Ultimate Guide to make

Prepared by: AI Marketing Automation Lab

Make (formerly Integromat) empowers marketers, operators, and business owners to connect their apps, move data seamlessly, and automate the repetitive grind—*without writing a single line of code*. What once took hours, now happens in seconds. What once required a developer, now takes only a few clicks.

This guide is your deep-dive into Make—from the fundamentals of scenarios and modules to advanced blueprints, best practices, and AI-powered workflows. You'll learn how to map, build, and scale powerful automations that free up your team, eliminate errors, and unlock serious growth.

Whether you're brand new to AI automation or already building scenarios, this resource is your go-to reference for everything Make. Ready to reclaim your time and turbocharge your marketing?

Let's get started.

Core Concepts: How Make Works

To truly master Make, you need to understand its foundational building blocks. Everything you automate on the platform is built from these core elements—combine them right, and you can automate nearly any digital workflow, no matter how complex.

Scenarios: The Engine of Automation

A scenario in Make is your complete workflow. It's a visual flowchart connecting multiple apps and services, moving data step by step. **Every scenario starts with a trigger** (the event that kicks things off) and follows with one or more modules (the actions you want to happen).

Example:
"When a new lead fills out a form on my website, add them to HubSpot, send them a welcome email, and ping my team on Slack."
That's a single scenario—one trigger, several modules, all working together automatically.

Modules: Your Building Blocks

A module is a single "action" or "step" inside a scenario. Each module can connect to an app (like Gmail, Salesforce, or Google Sheets), transform data, make decisions (filters, routers), or even interact with APIs.

- **App Modules:** Do things like "create a new contact in HubSpot" or "send an email via Outlook."
- **Utility Modules:** Include tools like filters ("if/else" logic), aggregators (combine items), iterators (loop through data), and formatters (change dates, numbers, text).
- **Webhooks & HTTP:** For connecting to web services or receiving real-time data from forms, shops, and other sources.

Modules are the pieces of your automation puzzle—string them together, and **you can move data anywhere.**

Triggers vs. Actions

- **Trigger:** The event that starts your scenario. Could be an incoming webhook, a new row in Google Sheets, a received email, etc.

- **Action:** The steps that follow your trigger—send, update, create, or manipulate data across your connected tools.

You can only have **one trigger per scenario**, but as many actions (modules) as you want. Some actions can even trigger "sub-scenarios" using Make's advanced features.

Data Mapping: Moving Information Between Apps

One of Make's superpowers is visual data mapping. When you add a module, you'll see fields (like "first name" or "email") you can populate with outputs from previous modules. **Just drag and drop—no code required.** Make automatically keeps track of what data is available at each step.

Advanced Tip: You can use functions and expressions (like formatDate(now; "YYYY-MM-DD") or toUpperCase({{1.name}})) to transform data as you move it through your scenario.

Blueprints: Share, Clone, and Collaborate

Every scenario you create in Make can be exported as a blueprint—a portable JSON file containing the structure and logic of your automation (not the actual data). Blueprints are ideal for backing up, sharing with your team, or replicating successful automations across multiple workspaces.

Connections: Linking Your Tools Securely

Each app you use in Make requires a connection—a secure handshake using OAuth, API keys, or login credentials. Connections are managed in your Make dashboard and can be reused across scenarios.
Pro tip: Always use dedicated integration users and keep your API keys private for maximum security.

CRM Integrations: HubSpot & Salesforce

A robust **CRM integration is the backbone of modern marketing automation.** Make connects seamlessly to top CRM platforms like HubSpot and Salesforce, empowering you to synchronize leads, automate follow-ups, and keep your sales pipeline perfectly aligned—all without manual data entry.

HubSpot CRM Modules

What You Can Do:
With HubSpot modules, you can automatically create, search, update, or delete contacts, companies, deals, and more. Use these tools to manage your sales funnel, assign new leads, enroll users in campaigns, and keep your marketing database clean and current.

Setup Details:
To connect HubSpot, you'll use OAuth. Once authenticated, all HubSpot modules in your scenarios will leverage this connection.

