

# Corrigé

P1C4 - Lisez et exploitez le format JSON



## Corrigé — Activité JSON / mapping (Storra)

### 1. Structure globale

La réponse commence par '[' et se termine par ']' : c'est un tableau. Il contient 2 objets, correspondant à 2 commandes en attente.

### 2. Données utiles et chemins d'accès

Donnée	Chemin d'accès	Valeur (commande 1042)	Valeur (commande 1043)
Identifiant	id	"1042"	"1043"
Statut	status	"pending"	"pending"
Montant total	total	59.90	34.00
Nom du client	customer.name	"Amara Diallo"	"Léo Ferreira"
Email du client	customer.email	"amara.diallo@email.com"	null
Produit commandé	items[0].product	"Tote bag Storra"	"Carnet de notes"

À noter : items est un tableau imbriqué. Pour accéder au premier produit, on utilise items[0].product — l'index [0] désigne le premier élément du tableau.

### 3. La valeur null

Le champ `customer.email` de la commande 1043 est null : le client n'a pas renseigné son adresse email. Si votre workflow tente d'envoyer un email Mailchimp avec cette valeur, il échouera — ou s'exécutera sans rien envoyer.

Pour anticiper ce cas dans n8n, vous pouvez ajouter un nœud conditionnel (IF) avant le nœud Mailchimp : si `customer.email` est null, la commande est routée vers une branche alternative (par exemple une ligne dans Airtable marquée « email manquant ») pour traitement manuel.

### 4. Mapping vers Airtable

Champ JSON	Colonne Airtable
<code>id</code>	ID_commande
<code>status</code>	Statut
<code>total</code>	Montant_total
<code>customer.name</code>	Nom_client
<code>customer.email</code>	Email_client
<code>items[0].product</code>	Produit

Dans n8n, après le nœud HTTP Request, vous ajoutez un nœud Airtable en mode Create record et vous reliez chaque variable JSON au champ Airtable correspondant.