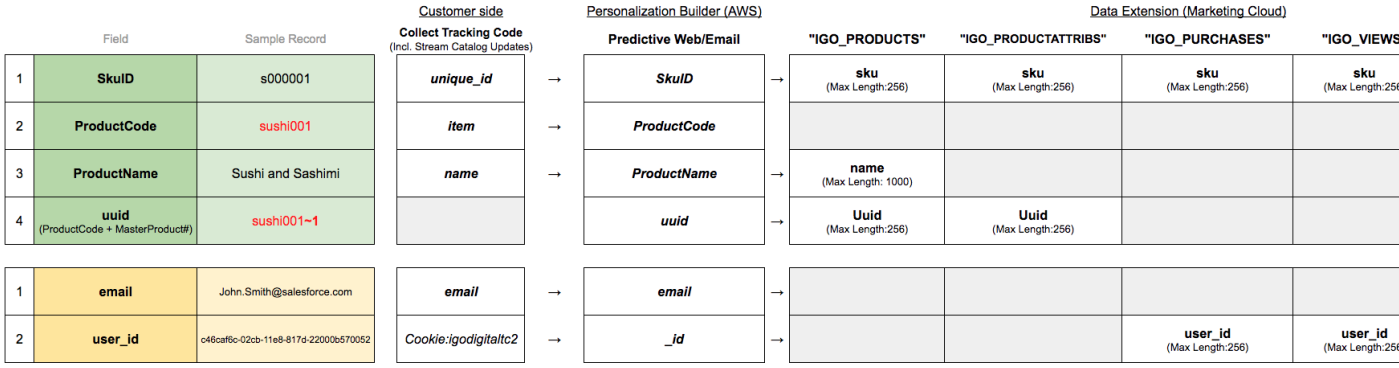
Create a catalog field mapping in Einstein Recommendations to ensure that catalog heading names are mapped to the required application field names.

1. Click **Field Mapping** or click the gear icon and select **Mapping**.
2. Review the fields for mapping accuracy.
3. To change, expand the selection box under Your Headings and change the selections listed in the catalog heading names.

   Unique catalog headings are required for each field. Duplication of a heading automatically deletes the heading from the previously mapped field.
4. For **Additional Fields**, indicate if multiple values are accepted.
5. Select the tagged fields.
6. Select fields to be used for personalization.
7. Select a data format that matches the date field in your catalog.
8. Click **Save**.

[2] Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

Example:



	Field	Sample Record	Customer side: Collect Tracking Code (Incl. Stream Catalog Updates)		Personalization Builder (AWS): Predictive Web/Email		"IGO_PRODUCTS"	"IGO_PRODUCTATTRIBS"	Data Extension (Marketing Cloud): "IGO_PURCHASES"	"IGO_VIEWS
1	SkuID	s000001	*unique_id*	→	*SkuID*	→	sku (Max Length:256)	sku (Max Length:256)	sku (Max Length:256)	sku (Max Length:256
2	ProductCode	sushi001	*item*	→	*ProductCode*					
3	ProductName	Sushi and Sashimi	*name*	→	*ProductName*	→	name (Max Length: 1000)			
4	uuid (ProductCode + MasterProduct#)	sushi001~1			*uuid*	→	Uuid (Max Length:256)	Uuid (Max Length:256)		
1	email	John.Smith@salesforce.com	*email*	→	*email*	→				
2	user_id	c46caf6c-02cb-11e8-817d-22000b570052	*Cookie:igodigitaltc2*	→	*_id*	→			user_id (Max Length:256)	user_id (Max Length:256

Import a Catalog

Import a catalog to use in Einstein Recommendations as a pipe-delimited value (PDV) flat-file format. The first row of data must be the column headings and each line contains non-breaking spaces. The catalog import file contains at least two rows of catalog data in addition to the headings. We recommend daily catalog updates to reflect the changes on the site by the next business morning. You can arrange an alternative schedule. For improved reliability, image size is limited to 2 MB when using the CSV catalog import option.

Note: Merge data from disparate sources before importing into Marketing Cloud. The Marketing Cloud data services team can help with data preparation. Contact your account representative for details.

The two catalog import options are:

- Daily Flat-File Upload. The daily flat file upload is added to an FTP account or publicly accessible web URL and imported into Einstein Recommendations at the same time daily. A full upload of all products and a daily delta file is supported.
- Streaming Updates. The Streaming Update option adds and updates products and content through another snippet of Server-Side JavaScript. All data must be available in the web layer sent in each call.

The data that is passed is the same no matter which method is chosen.

Tip: If you opt to use streaming updates and batch upload concurrently, both sources must contain the same data to avoid one overwriting the other.

Visit Personalization Builder Catalog Field Definitions on page 15 for a list of required catalog fields.

Product Recommendation customers often have a product catalog already generated and decide to reuse or fashion the daily import on something existing. Since the catalog data includes products, attributes, rules, and multiple languages the flat file catalog is often the more appropriate choice. If a specific Product Code has numerous SkuID, the streaming update call includes data for each variation, which can be a rather large payload. You can use Streaming Updates with Product Recommendations when all the data is available.

Content Recommendation clients are less likely to have a pre-created content catalog and often choose Streaming Updates. The data layer of the website or CMS contains the content to add to the catalog and is easily added via JavaScript.