Key Module Types:

- **Create a Record:** Instantly add new contacts, companies, or deals. Perfect for webform-to-CRM automations.
- **Update a Record:** Change details for any existing record—like updating a lead's status after a sales call.

- **Search for Records:** Find contacts or companies using filters such as email or company name.
- **Watch Records:** Trigger a scenario whenever new contacts, companies, or deals are created or updated.
- **Delete a Record:** Remove contacts or other objects when needed for data hygiene.
- **Field Mapping:** Each module allows you to match fields from previous steps in your scenario to HubSpot's required (and optional) properties—things like first name, email, company, or custom fields.

Sample JSON Blueprint: Add a New Contact to HubSpot

```json
{
  "id": 3,
  "module": "hubspot-crm:createContact",
  "parameters": {
    "__IMTCONN__": 987654
  },
  "mapper": {
    "properties": {
      "email": "{{1.email}}",
      "firstname": "{{1.first_name}}",
      "lastname": "{{1.last_name}}"
    }
  }
}
```

Best Practices:

- When using “Watch” modules, **be aware that they may trigger for every change**—not just specific fields. Use filters for more precision.
- HubSpot’s API has rate limits; for bulk operations, batch requests and monitor usage.
- Need advanced logic? Use the “Make an API Call” module to interact with custom endpoints.

Salesforce CRM Modules

Salesforce modules enable you to create, update, search, and delete objects like Leads, Contacts, Accounts, and Opportunities. Automate onboarding, sync data from e-commerce or support channels, and **keep Salesforce as your single source of truth.**

Setup Details:
Authenticate using Salesforce OAuth (with the right permissions for all objects you plan to manage).

Key Module Types:

- **Create a Record:** Add new leads, contacts, accounts, or custom objects to your CRM.
- **Update a Record:** Edit existing records by their Salesforce ID.
- **Search for Records:** Look up objects by criteria—like “find a Contact by email.”
- **Watch Records:** Trigger a scenario whenever a new record is added or an existing one is updated.
- **Delete a Record:** Remove old or duplicate entries.
- **Bulk Modules:** Leverage Salesforce’s bulk APIs for high-volume operations.
- **Field Mapping:** Map scenario data to Salesforce object fields, ensuring you use the correct types (text, picklist, date, etc.).

Sample JSON Blueprint: Add a New Contact to Salesforce

```JSON
{
  "id": 4,
  "module": "salesforce:createRecord",
  "parameters": { "__IMTCONN__": 112233 },
  "mapper": {
    "sObject": "Contact",
    "fields": {
      "FirstName": "{{1.first_name}}",
      "LastName": "{{1.last_name}}",
      "Email": "{{1.email}}"
    }
  }
}
```

Best Practices:

- Always verify field requirements in Salesforce (some fields are mandatory or have restricted values).
- To avoid duplicate data, search for an existing record before creating or use upsert logic.
- For large imports/updates, use the bulk job modules for efficiency.
- Keep error-handling in place to catch issues like invalid field types or API limits.

Email Integrations: Gmail & Outlook

Automating your email workflows is one of the quickest wins for any marketing or operations team. With Make, **you can streamline communication, manage inbound leads, automate replies, and keep your inbox organized**—whether you use Gmail or Microsoft 365/Outlook.

Gmail Modules

Capabilities:
Connect your Gmail account to Make and you'll be able to send, receive, monitor, and organize emails automatically. Use Gmail modules to build flows that acknowledge webform submissions, route sales inquiries, log customer emails, and more.

Authentication:
Set up your Gmail connection via OAuth. You'll have access to your inbox, sent mail, drafts, and all custom labels.

Key Module Types:

- **Watch Emails (Trigger):** Monitor a mailbox or label for new messages. Filter by sender, subject, or even message content.
- **Send an Email:** Automate outbound messages—customize To, CC, BCC, subject, and the message body (plain or HTML). You can add file attachments and personalize content with scenario data.
- **Create a Draft:** Build email drafts for review before sending.
- **Get an Email:** Retrieve details and content from any message by its unique ID.
- **Delete or Move Email:** Automatically clean up your mailbox or sort messages after processing.
- **Watch Drafts:** Use this trigger to launch scenarios when new drafts are created (less common, but available).

