

- BPM API 文件
 - 基本資訊
 - API 端點
 - 1. 獲取所有部門等級資訊
 - 請求
 - 響應
 - 欄位說明
 - 樣本數據
 - Python 範例
 - 2. 獲取所有部門資訊
 - 請求
 - 響應
 - 欄位說明
 - 樣本數據
 - Python 範例
 - 3. 獲取所有部門結構資訊
 - 請求
 - 響應
 - 欄位說明
 - 樣本數據
 - Python 範例
 - 4. 獲取所有職稱對應等級資訊
 - 請求
 - 響應
 - 欄位說明
 - 樣本數據
 - Python 範例
 - 5. 獲取所有成員資訊
 - 請求
 - 響應
 - 欄位說明
 - 樣本數據
 - Python 範例
 - 6. 獲取所有成員結構資訊
 - 請求
 - 響應
 - 欄位說明

- 樣本數據
- Python 範例
- 7. 查詢實體化視圖變更資訊
 - 請求
 - 響應
 - 欄位說明
 - 樣本數據 - fse7en_org_deptgradeinfo
 - 樣本數據 -fse7en_org_deptinfo
 - 樣本數據 -fse7en_org_deptstruct
 - 樣本數據 -fse7en_org_jobtitle2grade
 - 樣本數據 -fse7en_org_memberinfo
 - 樣本數據 -fse7en_org_memberstruct *複合主鍵
 - Python 範例

BPM API 文件

本文件提供 BPM API 的詳細說明，包括所有端點的輸入、輸出規格以及使用範例。

基本資訊

- 基礎 URL: `/api/v1/bpm`
- 內容類型: `application/json`
- 呼叫 Header Token:
 - "Authorization": "xxx"

API 端點

1. 獲取所有部門等級資訊

獲取系統中所有部門等級的相關資訊。

請求

- 方法: `GET`
- URL: `/api/v1/bpm/dept-grade-info`

- 參數: 無

響應

- 成功響應 (200 OK):

```
[
  {
    "gradeId": "string",
    "displayName": "string",
    "gradeNum": "string"
  }
]
```

- 無內容 (204 No Content): 未找到部門等級資訊
- 伺服器錯誤 (500 Internal Server Error): 處理請求時發生錯誤

欄位說明

欄位名稱	類型	描述
gradeId	字串	部門層級代碼
displayName	字串	部門名稱
gradeNum	字串	部門層級數值（整數），數字愈大層級愈高

樣本數據

```
[
  {
    "gradeId": "Level1",
    "displayName": "第1階",
    "gradeNum": "-1"
  },
  {
    "gradeId": "Level2",
    "displayName": "第2階",
    "gradeNum": "-2"
  },
  {
    "gradeId": "Level3",
    "displayName": "第3階",
    "gradeNum": "-3"
  }
]
```

```
}  
]
```

Python 範例

```
import requests  
  
def get_all_dept_grade_info():  
    url = "http://your-server/api/v1/bpm/dept-grade-info"  
  
    headers = {  
        "Authorization": "xxx"  
    }  
  
    try:  
        response = requests.get(url, headers=headers)  
  
        if response.status_code == 200:  
            return response.json()  
        elif response.status_code == 204:  
            print("未找到部門等級資訊")  
            return []  
        else:  
            print(f"錯誤: {response.status_code}, {response.text}")  
            return None  
    except Exception as e:  
        print(f"請求異常: {str(e)}")  
        return None
```

2. 獲取所有部門資訊

獲取系統中所有部門的基本資訊。

請求

- 方法: GET
- URL: /api/v1/bpm/dept-info
- 參數: 無

響應

- 成功響應 (200 OK):

```
[
  {
    "orgCode": "string",
    "orgName": "string"
  }
]
```

- **無內容 (204 No Content):** 未找到部門資訊
- **伺服器錯誤 (500 Internal Server Error):** 處理請求時發生錯誤

欄位說明

欄位名稱	類型	描述
orgCode	字串	部門的唯一碼 部門編號 (HrCode=DeptID)
orgName	字串	給一般使用者看的名稱

樣本數據

```
[
  {
    "orgCode": "100110011",
    "orgName": "總公司"
  },
  {
    "orgCode": "100111000",
    "orgName": "董事會"
  },
  {
    "orgCode": "100111100",
    "orgName": "董事長室"
  }
]
```