There's no direct functionality to delete a product recommendation. The current way to delete a product is to change the product's Online Availability via Attributes within the Catalog before upload. Set **Online Availability** for the product to **N** within the Catalog. This action prevents the product from showing up in recommendations.

There are two ways to name the file to be uploaded daily. A static filename or a wildcard.

- Static Filename: The static filename is uploaded daily. We recommend that you replace it on the FTP before the upload kicks off. If there are no changes in the file or a new one isn't uploaded, the catalog import is skipped. An example of static filename would be product_catalog.csv.

- Wildcards in Filename: You can add a wildcard (*) at the end of the filename that allows a new file name to be updated daily. Add the date for the file at the end of the file and replace it with an asterisk (*) when adding to the upload UI.

Tip: We recommend that you use the naming format *static_filename_YYYY_MM_DD.xxx*. For example, `product_catalog_2018_01_01.csv`. The filename is added to the upload as `product_catalog_*`. The filename is sorted in descending order, so we recommend adding the year first to ensure that it sorts correctly.

1. In Email Recommendations or Web Recommendations, hover over Admin and select **Implementation**.
2. Select your catalog sync method.
3. To use streaming updates, enter your website domain name and click **Register**. If you don't want to use streaming updates, you can skip this step.
4. Choose a catalog type.
5. Select the format of your catalog.
6. Name your catalog.
7. Indicate if it's a partial catalog.

   Tip: To upload multiple files, use an asterisk in place of part of a file name as a wildcard. For example, to import catalog_file_20130101.csv, catalog_file_20130102.csv, and catalog_file_20130103.csv, you can enter catalog_file_*.csv. The system sorts the files in descending order and imports the top result. Use the YYYY_MM_DD date format when using dates in wildcards to ensure that imports include the latest file.

8. To receive import status notifications and indicate the frequency of automatic notifications, enter an email address.
9. Click **Upload New File**.
10. To import more catalogs, repeat these steps.

    Tip: You can cancel a catalog import before completion.

    Note: You can only import one product, content, and banner catalog. When you do, previously imported catalogs of the same type are overwritten.

EINSTEIN RECOMMENDATIONS STATUS CONSOLE

Marketing Cloud Einstein Recommendations Status Console monitors the Collect Tracking Code and catalog import for common implementation errors. View where an error occurred and details to help resolve the issue.

This feature helps ensure that the Einstein Recommendations engine receives the data necessary to power optimal Einstein Recommendations and behavioral triggers for campaigns and journeys.

Many errors or a spike in errors indicates that something changed. For example, a catalog import failed to import with new products. These changes generate errors in views, carts, and conversions because they don't match. If there is an issue to resolve, an alert appears in the status column. Click the link in the description column to view the error and take steps to fix the issue.

The Status Console shows incoming Collect Tracking Code data by call type. For more details about Collect Tracking Code types, review Collect Tracking Code on page 3.

Currently, the number of products with errors is limited to 100 SKUs. If there is a major problem with hundreds of errors, we recommend that you check multiple times for new errors. There is no notification when there's an increase in errors or decrease in traffic.

Tip: We recommend that you review error rates on a weekly or monthly basis.

Fix an Implementation Error
Follow these steps to fix an implementation error from the Einstein Recommendations status console.

Fix an Implementation Error

Follow these steps to fix an implementation error from the Einstein Recommendations status console.

Complete these steps for each alert.

Note: If you only click **Mark as Resolved**, the error is not fixed. Follow the instructions in the window that opens from the error link. If the error reoccurs, the alert reappears in the status console.

1. Click the error link in the Details column.
2. Click **Take these steps to troubleshoot error**.
3. Complete the steps in the pop-up window.
4. Click **Mark as Resolved**.
   The alert in the status console changes to OK.

EINSTEIN RECOMMENDATIONS DATA EXTENSIONS

Data captured through Collect is stored on Einstein Recommendations servers and is not, by default, added to Marketing Cloud Data Extensions. Once Catalog and Collect are installed, the flow of data from Einstein Recommendations to Marketing Cloud Data Extensions can be enabled.

Depending on the type of implementation, different data extensions receive data. Profiles, Views, Purchases, Products, Product Attributes receive data from a Product Catalog, Content Attributes, Content Views, and Content receive data from a Content Catalog.

Currently, Abandon Cart Items and Abandon Cart Events contain no data. These data extensions are inactive and the Abandon Cart Trigger is the only option to get this data into Marketing Cloud Data Extensions. Session Ends is used for Discover reporting and is not on by default.

Once you set up Collect and Catalog, you can turn on Standard Data Extensions from the Status Console.

You can find all the Standard Data Extensions fields and schema here.

EINSTEIN RECOMMENDATIONS RULE MANAGER

Use Rule Manager to refine Einstein Recommendations for email and web.

You can create rules for these things:

- All web and email recommendations.
- Specific SKUs with the ability to search the attribute field within the catalog.
- Product or content release date using the release date field in the catalog.
- Greater-than or less-than rules for dates and numbers (specifically prices) and ratings.

Contact Events are conditional statements to fulfill actions in rules only when the user meets specified criteria: Past purchase includes a chosen attribute value. Past view includes a chosen attribute value. Last visit was greater than or less than X days ago.

Content Attributes are conditional statements to fulfill actions in rules only when the content being viewed meets specified criteria: All catalog attributes available. Must reference item SKU ID in recommendation call to work properly. Attributes are case-sensitive.