Example JSON Blueprint: Send Email via Gmail

```json
{
  "id": 5,
  "module": "email:ActionSendEmail",
  "version": 7,
  "parameters": {
    "folder": "[Gmail]/Sent Mail",
    "account": 755539,
    "accountImap": 755539,
    "saveAfterSent": true
  },
  "mapper": {
    "to": ["jane@example.com"],
    "cc": [],
    "bcc": [],
    "from": "",
    "subject": "Welcome to Our Service",
    "html": "<p>Hello {{name}},</p><p>Thank you for joining...</p>",
    "attachments": [],
    "replyTo": "",
    "inReplyTo": "",
    "headers": []
  }
}
```

Tips & Best Practices:

- To keep replies threaded, use the In-Reply-To header (especially for support or multi-step flows).
- Be mindful of Google's sending and API rate limits—avoid automating large blasts through personal accounts.
- For personalized outreach, map scenario variables to the email body and subject.
- Attachments must comply with Gmail's size restrictions (usually 25–35 MB total).

Microsoft 365/Outlook Modules

Outlook integration lets you automate every step of your email process—send confirmations, route inbound requests, sort and categorize emails, and more.

Authentication:

Connect using your Microsoft (Office 365) credentials through OAuth.

Key Module Types:

- **Watch Emails:** Keep an eye on an inbox or any folder for new messages, using filters for fine-tuned control.
- **Send an Email:** Deliver messages with custom subject, recipients, and body—plus attachments and category tags.
- **Create a Draft:** Prepare emails for review or manual sending.
- **Move/Organize Email:** Sort emails into folders, or categorize them automatically based on scenario logic.
- **Reply to Email:** Send threaded responses to keep conversations in context.
- **Get/Delete Email:** Retrieve or remove messages by their unique ID.

Example JSON Blueprint: Send Email via Outlook

```JSON
{
  "id": 6,
  "module": "microsoft-email:sendEmail",
  "parameters": { "__IMTCONN__": 445566 },
  "mapper": {
    "to": ["jane@example.com"],
    "subject": "Your Order Update",
    "body": "Hello, your order has shipped.",
    "contentType": "text",
    "attachments": []
  }
}
```

Tips & Best Practices:

- Rate limits and sending quotas apply; throttle batch sends for large campaigns.
- Use categories and folders for clean, automated inbox management.
- Shared mailbox support is available—set the correct permissions on your Microsoft account.
- Attachments are limited by Microsoft's file size constraints (generally 25 MB per email).
- Always test flows with dummy accounts or sandboxes before pushing to live customer lists.

Databases & Spreadsheets: Google Sheets, Airtable, MySQL

Managing data is central to every marketing and business workflow. Make's spreadsheet and **database modules let you automate updates, sync records across platforms, and keep all your critical data perfectly aligned**—no manual entry, no more copy-paste errors.

Google Sheets Modules

Connect Google Sheets to Make and automate every step: add new rows from form submissions, update records after customer actions, search for values, and generate custom reports—all without opening a spreadsheet manually.

Authentication:
Authenticate using your Google account via OAuth.

Key Module Types:

- **Watch Rows (Trigger):** Launch a scenario whenever a new row appears in a chosen sheet or tab.
- **Search Rows:** Find specific rows matching your filter (by value, range, or unique key).
- **Add Row:** Append a row of data—perfect for logging leads, orders, or survey responses.
- **Update Row:** Change cell values in a specific row (by index or unique key).
- **Get Cell/Range**: Fetch values from a cell or block of cells.
- **Clear Range:** Remove contents from selected cells or columns.
- **Make an API Call:** Advanced control for complex/batch actions beyond the standard modules.

