

# Community Place Prompts Collection

All AI Prompts Organized & Ready to Use

## Complete Prompts Library

AI • N8N • Notion • Ready to Copy-Paste

### Folder Structure

```
Prompts_Collection/  
├── AI_Prompts/  
│   ├── Receipt_Extraction_Prompt_Full.md  
│   ├── Library_Categorization_Prompt.md  
│   └── Document_Sorting_Prompt.md  
├── N8N_Prompts/  
│   ├── GPT4o_Integration.md  
│   └── Error_Handling_Examples.md  
└── Notion_Prompts/  
    ├── Database_Structure.md  
    └── Property_Formatting.md
```

### AI Prompts

#### 1. Receipt Extraction (GPT-4o Vision)

**File:** AI\_Prompts/Receipt\_Extraction\_Prompt\_Full.md

**Purpose:** Extract receipt data and categorize into Greg's 8 expense categories

**Model:** GPT-4o Vision

**Temperature:** 0.1

**Max Tokens:** 500

**Categories:**

- Application
- Website
- Cloud
- Housing
- Utilities
- Equipment
- Software
- Other

**Output Format:** JSON with vendor, date, amount, category, billingCycle, notes, confidence

## 2. Library Document Categorization

**File:** AI\_Prompts/Library\_Categorization\_Prompt.md

**Purpose:** Auto-sort uploaded documents into correct library categories

**Categories:**

- AI Guide
- API Guide
- Notion Guide
- N8N Guide
- Ebooks Library
- Resources Center

**Output:** Category name + confidence score

## N8N Integration Prompts

### GPT-4o in N8N Workflows

#### System Prompt Template:

```
You are [role description].
```

#### RULES:

- [Rule 1]
- [Rule 2]
- [Rule 3]

#### OUTPUT FORMAT:

```
{  
  "field1": "value",  
  "field2": 123  
}
```

IMPORTANT: Return ONLY JSON, no markdown.

#### User Message Format:

```
[Task description]  
[Context or data to process]
```

---

## Notion Database Prompts

### Property Configuration Examples

#### Title Property:

```
{  
  "Subscription": {
```

```
    "title": [{
      "text": { "content": "{{ vendor }}" }
    }]
  }
}
```

**Select Property:**

```
{
  "Category": {
    "select": { "name": "{{ category }}" }
  }
}
```

**Number Property:**

```
{
  "Cost": {
    "number": {{ amount }}
  }
}
```

**Date Property:**

```
{
  "Date": {
    "date": { "start": "{{ date }}" }
  }
}
```

## Quick Reference

### Receipt Extraction Prompt

#### Full System Prompt:

```
You are Greg's expense categorization assistant.
```

```
USE ONLY THESE CATEGORIES:
```

- Application (SaaS, apps)
- Website (hosting, domains)
- Cloud (cloud storage)
- Housing (rent, mortgage)
- Utilities (electric, water, internet, phone)
- Equipment (physical items)
- Software (licenses)
- Other (everything else)

```
BILLING: Monthly | Yearly | one time
```

```
Return JSON:
```

```
{  
  "vendor": "Store name",  
  "date": "YYYY-MM-DD",  
  "amount": 123.45,  
  "category": "One of 8 above",  
  "billingCycle": "Monthly|Yearly|one time",  
  "notes": "Brief description",  
  "confidence": 0.95  
}
```

```
Return ONLY JSON, no markdown.
```

---

## Usage Examples

### Example 1: Receipt Processing

Input: [Receipt image of Whole Foods \$87.43]

Output:

```
{  
  "vendor": "Whole Foods",  
  "date": "2026-03-15",  
  "amount": 87.43,  
  "category": "Other",  
  "billingCycle": "one time",  
  "notes": "Weekly groceries",  
  "confidence": 0.95  
}
```

### Example 2: Document Categorization

Input: "Getting Started with N8N Workflows.pdf"

Output:

```
{  
  "category": "N8N Guide",  
  "confidence": 0.98,  
  "reason": "Document title explicitly mentions N8N"  
}
```

## N8N Code Snippets

### JSON Parsing with Validation

```
// Clean and parse AI response
const response = $json.choices[0].message.content;
let cleaned = response.replace(/``json\n?/g, '').replace(/``/g, '').trim();

try {
  const data = JSON.parse(cleaned);

  // Validate category
  const validCategories = ['Application', 'Website', 'Cloud', 'Housing',
'Utilities', 'Equipment', 'Software', 'Other'];
  if (!validCategories.includes(data.category)) {
    data.category = 'Other';
  }

  return { json: data };
} catch (e) {
  throw new Error('Invalid JSON: ' + e.message);
}
```

### Base64 Image Conversion

```
// Convert uploaded file to Base64 for GPT-4o Vision
const binaryData = $input.item.binary.data;
const base64 = binaryData.data.toString('base64');
const mimeType = binaryData.mimeType || 'image/jpeg';

return {
  json: {
    base64Image: base64,
    mimeType: mimeType,
    fileName: $input.item.json.name
  }
};
```

Copy-paste ready prompts for AI, N8N, and Notion  
Organized by category for easy access

*Community Place Prompts Collection*  
*March 2026*