Python 範例

```
import requests

def get_all_dept_info():
    url = "http://your-server/api/v1/bpm/dept-info"

    headers = {
```

```
        "Authorization": "xxx"
    }

    try:
        response = requests.get(url, headers=headers)

        if response.status_code == 200:
            return response.json()
        elif response.status_code == 204:
            print("未找到部門資訊")
            return []
        else:
            print(f"錯誤: {response.status_code}, {response.text}")
            return None
    except Exception as e:
        print(f"請求異常: {str(e)}")
        return None
```

3. 獲取所有部門結構資訊

獲取系統中所有部門的結構關係資訊。

請求

- 方法: GET
- URL: /ee-midd/api/v1/bpm/dept-struct
- 參數: 無

響應

- 成功響應 (200 OK):

```
[
  {
    "orgCode": "string",
    "parentOrgCode": "string",
    "gradeId": "string",
    "gradeNum": "string",
    "topOrgtreeCode": "string"
  }
]
```

- 無內容 (204 No Content): 未找到部門結構資訊

- 伺服器錯誤 (500 Internal Server Error): 處理請求時發生錯誤

欄位說明

欄位名稱	類型	描述
orgCode	字串	部門識別碼
parentOrgCode	字串	上層部門識別碼
gradeId	字串	部門層級編號
gradeNum	字串	部門層級數值
topOrgtreeCode	字串	最上層部門代碼

樣本數據

```
[
  {
    "orgCode": "100110011",
    "parentOrgCode": null,
    "gradeId": "Level1",
    "gradeNum": "-1",
    "topOrgtreeCode": "100110011"
  },
  {
    "orgCode": "100111000",
    "parentOrgCode": "100110011",
    "gradeId": "Level2",
    "gradeNum": "-2",
    "topOrgtreeCode": "100110011"
  },
  {
    "orgCode": "100111100",
    "parentOrgCode": "100111000",
    "gradeId": "Level3",
    "gradeNum": "-3",
    "topOrgtreeCode": "100110011"
  },
  {
    "orgCode": "100111110",
    "parentOrgCode": "100111100",
    "gradeId": "Level4",
    "gradeNum": "-4",
    "topOrgtreeCode": "100110011"
  },
  {
    "orgCode": "100111120",
    "parentOrgCode": "100111100",
    "gradeId": "Level4",
    "gradeNum": "-4",
    "topOrgtreeCode": "100110011"
  }
]
```

```
    "topOrgtreeCode": "100110011"  
  }  
]
```

Python 範例

```
import requests  
  
def get_all_dept_struct():  
    url = "http://your-server/ee-midd/api/v1/bpm/dept-struct"  
  
    headers = {  
        "Authorization": "xxx"  
    }  
  
    try:  
        response = requests.get(url, headers=headers)  
  
        if response.status_code == 200:  
            return response.json()  
        elif response.status_code == 204:  
            print("未找到部門結構資訊")  
            return []  
        else:  
            print(f"錯誤: {response.status_code}, {response.text}")  
            return None  
    except Exception as e:  
        print(f"請求異常: {str(e)}")  
        return None
```

4. 獲取所有職稱對應等級資訊

獲取系統中所有職稱與等級的對應關係。

請求

- 方法: GET
- URL: /api/v1/bpm/job-title-grade
- 參數: 無

響應

- 成功響應 (200 OK):


```
[
  {
    "jobTitleCode": "string",
    "jobTitleName": "string",
    "jobGrade": "string"
  }
]
```

- **無內容 (204 No Content):** 未找到職稱對應等級資訊
- **伺服器錯誤 (500 Internal Server Error):** 處理請求時發生錯誤

欄位說明

欄位名稱	類型	描述
jobTitleCode	字串	職稱的唯一碼
jobTitleName	字串	給一般使用者看的名稱
jobGrade	字串	該職稱的等級(數字愈大愈大)

樣本數據

```
[
  {
    "jobTitleCode": "T00",
    "jobTitleName": "實習生",
    "jobGrade": "00"
  },
  {
    "jobTitleCode": "P13",
    "jobTitleName": "專案資深經理",
    "jobGrade": "13"
  },
  {
    "jobTitleCode": "P12",
    "jobTitleName": "專案經理",
    "jobGrade": "12"
  },
  {
    "jobTitleCode": "P11",
    "jobTitleName": "專案副理",
    "jobGrade": "11"
  }
]
```

```
import requests

def get_all_job_title_grade():
    url = "http://your-server/api/v1/bpm/job-title-grade"

    headers = {
        "Authorization": "xxx"
    }

    try:
        response = requests.get(url, headers=headers)

        if response.status_code == 200:
            return response.json()
        elif response.status_code == 204:
            print("未找到職稱對應等級資訊")
            return []
        else:
            print(f"錯誤: {response.status_code}, {response.text}")
            return None
    except Exception as e:
        print(f"請求異常: {str(e)}")
        return None
```