Example JSON Blueprint: Add Row to Google Sheet

```JSON
{
 "id": 9,
 "module": "google-sheets:addRow",
 "parameters": {
  "__IMTCONN__": 667788,
  "__IMTHOOK__": 99251
 },
 "mapper": {
  "spreadsheetId": "1AbCdEfGhIjKlMNopQRsTUVwxyz12345",
  "sheetName": "Form Responses",
  "values": [
   "{{formSubmission.name}}",
   "{{formSubmission.email}}",
   "{{formSubmission.message}}",
   "{{formatDate(now; 'YYYY-MM-DD HH:mm:ss')}}"
  ]
 }
}
```

Tips & Best Practices:

- Use unique IDs in your sheet to easily search and update rows.
- Large sheets can run into Google's API limits; use batch processing where possible.
- Don't overwrite formulas—map scenario data into blank columns or use ARRAYFORMULA.
- For complex reports or dashboards, automate data pulls at set intervals (hourly, daily, etc.).

Airtable Modules

Airtable is a powerful hybrid between a spreadsheet and a database. Make's modules let you automatically create, update, search, and organize records across your bases and tables.

Authentication:
Connect with your Airtable API key or personal access token.

Key Module Types:

- **Watch Records (Trigger):** Detect new or updated records in a view or table.
- **Search Records:** Locate matching records with field filters or formula logic.
- **Create Record:** Add a new row to any Airtable table—map every field from your scenario data.
- **Update Record:** Change data in a record using its unique Airtable ID.
- **Delete Record:** Remove unwanted or outdated records.
- **List Records:** Fetch multiple records for summaries, reporting, or iterating.
- **Make an API Call:** For advanced, custom, or batch operations.

Example JSON Blueprint: Create Record in Airtable

```JSON
{
 "id": 8,
 "module": "airtable:createRecord",
 "parameters": { "__IMTCONN__": 778899 },
 "mapper": {
  "baseId": "appABC123456",
  "table": "Projects",
  "record": {
   "Name": "New Website",
   "Status": "Planned",
   "Start Date": "2025-05-01"
  }
 }
}
```

Tips & Best Practices:

- Mind Airtable's API rate limits (5 requests/sec).
- Keep field types consistent—dates, numbers, and selects should match your Airtable schema.
- Store record IDs in your scenario for smooth future updates.
- Batch processes where possible for efficiency.

MySQL Modules

If your data lives in a classic MySQL database, **Make lets you connect and automate** inserts, updates, queries, and more. It's perfect for integrating legacy systems, ecommerce data, or internal apps.

Authentication:
Set up a secure connection with your MySQL host, database name, username, and password (and make sure Make's IPs are allowed on your server).

Key Module Types:

- **Execute a Query (Advanced):** Run any valid SQL—SELECT, INSERT, UPDATE, DELETE, even complex joins.
- **Insert Row(s):** Add new data directly via no-code field mapping.
- **Update Row(s):** Change values in specific records based on your filter logic.
- **Select Row(s):** Fetch matching records for reports, automations, or processing.
- **Delete Row(s):** Remove records based on defined conditions.
- **Execute Stored Procedure:** Call a stored procedure, passing in any required arguments.

Example JSON Blueprint: Insert Row in MySQL

```JSON
{
 "id": 7,
 "module": "mysql:insert",
 "parameters": { "__IMTCONN__": 223344 },
 "mapper": {
  "table": "users",
  "values": {
   "name": "Jane Doe",
   "email": "jane@example.com",
   "created_at": "{{formatDate(now; 'YYYY-MM-DD')}}"
  }
 }
}
```

Tips & Best Practices:

- To set SQL NULLs, use the advanced query module (empty strings in field mapping become “empty,” not true NULL).
- Secure your credentials and never expose API keys or passwords in logs.
- Limit and offset queries for bulk operations to avoid timeouts.
- Always backup your data and test with sample/dummy records.

Webhooks, HTTP Requests & API Integrations

No matter how many apps Make supports natively, you'll always have unique tools, in-house platforms, or edge-case services to connect. That's where webhooks and HTTP modules come in: they make Make an open automation hub, able to talk to virtually any service on the web.

Webhooks: Real-Time Triggers and Responses

Purpose:
A webhook is a special URL that can receive data from any service that knows how to "call" it. Use webhooks to trigger Make scenarios instantly—when a customer fills out a form, when your payment processor sends a receipt, or when an IoT device updates its status.