Note: If you use a SKU ID that isn't a master product in your catalog, the images don't generate correctly. Image generation is applied only for master products.

The Actions section is the heart of the Rule Manager and lets you add business logic to the recommendation engine. The Rule Manager limits or filters out recommendations based on a condition or makes certain characteristics more or less likely to be recommended within a call.

[3]

Common Rule Types

To create your own algorithms, use rules in the Einstein Recommendations Rule Manager tool.

Configure a Rule

Configure a rule to refine Einstein Recommendations.

Common Rule Types

To create your own algorithms, use rules in the Einstein Recommendations Rule Manager tool.

Tip: Rules can override top performing algorithms and can have a negative impact on predictive performance. We recommend applying them sparingly.

Scope	Rule	Syntax
Global Rule		
	Never recommend gift cards to customers.	ACTION Exclude where <GiftCard> <is> <Y>
	Recommend highly rated items to users.	ACTION Emphasize Items with <Ratings> <Greater Than> <3>

[3] Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

Scope	Rule	Syntax
Home Page Rule		
	Emphasize the newest items on the home page.	ACTION Emphasize Items with <ReleaseDate> <Less Than> <30> Days Ago
Category Page Rules		
	Recommend only items with the same category as the landing page the subscriber views.	ACTION Include Items with <Category> <is> <Same as Reference Item>
Detail Page Rule		
	Only recommend items of the same gender as the item the subscriber views.	ACTION Include <Gender> <is> <Same as Reference Item>
	Only recommend Sale items when the subscriber view a sale item.	ATTRIBUTE <isOnSale> <is> <False> ACTION Exclude <isOnSale> <is> <True>
	Exclude items based on intended age group.	ATTRIBUTE <AgeGroup> <is> <Child> ACTION Exclude <Inappropriate> <is> <True>
Exclude items based on intended age group.		
	Do not recommend a competing brand on a detail page.	ATTRIBUTE <Brand> <is> <Nike> ACTION Exclude <Brand> <is> <Adidas>
	Only recommend items of the same brand as the item that is being viewed.	ACTION Include <Brand> <is> <Same as Reference Item>

Configure a Rule

Configure a rule to refine Einstein Recommendations.

Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

EINSTEIN RECOMMENDATION SCENARIOS

These scenarios help you produce Einstein Recommendations for email or the web in Marketing Cloud.

Tip: To apply custom logic to a scenario, use Rule Manager.

Reference Item Scenarios

These scenarios require one or more specific reference items. These items typically include a value passed in a SKU or CATEGORY parameter via a recommendation call or email HTML content.

Scenario	Description	Catalog Field Requirement	Collect Field Requirement
Bought Bought	Subscribers or visitors who bought the item passed in the parameter also bought these items.	SKU ProductCode	trackConversion
Bought Bought Merge	Subscribers or visitors who bought any of the items passed in the parameter for this cart also bought these items.	SKU ProductCode	trackCart trackConversion
Click Bought	Subscribers or visitors who clicked the item passed in the parameter also bought these items.	SKU ProductCode	trackPageView trackConversion
Category Bought	Subscribers or visitors who viewed an item in the category attribute passed in the `category=` parameter also bought these items.	SKU ProductCode	trackPageView trackConversion
Category View	Subscribers and visitors who viewed an item in the category attribute passed in the `category=` parameter also viewed these items.	SKU ProductCode ContentID	trackPageView
Search Bought	Subscribers or visitors who used the search term passed in a `search=` parameter bought these items.	SKU ProductCode	trackPageView trackConversion
Search View	Subscribers or visitors who used the search term passed in a	SKU ProductCode	trackPageView

Scenario	Description	Catalog Field Requirement	Collect Field Requirement
	`search=` parameter viewed these items.	ContentID	
Tag	The most popular item based on purchases, clicks, or views with the attribute passed in the `category=` parameter. When no attribute is passed, but the `item` parameter is included, the system chooses a reference item tag value at random.	SKU ProductCode ContentID Attribute	trackPageView
Tag Intersection	Most popular items with an intersection of attributes that match the attributes of the parameter value for the item viewed by the subscriber or visitor.	SKU ProductCode ContentID Attribute	trackPageView
Tag Recent	Newly available items with the attribute passed in the `category=` parameter. When no attribute is passed, but the `item` parameter is included, the system chooses a reference item tag value at random.	SKU ProductCode ContentID Attribute Release Date	trackPageView
Tag Enjoyed	Most viewed and purchased items with the attribute passed in the `category=` parameter. When no attribute is passed, the system chooses a reference item tag value at random. Use at least one Collect field.	SKU ProductCode ContentID Attribute	trackPageView trackCart trackConversion
Tag Top Rated	Top rated item with the attribute passed in the `category=` parameter. When no attribute is passed, the system chooses a reference item tag value at random.	ContentID Attribute Rating	trackPageView
View Bought	Subscribers or visitors who viewed the item passed in the parameter also bought these items.	SKU ProductCode	trackPageView trackConversion
View View	Subscribers or visitors who viewed the item passed in the	SKU ProductCode	trackPageView

Scenario	Description	Catalog Field Requirement	Collect Field Requirement
	parameter also viewed these items.	ContentID	
View View Merge	Subscribers or visitors who viewed any of the items passed in the parameter for this cart also viewed these items.	SKU ProductCode	trackPageView trackCart

User Data Scenarios

These scenarios are based on a user's Predictive Intelligence profile and personal site behavior.