5. 獲取所有成員資訊

獲取系統中所有成員的基本資訊。

請求

- 方法: GET
- URL: /api/v1/bpm/member-info
- 參數: 無

響應

- 成功響應 (200 OK):

```
[
  {
    "employeeNo": "string",
    "fullName": "string",
```

```
    "extNo": "string",
    "emailAddress": "string",
    "isterminated": "string",
    "hireDate": "yyyy-MM-dd",
    "azureAccount": "string"
  }
]
```

- 無內容 (204 No Content): 未找到成員資訊
- 伺服器錯誤 (500 Internal Server Error): 處理請求時發生錯誤

欄位說明

欄位名稱	類型	描述
employeeNo	字串	人員的唯一碼
fullName	字串	人員顯示名稱(中文)
extNo	字串	分機號碼
emailAddress	字串	電子郵件
isterminated	字串	帳號是否可用 (0:停用, 1:啟用)
hireDate	日期	到職日期 (格式: yyyy-MM-dd)
azureAccount	字串	Azure AD 帳號 (Email)

樣本數據

```
[
  {
    "employeeNo": "004457",
    "fullName": "陳彥樺",
    "extNo": "2864",
    "emailAddress": "004457@sogo.com.tw",
    "isterminated": "0",
    "hireDate": "2002-10-01",
    "azureAccount": "004457@sogo.com.tw"
  },
  {
    "employeeNo": "004466",
    "fullName": "馮縱才",
    "extNo": "500",
    "emailAddress": "004466@sogo.com.tw",
    "isterminated": "0",
    "hireDate": "2002-10-09",
    "azureAccount": "004466@sogo.com.tw"
  }
]
```

```

    },
    {
        "employeeNo": "004490",
        "fullName": "蕭淑鑾",
        "extNo": "1534",
        "emailAddress": "004490@sogo.com.tw",
        "isTerminated": "0",
        "hireDate": "2003-01-01",
        "azureAccount": "004490@sogo.com.tw"
    }
]

```

Python 範例

```

import requests
from datetime import datetime

def get_all_member_info():
    url = "http://your-server/api/v1/bpm/member-info"

    headers = {
        "Authorization": "xxx"
    }

    try:
        response = requests.get(url, headers=headers)

        if response.status_code == 200:
            return response.json()
        elif response.status_code == 204:
            print("未找到成員資訊")
            return []
        else:
            print(f"錯誤: {response.status_code}, {response.text}")
            return None
    except Exception as e:
        print(f"請求異常: {str(e)}")
        return None

```

6. 獲取所有成員結構資訊

獲取系統中所有成員的組織結構關係。

請求

- 方法: GET
- URL: /api/v1/bpm/member-struct

- 參數: 無

響應

- 成功響應 (200 OK):

```
[
  {
    "employeeNo": "string",
    "orgCode": "string",
    "fullName": "string",
    "orgName": "string",
    "jobTitleCode": "string",
    "jobGrade": "string",
    "isMainJob": "string",
    "approveRight": "string",
    "enable": "string",
    "instructor": "string"
  }
]
```

- 無內容 (204 No Content): 未找到成員結構資訊
- 伺服器錯誤 (500 Internal Server Error): 處理請求時發生錯誤

欄位說明

欄位名稱	類型	描述
employeeNo	字串	人員的唯一碼
orgCode	字串	部門的唯一碼
fullName	字串	人員名稱
orgName	字串	組織名稱
jobTitleCode	字串	職稱的唯一碼
jobGrade	字串	職級
isMainJob	字串	是否專職 (1: 轉職 0: 兼職)
approveRight	字串	是否有簽核權 (1: 啟用 0: 停用)
enable	字串	是否預設啟用(1:啟用 0:停用)
instructor	字串	* 單線上級主管：簽核主管員編@該主管的實體組織代號