Setup:
Add a Custom Webhook module as the first step in your scenario. Make will generate a unique URL—paste that into the "webhook"/notification field in your external app or service.

- **Webhook (Trigger):** Listens for inbound HTTP requests and starts your scenario when data arrives.
- **Webhook Response:** Lets your scenario send an immediate reply—status code, headers, and body—back to the sender. This is great for APIs that expect a confirmation.

Example JSON: Custom Webhook & Response

```
JSON
{
  "id": 1,
  "module": "gateway:CustomWebHook",
  "parameters": { "hook": 721369, "maxResults": 1 },
  "mapper": {}
},
{
  "id": 2,
  "module": "gateway:WebhookRespond",
  "parameters": {},
  "mapper": {
    "statusCode": 200,
    "body": "{ \"success\": true }",
    "headers": { "Content-Type": "application/json" }
  }
}
```

Best Practices:

- If you expect a lot of traffic, use the maxResults parameter to batch incoming webhook calls.
- Many services require you to validate incoming data—use tokens, check IP addresses, or validate signatures for security.
- Only one Webhook Response module is allowed per scenario; plan your branching accordingly.
- Some services will time out if your scenario takes too long—send an early response, then process heavy logic afterward.

HTTP Module: Call Any REST API

The HTTP module is Make's Swiss army knife: it lets you call any URL, submit any HTTP method (GET, POST, PUT, PATCH, DELETE), and pass custom headers, query parameters, and body data. This means **you can integrate with *any* modern API,** even if it's not in Make's module library.

Setup:

- Specify the URL and method.
- Add headers for authentication (API keys, OAuth tokens, etc.) and content type.
- Fill out query parameters and request body as needed.
- Parse and use the response (JSON, XML, text, or even files) downstream in your scenario.

Example JSON: POST to a REST API

```
JSON
{
 "id": 10,
 "module": "http:ActionSend",
 "parameters": {},
 "mapper": {
  "url": "https://api.example.com/v1/users",
  "method": "post",
  "headers": {
   "Authorization": "Bearer {{token}}",
   "Content-Type": "application/json"
  },
  "body": "{ \"name\": \"{{username}}\", \"email\": \"{{useremail}}\" }",
  "allowIncomplete": false
 }
}
```

Best Practices:

- Always check your API's documentation for required fields, authentication methods, and expected responses.
- For repeated, high-volume calls, add throttling or delay steps to avoid hitting rate limits.
- Handle errors gracefully: check the response code and route errors to separate branches or logs.
- Use parsing functions to extract the data you need from complex JSON or XML responses.
- Keep your API keys secure; don't include them in logs or error messages.

Connecting In-House or Unsupported Apps

Many organizations use custom-built platforms, legacy databases, or niche SaaS tools. As long as your app can send an HTTP request or receive one, you can automate it with Make. Use outgoing webhooks to notify Make, and the HTTP module to send or fetch data from anywhere.

Typical Scenarios:

- Receive real-time order notifications from e-commerce tools.
- Push CRM updates to a proprietary internal dashboard.
- Fetch data from analytics APIs and aggregate reports in Google Sheets or Airtable.

AI & LLM Integrations: OpenAI, Claude, Gemini, and More

Artificial intelligence is revolutionizing how we work, and Make gives you direct access to the most advanced AI models—right inside your automations. From content generation and summarization to classification, image creation, and audio transcription, you can layer AI smarts into any scenario, no code required.

OpenAI (ChatGPT, Whisper, DALL-E)

With OpenAI modules, you can automate everything from generating personalized content (using GPT-3 or GPT-4), to transcribing audio (with Whisper), to creating images on the fly (via DALL-E).

Authentication:
Connect using your OpenAI API key, which you'll manage within your Make connections.