Scenario	Description	Catalog Field Requirement	Collect Field Requirement
Bought Bought Last Cart	Subscribers or visitors who bought the last item added to the cart also bought these items.	SKU ProductCode	trackCart trackConversion
Bought Bought Last Purchase	Subscribers or visitors who bought the item last purchased also bought these items.	SKU ProductCode	trackConversion
Favorite Tag	Most popular items with attributes that match the subscriber's or visitor's interest based on their affinity profile. These recommendations are based on the user's top affinity. Returns the most popular items that match the single top tag on their weighted affinity graph.	SKU ProductCode ContentID Attributes	trackPageView
Last Views Merge	Subscribers or visitors who viewed any of the last 7 items viewed by this subscriber or visitor also viewed these items.	SKU ProductCode ContentID	trackPageView
Recently Viewed	Any of the last 7 items viewed by the subscriber or visitor.	SKU ProductCode ContentID	trackPageView
User Affinities	Recommendations based on the user's complete affinity profile. Returns items that match the top five affinities listed in their contact profile, based on purchases, clicks, and views over	SKU ProductCode ContentID Attributes	trackPageView

Scenario	Description	Catalog Field Requirement	Collect Field Requirement
	time. Preference is given to items that are tagged with multiple attributes on the top five affinity list.		
View Bought Last Cart	Subscribers or visitors who viewed this item in a cart also bought these items.	SKU ProductCode	trackPageView trackCart trackConversion
View Bought Last Purchase	Subscribers or visitors who viewed the item last purchased by this subscriber or visitor also bought these items.	SKU ProductCode	trackPageView trackConversion
View View Last Cart	Subscribers or visitors who viewed this item in the cart also viewed these items.	SKU ProductCode	trackPageView trackCart
View View Last Purchase	Subscribers or visitors who viewed the item last purchased by the subscriber or visitor also viewed these items.	SKU ProductCode	trackPageView trackConversion

Site Data Scenarios

These scenarios are based on the collected aggregated site behavior. No information is personalized at the user level.

Scenario	Description	Catalog Field Requirement	Collect Field Requirement
New Releases	This information includes 100 items across the site where the ReleaseDate parameter is in the past 14 days. The values are sorted by newest date.	SKU ProductCode ContentID ReleaseDate	trackPageView
Top Enjoyed	Most popular products across the site based on a combination of viewed and purchased items.	SKU ProductCode ContentID Attribute	trackPageView trackCart trackConversion
Top Grossing	Highest grossing items in terms of total revenue generated.	SKU ProductCode	trackConversion

Scenario	Description	Catalog Field Requirement	Collect Field Requirement
Top Sellers	Top selling products across the site based on units sold.	SKU ProductCode	trackConversion
Top Views	Up to 100 of the most viewed items across the site.	SKU ProductCode ContentID	trackPageView

[4]

Scenario Hierarchy
Scenarios are the specific type of recommendations returned. By default, scenarios appear in order by hierarchy, and the scenario must fill all recommendations to be returned. For example, you configure a display to show four recommendations, but the first scenario produces only three. To try to fill the remaining recommendation, the system moves to the next scenario in the hierarchy .

Waterfall Predictions
When you enable waterfall predictions in Einstein Recommendations, the system begins with your top priority scenario and continues down the list to your lowest priority scenario. At each step, it returns all available matching items until it reaches the total number of recommendations you set.

Scenario Hierarchy

Scenarios are the specific type of recommendations returned. By default, scenarios appear in order by hierarchy, and the scenario must fill all recommendations to be returned. For example, you configure a display to show four recommendations, but the first scenario produces only three. To try to fill the remaining recommendation, the system moves to the next scenario in the hierarchy .

If no scenarios return enough recommendations, Einstein Recommendations includes filler recommendations based on best sellers or top content. Scenario ordering is more art than science, but here are the general rules.

- Specific to General: Start with the most personal, specific scenario and move to the most generic.
- Purchase > Cart > View: Think through what is most important and shows the most intent.
- Tag Scenarios: To use a specific scenario order, consider tag scenarios.
- User Affinity: To generate recommendations based on a subscriber's top five affinities, for example, add User Affinity. A recommendation situation without a specific context, such as a home page or weekly newsletter, is a good candidate for the top user affinities scenario. On a cart or product page, it appears below the product and above the generic scenarios.
- Every Business Is Different: We offer so many scenarios because different use-cases and business types work best with different hierarchies.
- Always Be Optimizing: Einstein Recommendations reports appear segmented by scenario to provide insight into which scenarios perform the best.

[4] Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

Waterfall Predictions

When you enable waterfall predictions in Einstein Recommendations, the system begins with your top priority scenario and continues down the list to your lowest priority scenario. At each step, it returns all available matching items until it reaches the total number of recommendations you set.

Example: You set the recommendation number to four items. However, your top priority scenario only returns one recommendation. The system moves to the next scenario on the list, and then the next, adding recommendations until there is a total of four recommended items.

Note: If you choose not to enable Waterfall Predictions, the system only returns recommendations from the first algorithm that provides the total number of recommendations you set.

EINSTEIN EMAIL RECOMMENDATIONS

Use Einstein Email Recommendations to observe customer behavior, build preference profiles, and deliver the next-best content or product. You can also refine recommendations to match your business rules. Recommendation content surfaces in placeholders for recommendations on your website.