樣本數據

```
[
  {
    "employeeNo": "005187",
    "orgCode": "100114110",
    "fullName": "魏琮祐",
    "orgName": "人事課",
    "jobTitleCode": "M11",
    "jobGrade": "11",
    "isMainJob": "1",
    "approveRight": "1",
    "enable": "1",
    "instructor": "005704@100114100"
  },
  {
    "employeeNo": "005187",
    "orgCode": "100114120",
    "fullName": "魏琮祐",
    "orgName": "訓練發展課",
    "jobTitleCode": "M11",
    "jobGrade": "11",
    "isMainJob": "0",
    "approveRight": "1",
    "enable": "1",
    "instructor": "005704@100114100"
  },
  {
    "employeeNo": "500138",
    "orgCode": "100114130",
    "fullName": "鍾怡君",
    "orgName": "顧客服務中心",
    "jobTitleCode": "M10",
    "jobGrade": "10",
    "isMainJob": "1",
    "approveRight": "1",
    "enable": "1",
    "instructor": "005704@100114100"
  }
]
```

Python 範例

```
import requests

def get_all_member_struct():
    url = "http://your-server/api/v1/bpm/member-struct"

    headers = {
        "Authorization": "xxx"
    }
```

```

try:
    response = requests.get(url, headers=headers)

    if response.status_code == 200:
        return response.json()
    elif response.status_code == 204:
        print("未找到成員結構資訊")
        return []
    else:
        print(f"錯誤: {response.status_code}, {response.text}")
        return None
except Exception as e:
    print(f"請求異常: {str(e)}")
    return None

```

7. 查詢實體化視圖變更資訊

根據日期和視圖名稱查詢實體化視圖的變更資訊。

請求

- 方法: **GET**
- URL: **/api/v1/bpm/view-changes**
- 參數:
 - **date** (必填): 查詢日期，格式為 yyyy-MM-dd
 - **view_name** (可選)，若沒有填就是該日期所有視圖全列: 視圖名稱列表
 - fse7en_org_deptgradeinfo
 - fse7en_org_deptinfo
 - fse7en_org_deptstruct
 - fse7en_org_jobtitle2grade
 - fse7en_org_memberinfo
 - fse7en_org_memberstruct

響應

- 成功響應 (200 OK):

```

[
  {
    "viewName": "string",
    "refreshedAt": "yyyy-MM-dd HH:mm:ss",
    "diffCount": number,

```

```
    "diffDetails": [
      {
        "key": {
          "field1": "value1",
          "field2": "value2"
        },
        "data": {
          "field1": "value1",
          "field2": "value2",
          "field3": "value3"
        },
        "operation": "string"
      }
    ]
  }
}
```

- **無內容 (204 No Content):** 未找到實體化視圖變更資訊
- **伺服器錯誤 (500 Internal Server Error):** 處理請求時發生錯誤

欄位說明

欄位名稱	類型	描述
viewName	字串	視圖名稱
refreshedAt	日期時間	更新時間 (格式: yyyy-MM-dd HH:mm:ss)
diffCount	數字	變更記錄數量
diffDetails	陣列	變更詳細資訊陣列
diffDetails[].key	對象	變更記錄的主鍵資訊(不同 view 有不同主鍵)
diffDetails[].data	對象	變更記錄的資料內容(依不同操作類型有不同結構)
diffDetails[].operation	字串	操作類型 (INSERT, UPDATE, DELETE)

[註]欄位說明部分比較不亦說明，請參考參考樣本數據比較好了解

樣本數據 - fse7en_org_deptgradeinfo

```
[
  {
    "viewName": "fse7en_org_deptgradeinfo",
    "refreshedAt": "2025-03-11 10:15:30",
    "diffCount": 3,
    "diffDetails": [
```



```

{
  "key": {
    "gradeId": "Level1"
  },
  "data": {
    "gradeId": "Level1",
    "displayName": "第1階",
    "gradeNum": "-1"
  },
  "operation": "INSERT"
},
{
  "key": {
    "gradeId": "Level2"
  },
  "new": {
    "gradeId": "Level2",
    "displayName": "第2階",
    "gradeNum": "-2"
  },
  "old": {
    "gradeId": "Level2",
    "displayName": "第8階",
    "gradeNum": "-2"
  },
  "operation": "UPDATE"
},
{
  "key": {
    "gradeId": "Level3"
  },
  "data": {
    "gradeId": "Level3",
    "displayName": "第3階",
    "gradeNum": "-3"
  },
  "operation": "DELETE"
}
]
}
]