Key Module Types:

- **Create a Completion (Prompt):** Send custom prompts to ChatGPT and return the model's response for use anywhere downstream (emails, Slack, Airtable, etc.).
- **Create Moderation:** Check text for compliance with OpenAI's safety policies—ideal for user-generated content.
- **Transcribe Audio (Whisper):** Upload audio files and instantly receive a transcript.
- **Translate Audio (Whisper):** Convert audio in other languages into English text.
- **Generate Image (DALL-E):** Create on-demand visuals from natural language prompts.
- **Make an API Call:** For advanced features and full control, interact directly with OpenAI's endpoints.

Example JSON: Summarize Text with GPT

```JSON
{
 "id": 11,
 "module": "openai:CreateCompletion",
 "parameters": { "__IMTCONN__": 334455 },
 "mapper": {
  "model": "gpt-4-turbo",
  "prompt": "Summarize the following in one sentence: {{emailBody}}",
  "maxTokens": 50,
  "temperature": 0.7
 }
}
```

Best Practices:

- Watch your API token and request limits; usage can grow quickly in heavy automations.
- For reliable outputs, **be as clear and specific as possible in your prompts.** If you want structured data, ask for a specific format (e.g., “Return the answer in JSON”).
- Moderate content automatically to prevent inappropriate or off-brand results.
- When using Whisper or DALL-E, ensure your files meet format/size requirements for the API.

Anthropic Claude, Google Gemini, and Other LLMs

The AI world is moving fast—**Anthropic's Claude, Google's Gemini, and other LLMs offer similar abilities to OpenAI**, with their own strengths (compliance, search integration, or image understanding). Make's HTTP and webhook modules let you connect to any of these platforms using their API.

Typical Scenarios:

- Summarize incoming messages, docs, or support requests.
- Classify customer feedback or route tickets by sentiment and topic.
- Generate **campaign copy, blog outlines, or social content** automatically.
- Pull real-time answers from a search-augmented LLM (like Perplexity).

Best Practices:

- Always read your AI vendor's API documentation—setup, security, and formatting can differ from OpenAI's.
- Secure your API keys, especially for paid or high-privilege accounts.
- Build error handling for API timeouts or invalid responses—route failures to Slack, Sheets, or email so nothing is lost.

Vision & Data Science APIs

- **Vision APIs (Google, AWS, Azure):** Analyze images, detect objects or text, and process receipts or invoices as part of your automation flow.
- **Custom Models:** Connect to any REST-accessible machine learning or analytics service for predictions, classification, or scoring.
- **Bringing It All Together:** Multi-Step AI Automation

The real power comes when you combine AI modules with your regular Make automations:

- Transcribe a support voicemail (Whisper), summarize and tag it (GPT/Claude), create a ticket (Airtable), and route a Slack alert—all without human intervention.
- Have inbound leads classified by industry or readiness using AI, and trigger tailored follow-up automations.

Advanced Scenarios, Blueprints & Best Practices

Make isn't just about automating simple tasks; it's powerful enough to orchestrate entire business processes. The magic comes when you layer modules, conditional logic, and even AI to create sophisticated, multi-step workflows. Here's how to take your automations to the next level.

Blueprint 1: Seamless Lead Capture and Instant Team Notification

Workflow:
When a new lead submits your website form, Make instantly pushes their info into your CRM (HubSpot, Salesforce, or Pipedrive), then pings your sales team on Slack—so nobody ever misses a hot opportunity.

Steps:

1. **Trigger:** Custom Webhook receives data from your website form.
2. **Action:** Create a new contact in your CRM, mapping all form fields to CRM fields.
3. **Action:** Send a detailed Slack message to the sales team's channel.

Example JSON Blueprint:

```
JSON
[
 {
  "id": 1,
  "module": "gateway:CustomWebHook",
  "parameters": {"hook": 543210, "maxResults": 1},
  "mapper": {}
 },
 {
  "id": 2,
  "module": "hubspot-crm:createContact",
  "parameters": {"__IMTCONN__": 102938},
  "mapper": {
   "properties": {
    "email": "{{1.email}}",
    "firstname": "{{1.first_name}}",
    "lastname": "{{1.last_name}}",
    "company": "{{1.company}}"
   }
  }
 },
 {
  "id": 3,
  "module": "slack:sendMessage",
  "parameters": {"__IMTCONN__": 123321},
  "mapper": {
   "channel": "#new-leads",
   "text": "A new lead is in! Name: {{1.first_name}} {{1.last_name}} ({{1.email}}) |
Company: {{1.company}}"
  }
 }
]
```

Blueprint 2: Automated Order Processing and Inventory Sync

Workflow:
Every time an e-commerce order comes in, Make logs the order in your database, updates your inventory in Airtable, and sends the customer a branded confirmation email.