Email recommendations are delivered as a pre-built image and link pair and are activated when a user engages with the email. We refer to email recommendations as engagement recommendations, meaning the logic and scenarios are processed immediately before they see the recommendation.

The image in the recommendation is based on an HTML template that is used to style what one recommendation would look like. Each product or content recommendation is generated ahead of time and stored to ensure that the image is instantly available. By pre-processing displays and logic for each product or content recommendation, the most up-to-date recommendation is delivered without sacrificing performance when a subscriber engages with the email.

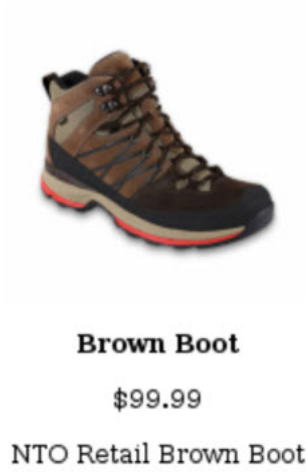
Here's the required HTML that that is included in an email template for a block with two recommendations.

```
<a href="https://MID.recs.igodigital.com/rr/v2/HASH/l/1/%%emailaddr%%">
<img src="https://MID.recs.igodigital.com/rr/v2/HASH/i/1/%%emailaddr%%">
</a>
<a href="https://MID.recs.igodigital.com/rr/v2/HASH/l/2/%%emailaddr%%">
<img src="https://MID.recs.igodigital.com/rr/v2/HASH/i/2/%%emailaddr%%">
</a>
```

Here's the HTML to create the image used in this call.

```
<div style="font-size:12px;color:black;width:150px; text-align:center;">
<img style="max-width:150px;" src="%%ImageLink%%">
<p><b>%%ProductName%%</b></p>
%%[ if SalePrice &lt; RegularPrice then ]%%
<s>Regular Price: %%RegularPrice%%</s><br>
<span style="color:red;">Sale Price: %%SalePrice%%</span>
%%[ else ]%%
%%RegularPrice%%
%%[ endif ]%%
<p>%%Description%%</p>
</div>
```

Here's an example of the image generated and called via the link in the email recommendations. The link redirects to the page on the site while tracking the click:



Each recommendation is an individual image, so if you needed four recommendations in a call, the HTML would include four image and link pairs. The display of the recommendation is unified across the entire recommendation, so with engagement recommendations, all four recommendations use the same HTML and AMPscript. Because email recommendations are displayed as an image, consider the mobile responsiveness of the design.

For information about how recommendations are applied to messages, see Multipliers for Marketing Cloud.

[5]

Create Einstein Email Recommendations
Follow these steps to create Einstein Email Recommendations.

Create Einstein Email Recommendations

Follow these steps to create Einstein Email Recommendations.

Note: When a customer profile privacy is set to one the following, the system generates generic recommendations from the scenarios you configured.

- Do Not Track
- Do Not Profile
- Do Not Process

For example, if you configure View as the top scenario, a visitor who views a product detail page sees recommendations for products that other customers also viewed.

[6]

1. Create Einstein Email Recommendations Content
   Determine what content to include in the recommendation.
2. Create Einstein Email Recommendations Logic
   Use Einstein Email Recommendations in to create custom recommendations logic or choose from the list.

[5] Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.
[6] Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

3. Configure the Einstein Email Recommendations Layout
   Determine the layout for how recommendations appear.
4. Configure an Einstein Email Recommendations Scenario
   Choose the recommendation scenarios to use in Einstein Email Recommendations.
5. Insert Einstein Email Recommendations Code
   Copy and insert Einstein Email Recommendations code into your email.
6. Email Recommendation Hash
   The hash is the unique ID that ties logic to a display assigned when generating an email recommendation. It is used in the call and in reporting. Email hashes are set up using an intuitive user interface which produces an individual hash for an email template. The email recommendation is made of two sections within the app, the logic, and the display. The Displays can be reused across multiple logic models. Each logic can only include one display, meaning that all email recommendations within an email hash must have the same HTML display.

Create Einstein Email Recommendations Content

Determine what content to include in the recommendation.

1. To create a custom display, click **Create New**. To choose a standard display, hover over the image and click **Select** or **Edit** and go to step 3.
2. For a custom display, enter a name and click **Create**.
3. **Create**.
4. Select an identifier to add to the display and click **Add**. To include more identifiers, repeat this step.
   Code for your selections appears.
5. Click **Preview**.
   A visual preview appears.
6. To save your display without publishing, select **Stage Changes**. Or click **Save and Publish**.

   Note: Publish the content before using it in an email.

