N8N Basic Workflow: ClickUp Tasks to Gmail Report

Simple Workflow Overview

This workflow will:

1. Fetch completed tasks from ClickUp

2. Format them into a professional email

3. Send the report to your superiors via Gmail

Required Nodes (4 Nodes Total)

1. Schedule Trigger

- **Node Type**: Schedule Trigger

- **Purpose**: Automatically run the workflow at set times

- **Configuration**:

  - **Trigger Interval**: `Custom`

  - **Expression**: `0 17 * * 1-5` (5 PM, Monday-Friday)

  - Or set to Daily/Weekly as preferred

2. ClickUp - Get Tasks

- **Node Type**: ClickUp

- **Operation**: Get Tasks

- **Configuration**:

  ```json
  {
    "operation": "getAll",
    "resource": "task",
    "list_id": "YOUR_LIST_ID",
    "status": ["complete"],
    "date_done_gt": "{{ $now.minus({days: 1}).toFormat('yyyy-MM-dd') }}"
  }
  ```

Important: Replace `YOUR_LIST_ID` with your actual ClickUp list ID

3. Code Node - Format Email Content

- **Node Type**: Code

- **Language**: JavaScript

- **Purpose**: Transform ClickUp data into email format

javascript

```javascript
// Get all tasks from ClickUp
const tasks = $input.all();

// If no tasks, create empty report
if (!tasks || tasks.length === 0) {
  return [{
    json: {
      taskCount: 0,
      taskTable: '<p>No tasks completed in the last 24 hours.</p>',
      reportDate: new Date().toLocaleDateString()
    }
  }];
}

// Format each task
const formattedTasks = tasks.map(item => {
  const task = item.json;
  return {
    name: task.name || 'Unnamed Task',
    status: task.status?.status || 'Unknown',
    dateCompleted: task.date_done ? new Date(parseInt(task.date_done)).toLocaleDateString() : '
    assignee: task.assignees?.map(a => a.username).join(', ') || 'Unassigned',
    priority: task.priority?.priority || 'Normal',
    url: task.url || '#'
  };
});

// Create HTML table
let tableHTML = `
<table border="1" cellpadding="10" cellspacing="0" style="border-collapse: collapse; width: 100
  <thead>
    <tr style="background-color: #f5f5f5;">
      <th style="text-align: left; padding: 12px;">Task Name</th>
      <th style="text-align: left; padding: 12px;">Date Completed</th>
      <th style="text-align: left; padding: 12px;">Assignee</th>
      <th style="text-align: left; padding: 12px;">Priority</th>
    </tr>
  </thead>
  <tbody>
`;

formattedTasks.forEach(task => {
  tableHTML += `
```

```
    <tr>
      <td style="padding: 12px;"><a href="${task.url}" target="_blank" style="color: #0066cc; t
      <td style="padding: 12px;">${task.dateCompleted}</td>
      <td style="padding: 12px;">${task.assignee}</td>
      <td style="padding: 12px;">${task.priority}</td>
    </tr>
  `;
});

tableHTML += '</tbody></table>';

return [{
  json: {
    taskCount: formattedTasks.length,
    taskTable: tableHTML,
    reportDate: new Date().toLocaleDateString()
  }
}];
```

4. Gmail - Send Email

- **Node Type**: Gmail

- **Operation**: Send Email

- **Configuration**:

Recipients:

- **To**: `manager@company.com` (replace with actual email)

- **CC**: `team-lead@company.com` (optional)

Subject: `Daily Task Report - {{ $now.toFormat('MMM dd, yyyy') }}`

Email Body (HTML):

html

```html
<!DOCTYPE html>
<html>
<head>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 0;
            padding: 20px;
            background-color: #f9f9f9;
        }
        .container {
            max-width: 800px;
            margin: 0 auto;
            background-color: white;
            border-radius: 8px;
            overflow: hidden;
            box-shadow: 0 2px 10px rgba(0,0,0,0.1);
        }
        .header {
            background-color: #4CAF50;
            color: white;
            padding: 20px;
            text-align: center;
        }
        .content {
            padding: 30px;
        }
        .summary {
            background-color: #f8f9fa;
            padding: 20px;
            border-radius: 6px;
            margin-bottom: 25px;
            border-left: 4px solid #4CAF50;
        }
        .footer {
            background-color: #f1f1f1;
            padding: 15px;
            text-align: center;
            font-size: 12px;
            color: #666;
        }
        table {
            width: 100%;
```

```html
            margin-top: 15px;
        }
        th {
            background-color: #f5f5f5;
            font-weight: bold;
        }
        tr:nth-child(even) {
            background-color: #f9f9f9;
        }
    </style>
</head>
<body>
    <div class="container">
        <div class="header">
            <h2>📋 Daily Task Completion Report</h2>
            <p>{{ $json.reportDate }}</p>
        </div>

        <div class="content">
            <div class="summary">
                <h3>📊 Summary</h3>
                <p><strong>Tasks Completed:</strong> {{ $json.taskCount }}</p>
                <p><strong>Report Period:</strong> Last 24 hours</p>
            </div>

            <h3>✅ Completed Tasks</h3>
            {{ $json.taskTable }}
        </div>

        <div class="footer">
            <p>🤖 This report was automatically generated from ClickUp</p>
            <p>Generated at {{ $now.toFormat('HH:mm') }} on {{ $now.toFormat('MMM dd, yyyy') }}</p>
        </div>
    </div>
</body>
</html>
```

Setup Steps

Step 1: Create Workflow

1. Open N8N

2. Click "New Workflow"

3. Add the 4 nodes in sequence

Step 2: Connect ClickUp

1. Add ClickUp credentials in N8N settings

2. Get your API token from ClickUp → Settings → Apps → API Token

3. Find your List ID from the ClickUp URL or API

Step 3: Connect Gmail

1. Add Gmail credentials

2. Use OAuth2 authentication

3. Grant necessary permissions

Step 4: Test

1. Click "Execute Workflow" manually first

2. Check if email is received

3. Verify task data is correct

Step 5: Activate

1. Save the workflow

2. Turn on the Schedule Trigger

3. The workflow will run automatically

Quick Troubleshooting

- **No tasks found**: Check your List ID and date filter

- **Email not sending**: Verify Gmail credentials and permissions

- **Missing data**: Ensure ClickUp API token has proper access

- **Wrong date format**: Adjust timezone in the date expressions

This basic workflow will send you clean, professional task reports automatically!