Steps:

1. **Trigger:** Shopify (or WooCommerce) “Watch Orders” module.
2. **Action:** Insert order details into your SQL database.
3. **Action:** Update the product inventory count in Airtable.
4. **Action:** Send a confirmation email through Gmail.

Example JSON Blueprint:

```JSON
[
 {
  "id": 1,
  "module": "shopify:WatchOrders",
  "parameters": {"__IMTCONN__": 225588},
  "mapper": {}
 },
 {
  "id": 2,
  "module": "mysql:insert",
  "parameters": {"__IMTCONN__": 776655},
  "mapper": {
   "table": "orders",
   "values": {
    "order_id": "{{1.id}}",
    "customer_email": "{{1.email}}",
    "total": "{{1.total_price}}",
```

```
      "date": "{{1.created_at}}"
     }
    }
  },
  {
    "id": 3,
    "module": "airtable:updateRecord",
    "parameters": {"__IMTCONN__": 889900},
    "mapper": {
     "baseId": "appDEF789012",
     "table": "Inventory",
     "recordId": "{{1.product_id}}",
     "record": {
      "Stock": "{{subtract({{3.Stock}}, {{1.quantity}})}}"
     }
    }
  },
  {
    "id": 4,
    "module": "gmail:sendEmail",
    "parameters": {"__IMTCONN__": 445566},
    "mapper": {
     "to": "{{1.email}}",
     "subject": "Your Order Confirmation",
     "html": "<p>Thank you for your order, {{1.name}}!</p><p>We'll notify you once it
ships.</p>"
    }
  }
]
```

Blueprint 3: AI-Powered Support Ticket Triage

Workflow:
A customer email lands in your support inbox, gets summarized and classified by GPT-4, routed to the right Slack channel, and logged in Airtable for reporting.

Steps:

1. **Trigger:** Gmail “Watch Emails” module listens for new messages.
2. **Action:** Send the email body to OpenAI’s GPT for a short summary and topic classification.
3. **Action:** Route a Slack notification to the “support” or “sales” channel based on topic.
4. **Action:** Create a new record in Airtable, logging the summary and assignment.

Example JSON Blueprint:

```JSON
[
 {
  "id": 1,
  "module": "gmail:watchEmails",
  "parameters": {"__IMTCONN__": 564738, "label": "Inbox"},
  "mapper": {}
 },
 {
  "id": 2,
  "module": "openai:CreateCompletion",
  "parameters": {"__IMTCONN__": 334455},
  "mapper": {
   "model": "gpt-4-turbo",
   "prompt": "Summarize and classify this email as 'Sales' or 'Support': {{1.body}}",
   "maxTokens": 60,
   "temperature": 0.3
  }
```

```
  },
  {
    "id": 3,
    "module": "slack:sendMessage",
    "parameters": {"__IMTCONN__": 443322},
    "mapper": {
      "channel": "{{if(equals({{2.choices[0].text}}, 'Sales'), '#sales', '#support')}}",
      "text": "New customer message: {{2.choices[0].text}}"
    }
  },
  {
    "id": 4,
    "module": "airtable:createRecord",
    "parameters": {"__IMTCONN__": 112244},
    "mapper": {
      "baseId": "appQWERTY09876",
      "table": "Customer Cases",
      "record": {
        "Email": "{{1.from}}",
        "Summary": "{{2.choices[0].text}}",
        "Assigned Channel": "{{3.channel}}"
      }
    }
  }
]
```

Best Practices for Scaling & Reliability

- **Error Handling:** Always use error routes—log errors to Sheets or Slack, and send critical failures to your team.
- **Testing:** Use Make's "Run once" and log inspection tools to check every step before enabling scenarios.
- **Version Control:** Export and save your blueprints often, especially before big changes.