7. Click **Exit**.

[7]

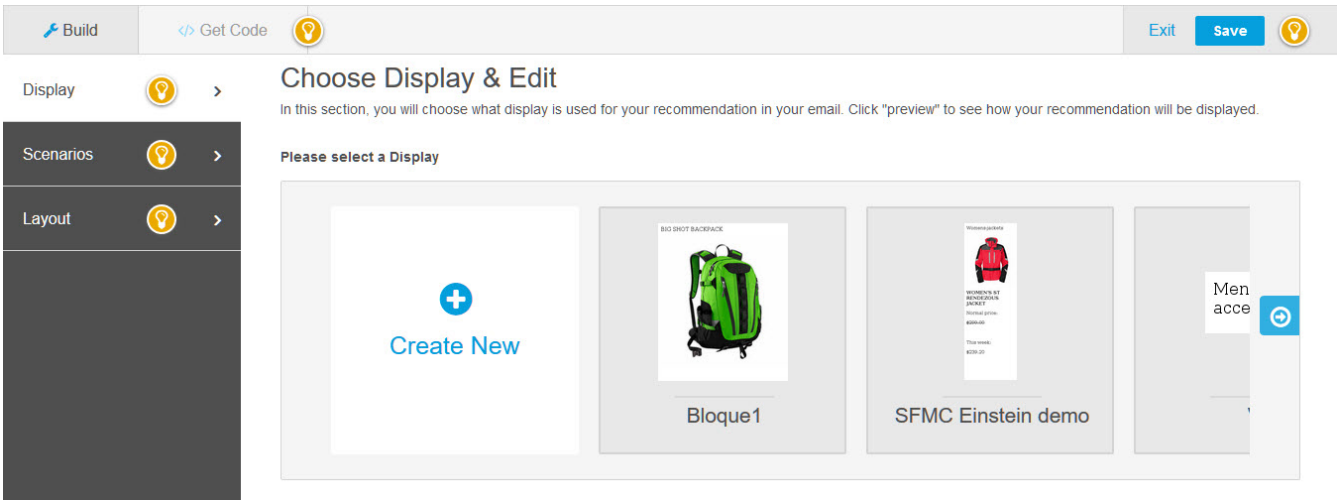
Create Einstein Email Recommendations Logic

Use Einstein Email Recommendations in to create custom recommendations logic or choose from the list.

1. Navigate to Einstein and click **Einstein Email Recommendations**.
2. To use standard logic, click the name.
3. Or, click **Create New Logic** .
   a. To create custom logic, enter a name.
   b. Select a catalog.
   c. Click **Create**.

[7] Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

The Display tab appears.



Configure the Einstein Email Recommendations Layout

Determine the layout for how recommendations appear.

1. Click the **Logic** tab and click the **Layout** tab.
2. To select the number of columns and rows for the recommendations content, hover over the grid, and click the last cell to include. For example, to render in the columns with two rows, click the cell labeled *3X2*.
3. Click **Preview**.
4. Edit your selection, if necessary.
5. Click **Save**.

[8]

Configure an Einstein Email Recommendations Scenario

Choose the recommendation scenarios to use in Einstein Email Recommendations.

1. Click the **Scenarios** tab.
2. Select a scenario from the list and click **Add**.

   Note: For details about what each scenario does, see Einstein Recommendation Scenarios.

   The scenario appears on the canvas. It automatically applies to all recommendation areas.
3. Click **all areas** and select a recommendation area from the dropdown list.
4. To add more scenarios, repeat steps 2 and 3.

   Tip: To remove a scenario from the canvas, click .

   Tip: Einstein Recommendations returns predictions from the top ranked scenario.

[8] Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

5. To preview the recommendations, click **Preview**.
6. To reorder the priority of scenarios, drag them into the order from top to bottom on the canvas.
7. To use waterfall predictions, select **Enable Waterfall Prediction**.

   Tip: Learn more about Waterfall Predictions.

8. To suppress system-generated scenarios, select **Disable System Scenarios**.
9. Click **Save**.

   Note: To restore the default scenario configuration, click **Restore to Default Best Practices** and click **Save**.

10. Click **Exit**.

[9]

Insert Einstein Email Recommendations Code

Copy and insert Einstein Email Recommendations code into your email.

Tip: To customize how recommendations render in an email, use AMPscript.

1. Click the **Logic** tab and click the recommendation to use.
2. Click the **Get Code** tab.
3. Note: Einstein Contact Code generates content in a preview only after you select a subscriber.

   To preview the recommendation, click **Preview**.
4. Copy the code.
5. Paste the code into the body of the email.

[10]

Email Recommendation Hash

The hash is the unique ID that ties logic to a display assigned when generating an email recommendation. It is used in the call and in reporting. Email hashes are set up using an intuitive user interface which produces an individual hash for an email template. The email recommendation is made of two sections within the app, the logic, and the display. The Displays can be reused across multiple logic models. Each logic can only include one display, meaning that all email recommendations within an email hash must have the same HTML display.

Set Up the Email Hash
Configure the email hash for Einstein Email Recommendations.

Set Up the Email Hash

Configure the email hash for Einstein Email Recommendations.

[9] Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.
[10] Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

1. Navigate to **Email Recommendations**.
2. Click **Create New Logic**.
3. Enter a unique name for the email template.
4. Select the **Type of Catalog**.
5. Click **Create**.

EINSTEIN WEB RECOMMENDATIONS

Einstein Web Recommendations use Einstein's capabilities to observe behavior, build preference profiles, and deliver the next-best personalized content to each website visitor. Use scenarios within the application to refine recommendations to match your specific business rules. Predictive Content is surfaced in placeholders for recommendations on your website.

Web Recommendations are delivered in two ways, via a JSON response or HTML/JS. The JSON response is the best practice delivery method unless their web team cannot work with JSON. The JSON method is the most flexible and allows their web team to style and utilize the recommendation data in any way. Because of the nature of the web recommendation and the JSON response, you are responsible for parsing and styling the recommendation within their web environment.