```

樣本數據 -fse7en_org_deptinfo

```

[
  {
    "viewName": "fse7en_org_deptinfo",
    "refreshedAt": "2025-03-11 10:15:30",
    "diffCount": 3,
    "diffDetails": [
      {
        "key": {
          "orgCode": "100314110"
        },

```

```

    "data": {
      "orgCode": "100314110",
      "orgName": "總務課/復興"
    },
    "operation": "INSERT"
  },
  {
    "key": {
      "orgCode": "100213240"
    },
    "new": {
      "orgCode": "100213240",
      "orgName": "童裝內衣課/忠孝"
    },
    "old": {
      "orgCode": "100213240",
      "orgName": "內衣課/忠孝"
    },
    "operation": "UPDATE"
  },
  {
    "key": {
      "orgCode": "100213260"
    },
    "data": {
      "orgCode": "100213260",
      "orgName": "童裝課/忠孝"
    },
    "operation": "DELETE"
  }
]
}
]

```

樣本數據 -fse7en_org_deptstruct

```

[
  {
    "viewName": "fse7en_org_deptstruct",
    "refreshedAt": "2025-03-11 10:15:30",
    "diffCount": 3,
    "diffDetails": [
      {
        "key": {
          "orgCode": "100314110"
        },
        "data": {
          "gradeId": "Level8",
          "orgCode": "100314110",
          "gradeNum": "-8",
          "parentOrgCode": "100314100",
          "topOrgtreeCode": "100110011"
        },
        "operation": "INSERT"
      }
    ]
  }
]

```

```

    },
    {
      "key": {
        "orgCode": "100114200"
      },
      "new": {
        "gradeId": "Level6",
        "orgCode": "100114200",
        "gradeNum": "-6",
        "parentOrgCode": "100114000",
        "topOrgtreeCode": "100110011"
      },
      "old": {
        "gradeId": "Level5",
        "orgCode": "100114200",
        "gradeNum": "-5",
        "parentOrgCode": "100111206",
        "topOrgtreeCode": "100110011"
      },
      "operation": "UPDATE"
    },
    {
      "key": {
        "orgCode": "100213260"
      },
      "data": {
        "gradeId": "Level8",
        "orgCode": "100213260",
        "gradeNum": "-8",
        "parentOrgCode": "100213200",
        "topOrgtreeCode": "100110011"
      },
      "operation": "DELETE"
    }
  ]
}
]

```

樣本數據 -fse7en_org_jobtitle2grade

```

[
  {
    "viewName": "fse7en_org_jobtitle2grade",
    "refreshedAt": "2025-03-11 10:15:30",
    "diffCount": 3,
    "diffDetails": [
      {
        "key": {
          "jobTitleCode": "M13"
        },
        "data": {
          "jobGrade": "13",
          "jobTitleCode": "M13",
          "jobTitleName": "資深主任工程師"
        }
      }
    ]
  }
]

```

```

    },
    "operation": "INSERT"
  },
  {
    "key": {
      "jobTitleCode": "P11"
    },
    "new": {
      "jobGrade": "11",
      "jobTitleCode": "P11",
      "jobTitleName": "專案副理"
    },
    "old": {
      "jobGrade": "11",
      "jobTitleCode": "P11",
      "jobTitleName": "副理"
    },
    "operation": "UPDATE"
  },
  {
    "key": {
      "jobTitleCode": "T00"
    },
    "data": {
      "jobGrade": "00",
      "jobTitleCode": "T00",
      "jobTitleName": "實習生"
    },
    "operation": "DELETE"
  }
]
}
]

```

樣本數據 -fse7en_org_memberinfo

```

[
  {
    "viewName": "fse7en_org_memberinfo",
    "refreshedAt": "2025-03-11 10:15:30",
    "diffCount": 3,
    "diffDetails": [
      {
        "key": {
          "employeeNo": "007737"
        },
        "data": {
          "extNo": "8547",
          "fullName": "呂博寧",
          "hireDate": "2024-11-05T00:00:00",
          "employeeNo": "007737",
          "azureaccount": "007737@sogo.com.tw",
          "isterminated": "0",
          "emailAddress": "007737@sogo.com.tw"
        }
      }
    ]
  }
]