- **Rate Limits:** Be aware of quotas for every service; batch or delay large jobs to stay compliant.
- **Secure Credentials:** Store API keys in Make's connection manager—not in your scenario code or logs.
- **Documentation:** Add scenario descriptions, label branches, and comment on your blueprints for easy maintenance.
- **Scheduling:** For routine jobs, use Make's scheduler to trigger scenarios at set times—great for reporting or batch data syncs.

Troubleshooting & Maintenance: Keeping Your Automations Resilient

No matter how carefully you build, even the best automations hit snags—API outages, data changes, quota limits, or simple human error. Make gives you robust tools to monitor, troubleshoot, and fix problems before they become showstoppers.

Scenario Logging & Monitoring

- **Scenario Logs:** Every run—successful or not—is recorded in Make's log. Check each step for details: inputs, outputs, errors, and processing time. Use these logs to pinpoint where something went wrong.

- **Execution History:** Drill down into past runs to see trends, patterns, or repeat errors. Helpful for debugging intermittent or timing-based issues.

- **Notifications:** Set up email, Slack, or Teams notifications for failures, timeouts, or specific error codes. You'll know the moment a scenario needs your attention.

Common Pitfalls & How to Fix Them

- **API Rate Limits:** If a service cuts you off, Make will log the failure. Use routers or delays to slow down batch jobs, and always check API documentation for limits.

- **Field Mismatches:** When data structures change—like a CRM field being renamed or removed—update your module mappings right away. Use test runs to catch these issues before going live.

- **Invalid Credentials:** If your OAuth token or API key expires, re-authenticate the connection in Make. For long-lived integrations, consider service accounts or app-specific passwords where supported.

- **Timeouts:** Some modules (especially API calls and database queries) may time out on slow responses. Optimize queries, add error handling, and avoid huge data pulls in a single run.

- **Uncaught Errors:** Always use error handlers—don't let failed steps break the whole process. Log errors, retry if needed, or alert a human for review.

Scenario Updates & Version Control

- **Export Blueprints:** Save your scenario blueprints (JSON) regularly, especially before making changes. Keep them organized for quick rollbacks if needed.

- **Change Management:** Document what each scenario does, who owns it, and when it was last updated. For teams, use a shared doc or project board for tracking.

- **Testing Changes:** Use Make's "Run once" mode and logs to test all paths—especially after API changes, platform updates, or big workflow tweaks.

Collaboration & Knowledge Sharing

- **Scenario Descriptions:** Add clear descriptions and comments for every scenario, router, or important step. Future you (or your teammates) will thank you.

- **Labeling & Organization:** Group scenarios by purpose or department (marketing, sales, ops) to keep your Make dashboard organized.

- **Community & Support:** Use the Make Community Forum for expert tips, troubleshooting, and sharing best practices with peers.

Routine Maintenance Checklist

- ☐ Review scenario logs weekly for errors or slowdowns.
- ☐ Update connections and credentials as needed.
- ☐ Export and back up blueprints after any major edit.
- ☐ Monitor API docs for breaking changes.
- ☐ Regularly test all critical paths with sample data.
- ☐ Communicate changes and learnings with your team.

Resources

Essential Links & Further Reading

- **Make Help Center:** https://www.make.com/en/help
  Official documentation, module guides, and how-to articles for every supported app and service.

- **Make Community Forum:** https://community.make.com
  Ask questions, find blueprints, get troubleshooting help, and connect with fellow automation enthusiasts.

- **Template Library:** https://www.make.com/en/templates
  Browse, clone, and adapt ready-made scenarios for marketing, sales, operations, AI, and more.

- **API Docs:** Every app or tool you connect will have its own API documentation—consult these for advanced or custom integrations.

The AI Marketing Automation Lab 

Automation is better together.

Join the AI Marketing Automation Lab Community and connect with other marketers who are building advanced systems, testing bold ideas, and sharing results in real time.

Why figure it out alone when you can learn, build, and grow with the best?