Here is an example of the response for a web recommendation:


```
[
    {
    "name": "product_rec",
    "title": "People who bought MEN'S BASE CAMP SLIP-ON III also bought",
    "priority": 1,
    "items": [
        {
            "image_link": "https://s3-us-west-2.amazonaws.com/nto-products/products370/1816100.jpg",
            "link": "https://stack7.collect.igodigital.com/redirect/v3QkFoSklnRytleUp1SWpvaVFtOT",
            "sale_price": 120,
            "regular_price": 150
        },
        {
            "image_link": "https://s3-us-west-2.amazonaws.com/nto-products/products370/1803100.jpg",
            "link": "https://stack7.collect.igodigital.com/redirect/v3QkFoSklnRytleUp1SWpvaVFtOTFaM",
            "sale_price": 88,
            "regular_price": 110
        },
        {
            "image_link": "https://s3-us-west-2.amazonaws.com/nto-products/products370/1848100.jpg",
            "link": "https://stack7.collect.igodigital.com/redirect/v3QkFoSklnRytleUp1SWpvaVFtOTFaM",
            "sale_price": 44,
            "regular_price": 55
        },
        {
            "image_link": "https://s3-us-west-2.amazonaws.com/nto-products/products370/1115100.jpg",
            "link": "https://stack7.collect.igodigital.com/redirect/v3QkFoSklnRzlleUp1SWpvaVFtOTFaMmg"
            ,
            "sale_price": 239.2,
            "regular_price": 299
        }]
    }
]
```


You can pull any catalog field into the web recommendation call, so it is vital to understand what data generates the recommendation during the catalog setup.

A unique web recommendation call is created for each type of page. We recommend that you include Home, Product, Category, Cart, and Conversion pages as a best practice. However, there is no limit to the number of pages that can be set up in the UI. Einstein Recommendations uses page templates, so a Product Page is a template used to build personalized recommendations for a product page.

Each type of page template can have different scenarios and a different context. For example, your product page can include recommendations based on the product a visitor is viewing based on that added context. However, your homepage recommendations are based on the overall user affinity because we don't have more context.

If you have a service engagement, the State of Work sets the number of pages services configures. However, you can add as many as you need.

11

[11] Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

Create an Einstein Web Recommendation
Use Marketing Cloud Einstein Recommendations to create product or content recommendations to display on your website. The Einstein Recommendation Engine requires at least three active items in your product catalog.

Create an Einstein Web Recommendation

Use Marketing Cloud Einstein Recommendations to create product or content recommendations to display on your website. The Einstein Recommendation Engine requires at least three active items in your product catalog.

Note: When your customer profile privacy field is set to one of the following, the system generates generic recommendations from the scenarios you configured.

- Do Not Track
- Do Not Profile
- Do Not Process

For example, you configure View as the top scenario. The customer who views a product detail page sees recommendations for products that other customers also viewed.

Tip: System Scenarios are enabled by default and provide filler content when configured scenarios do not produce the expected number of recommendations. To disable this feature, select **Disable System Scenarios** box.

[12]

1. Set the Location and Type of Web Recommendation
   Configure where Einstein Web Recommendations appear on your site and the type of recommendations to provide.
2. Create the Einstein Web Recommendations Content Area
   Select and configure the content to include in your web recommendations.
3. Configure the Web Recommendations Display Format
   Format how your recommendations appear on your website.
4. Configure the Einstein Web Recommendations Scenario
   Determine which recommendation scenarios to use in Einstein Web Recommendations.
5. Preview and Insert Einstein Web Recommendations
   Include the Einstein Web Recommendations code in the selected page's code. Both the JavaScript for the recommendation call and the HTML recommendation zone placeholder are provided.
6. Get Code Procedures
   Learn how to add the Einstein Web Recommendations code you create in Einstein Recommendations to your website.

Set the Location and Type of Web Recommendation

Configure where Einstein Web Recommendations appear on your site and the type of recommendations to provide.

1. Navigate to Einstein and select **Einstein Web Recommendations**.
2. To create a page, click **Create New Page** and enter a page name. To use an existing page, select a page from the list.

[12] Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

Note: The page name can't include spaces. Standard page names are Home, Cart, Product, Search, and Category. If you attempt to create a page using the same name as an existing page, the new page is not created.

3. Choose the recommendation type from the list.
4. Click **Create**.

The Build tab opens with a recommendation area with the default name *igdrec_1*.

Create the Einstein Web Recommendations Content Area

Select and configure the content to include in your web recommendations.

1. Click the name **igdrec_1** and enter a new name.
2. Choose **Number** or **Range** from the dropdown.
3. Enter the number, or select the range, of recommendations to return.

   To preview your recommendation, click **Preview**.

   Tip: If there are not enough items available for recommendations, nothing is returned unless you select **Waterfall Predictions**.
4. To add more recommendation areas to this page, click **Add Another Area** and repeat steps 1–4.
5. Click **Save**.

The Build tab closes and returns you to the web personalization dashboard. The new page is listed.

Configure the Web Recommendations Display Format

Format how your recommendations appear on your website.

1. Select the page to be modified and click the **Output** tab.

   Note: The canvas displays previously chosen output fields for the page. If nothing is chosen, the system generates recommended output fields based on your catalog fields.
2. Choose the output preview format.
3. Select a field to display from the list and click **Add**.
4. To add more fields, repeat the previous step.
5. To change the display fields order, drag the fields into order from top to bottom on the canvas.

   Note: To delete a field, click .

   The Output Preview displays the output code.
6. To see how the code renders, click **Preview**.