```

```

    },
    "operation": "INSERT"
  },
  {
    "key": {
      "employeeNo": "900388"
    },
    "new": {
      "extNo": "6327",
      "fullName": "鄧雅云",
      "hireDate": "2021-10-20T00:00:00",
      "employeeNo": "900388",
      "azureaccount": "900388@sogo.com.tw",
      "isterminated": "0",
      "emailAddress": "900388@sogo.com.tw"
    },
    "old": {
      "extNo": "6804",
      "fullName": "鄧雅云",
      "hireDate": "2021-10-20T00:00:00",
      "employeeNo": "900388",
      "azureaccount": "900388@sogo.com.tw",
      "isterminated": "0",
      "emailAddress": "900388@sogo.com.tw"
    },
    "operation": "UPDATE"
  },
  {
    "key": {
      "employeeNo": "007751"
    },
    "data": {
      "extNo": "8731",
      "fullName": "許宇青",
      "hireDate": "2025-01-02T00:00:00",
      "employeeNo": "007751",
      "azureaccount": "007751@sogo.com.tw",
      "isterminated": "0",
      "emailAddress": "007751@sogo.com.tw"
    },
    "operation": "DELETE"
  }
]
}
]

```

樣本數據 -fse7en_org_memberstruct *複合主鍵

```

[
  {
    "viewName": "fse7en_org_memberstruct",
    "refreshedAt": "2025-03-11 10:15:30",
    "diffCount": 3,
    "diffDetails": [

```

```
{
  "key": {
    "orgCode": "100113100",
    "employeeNo": "000065"
  },
  "data": {
    "enable": "1",
    "orgCode": "100113100",
    "orgName": "營業管理室",
    "fullName": "吳素吟",
    "jobGrade": "17",
    "instructor": "006253@100111205",
    "employeeNo": "000065",
    "isMainJob": "0",
    "approveRight": "1",
    "jobTitleCode": "M17"
  },
  "operation": "INSERT"
},
{
  "key": {
    "orgCode": "100112110",
    "employeeNo": "004007"
  },
  "new": {
    "enable": "1",
    "orgCode": "100112110",
    "orgName": "企劃課",
    "fullName": "邱信宏",
    "jobGrade": "12",
    "instructor": "003295@100112100",
    "employeeNo": "004007",
    "isMainJob": "1",
    "approveRight": "1",
    "jobTitleCode": "M12"
  },
  "old": {
    "enable": "1",
    "orgCode": "100112110",
    "orgName": "企劃課",
    "fullName": "邱信宏",
    "jobGrade": "11",
    "instructor": "003295@100112100",
    "employeeNo": "004007",
    "isMainJob": "1",
    "approveRight": "1",
    "jobTitleCode": "M11"
  },
  "operation": "UPDATE"
},
{
  "key": {
    "orgCode": "100111201",
    "employeeNo": "000004"
  },
  "data": {
    "enable": "0",
```

```

        "orgCode": "100111201",
        "orgName": null,
        "fullName": "李光榮",
        "jobGrade": "130",
        "instructor": null,
        "employeeNo": "000004",
        "isMainJob": "0",
        "approveRight": "1",
        "jobTitleCode": "130"
    },
    "operation": "DELETE"
}
]
}
]

```

Python 範例

```

import requests
from datetime import datetime

def get_view_changes_by_date(date, view_name=None):
    url = "http://your-server/api/v1/bpm/view-changes"

    headers = {
        "Authorization": "xxx"
    }

    params = {
        "date": date
    }

    if view_name:
        params["view_name"] = view_name

    try:
        response = requests.get(url, params=params, headers=headers)

        if response.status_code == 200:
            return response.json()
        elif response.status_code == 204:
            print("未找到實體化視圖變更資訊")
            return []
        else:
            print(f"錯誤: {response.status_code}, {response.text}")
            return None
    except Exception as e:
        print(f"請求異常: {str(e)}")
        return None

# 使用範例
# 查詢特定日期和視圖的變更
specific_changes = get_view_changes_by_date("2025-03-11",
"fse7en_org_memberstruct")

```

```
if specific_changes:
    for change in specific_changes:
        print(f"視圖: {change['viewName']}, 更新時間: {change['refreshedAt']},
變更數量: {change['diffCount']}")

    # 處理變更詳細資訊
    if change['diffDetails'] and len(change['diffDetails']) > 0:
        for diff in change['diffDetails'][:5]: # 只顯示前5筆變更
            print(f"操作類型: {diff['operation']}")
            print(f"主鍵: {diff['key']}")
            print(f"資料: {diff['data']}")
            print("----")
```