   To return to the canvas, click **Close**.
7. Click **Save**.

Configure the Einstein Web Recommendations Scenario

Determine which recommendation scenarios to use in Einstein Web Recommendations.

1. Click the recommendation to configure.
2. Click the **Scenarios** tab.
3. Select a scenario and click **Add**.

   Note: For details about what each scenario does, see Einstein Recommendation Scenarios.

   The scenario appears on the canvas. It is automatically set to apply to all recommendation areas.
4. Click **all areas** and select a specific recommendation area.
5. To add more scenarios, repeat steps 3 and 4.

   Tip: To remove a scenario from the canvas, click .

   Tip: Personalization Builder returns predictions from the top ranked scenario.
6. To reorder the priority of scenarios, drag them into the order from top to bottom.
7. To use waterfall predictions, select **Enable Waterfall Prediction**.

   Tip: Learn more about Waterfall Predictions.
8. To suppress system-generated scenarios, select **Disable System Scenarios**.
9. Click **Save**.

   Note: To restore the default scenario configuration, click **Restore to Default Best Practices** and click **Save**.

[13]

Preview and Insert Einstein Web Recommendations

Include the Einstein Web Recommendations code in the selected page's code. Both the JavaScript for the recommendation call and the HTML recommendation zone placeholder are provided.

1. To insert the code, click the **Get Code** tab.
2. Copy the code.

   For more information and to learn how to customize the code, review Get Code Details and Customize the Calls.
3. Place the code into your website code.

[14]

Get Code Procedures

Learn how to add the Einstein Web Recommendations code you create in Einstein Recommendations to your website.

Note: Basic HTML and CSS skills are required to complete this process.

1. Select a page from the Web Recommendation dashboard.

[13] Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.
[14] Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

2. Click the **Get Code** tab.
3. Copy each piece of code and embed it in the page code .

   For more information on embedding the code and customizing calls, review Get Code Details in Personalization Builder and Customize the Calls.

Tip: After the code is applied to your webpages, the recommendations are live. We recommend that you test recommendations to a test environment before you add them to a live site.

15

Get Code Details in Einstein Recommendations

Find the JavaScript and HTML code required to render Einstein Recommendations on your page. Copy the code and embed it in your page source for each piece of code.

Get Code Details in Einstein Recommendations

Find the JavaScript and HTML code required to render Einstein Recommendations on your page. Copy the code and embed it in your page source for each piece of code.

Customize the provided basic recommendation calls to match your site. The description key in each box shows additional information that you can pass in the JavaScript source URL. Replace these client parameters in the base URL of the call with valid values.

- <client>
- <pagename>

These client parameters are not run-time variables. The variables after *?* are supplied in the call at run time.

There are required fields for some pages.

Page	Required Field
Product Detail	ProductCode
Category	CategoryName
Cart	List of cart ProductCodes
Search	Search Term

Example: **Homepage**

Place the div where it's displayed in the recommendation. Place this value exactly where it appears on the page.

```
<div id="igdrec_1"></div>.
```

Next, place a script tag with this code.

```
<script
src="http://<client_name>.recs.igodigital.com/a/v2/<client>/home/recommend.js"></script>
```

Example: **Product Detail Page**

Place the div where it's displayed in the recommendation. Place this value exactly where it appears on the page.

15 Rights of ALBERT EINSTEIN are used with permission of The Hebrew University of Jerusalem. Represented exclusively by Greenlight.

`<div id=”igdrec_1”></div>`.

Next, place a script tag with this code.

```
<script
src="http://<client_name>.recs.igodigital.com/a/v2/<client>/product/recommend.js?item=ProductCode"></script>..
```

ProductCode is the item viewed and must match the same value passed in the Collect Tracking Code `item` variable.

Example: **Category and Sub-Category Page**

Place the div where the recommendation displays. Place this value exactly where it appears on the page.

`<div id=”igdrec_1”></div>`.

Next, place a script tag with this code.

`<script src="http://<client_name>.recs.igodigital.com/a/v2/<client>/category/recommend.js?category=Category_name"></script>`.

Replace Category with the actual category page name where the recommendation call is being made. This code must exactly match the categories stored in the product catalog category.

If the category name includes any spaces or special characters, the URL value must be encoded. For example, if the category is *Category & Name*, then the URL parameter is `category=Category%20%26%20Name`.

Example: **Shopping Cart Review Page**

Place the div where the recommendation displays. Place this value exactly where it appears on the page.

`<div id=”igdrec_1”></div>`.

Place a script tag with this code after the div.

`<script src="https://<client_name>.recs.igodigital.com/a/<client>/cart/recommend.js?cart=ProductCode".</script>`.

Match the value in ProductCode to the value sent in the Collect Tracking Code `cart` variable as a pipe-delimited list.

USE CASES

Review tips for common Einstein Recommendations use cases.

Add Multiple Recommendations Areas to a Page with Different Scenarios.

You can add two recommendation areas on a product page, for example. We recommend including one area of recommendations next to the product being viewed that focuses on similar products or upsells. This area needs four recommendations. You can add a second block of recommendations under the product details with recently viewed products. This area can have between two and eight recommendations, depending on how active the visitor has been on the site.

Show Recommendations Based on a Specific Language or Locale.

Einstein Recommendations provides multilingual recommendations when data is correctly specified within the content extract feed and the recommendation calls. Review Localized Recommendations for details.