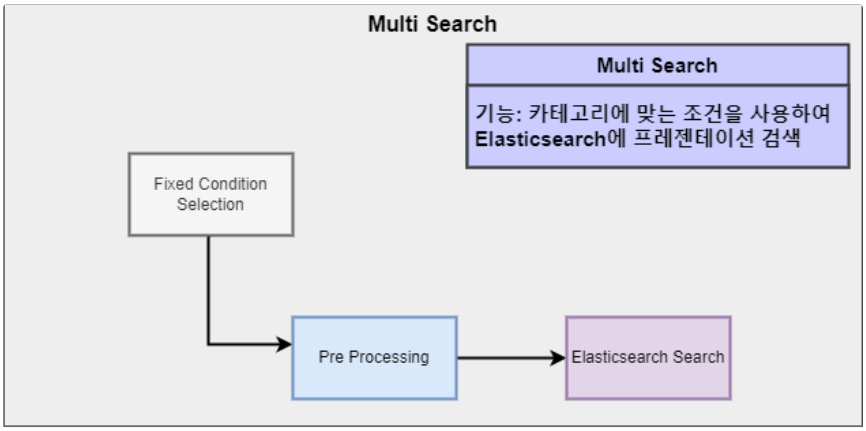
Multi Search API Documents

Multi Search API Documents Table of contents

Kibana

- **API 기능 요약**
- **API 사용 설명서**
- **API 코드 설명**

★ API 기능 요약



Pre Processing

조건에 맞는 프레젠테이션을 검색하기 위해 **Elasticsearch** 문법 처리

pagination, price, likes, sort, main, middle, mood, style 등 처리

단독 검색, 조합 검색 등에 따라 처리

다면 검색을 위한 API로써, 각 카테고리를 이용하여 검색

★ API 사용 설명서

해당 API는 3000번 포트를 사용.

	/category_search
methods	POST, GET

end_point	/category_search POST http://192.168.1.46:3000/category_search

parameters	JSON
	{
	"main": "",
	"page": 1,
	"middle": "",
	"price": [0, 10000],
	"moods": "",
	"styles": "",
	"perPage": 20,
	"sort": "",
	"likes": [0, 100],
	"isSystem": "True"
	}

★API 코드 설명

- **pre-processing**

전달받은 Parameter를 전처리하여 Elasticsearch에 검색이 되도록 처리하는 code

```
    try:
        # pagination
        try:
            page_number = int(parameters['page'])
        except:
            page_number = 1

        if page_number == 1:
            from_range = 0
        elif page_number >= 2:
            from_range = (page_number-1)*parameters['perPage']

        # setting price range
        if not parameters['price']:
            price_range = {
                'gte': 0
            }
        else:
            if len(parameters['price']) == 1: # 이하 처리
                price_range = {
                    'gte': 0,  # gte: 이상 / gte: 초과
                    'lte': int(parameters['price'][0])  # lte: 이하 / lt: 미만
                }
            elif len(parameters['price']) == 2:
                price_range = {
                    'gte': int(parameters['price'][0]),
                    'lte': int(parameters['price'][1])
                }

        # setting likes range
        if not parameters['likes']:
            likes_range = {
                'gte': 0
            }
        else:
            if len(parameters['likes']) == 1:  # 이하 처리
```

```
                likes_range = {
                    'gte': 0,  # gte: 이상 / gte: 초과
                    'lte': int(parameters['likes'][0])  # lte: 이하 / lt: 미만
                }
            elif len(parameters['price']) == 2:
                likes_range = {
                    'gte': int(parameters['likes'][0]),
                    'lte': int(parameters['likes'][1])
                }

        # setting sort
        Tosort = {
            'updatedDate': {
                'order': 'desc'}
        }
        try:
            if parameters['sort'] == 'updatedDate':
                Tosort = {
                    'updatedDate': {
                        'order': 'desc'}
                }
            elif parameters['sort'] == 'downloadCount':
                Tosort = {
                    'downloadCount': {
                        'order': 'desc'}
                }
            elif parameters['sort'] == 'low_price':
                Tosort = {
                    'price': {
                        'order': 'asc'}
                }
            elif parameters['sort'] == 'high_price':
                Tosort = {
                    'price': {
                        'order': 'desc'}
                }
        except:
            Tosort = {
                'updatedDate': {
                    'order': 'desc'}
            }
```

- **case 1: 아무것도 선택하지 않았을 경우**

아무것도 선택하지 않고 검색을 눌렀을 경우에 적용되는 code (현재는 사용x)

```
        # When you didn't choose anything
        if not parameters['price'] and (not parameters['main']) and (not parameters['moods']) and (not parameter
            body = {
                'query': {
                    'bool': {
                        'must': {
                            'match_all': {}, # 모든 것 검색
                        },
                        'filter': [
                            {
```

```
                                "match": {
                                    "isSystem": parameters['isSystem']
                                }
                            },
                        ]
                    }
                },
                'from': from_range,
                'size': parameters['perPage'],
                'sort': [
                    Tosort
                ]
            }

            category_search_result = es.search(index='presentations', body=body)

            for i in category_search_result['hits']['hits']:
                if i['_id'] not in res['_id']:  # 중복제거
                    # Presentation Information
                    i['_source']['_id'] = i['_id']
                    res['_id'].append(i['_id'])
                    res['Info'].append(i['_source'])

                    # Calculate Presentation Page
                    pages['current'] = parameters['page']
                    if parameters['perPage'] == 1:
                        pages['prev'] = 0
                        parameters['hasPrev'] = False
                    else:
                        pages['prev'] = parameters['page']-1
                        parameters['hasPrev'] = True

                    if category_search_result['hits']['total']['value'] / (parameters['perPage']*parameters['pag
                        pages['next'] = pages['current'] + 1
                        pages['hasNext'] = True
                    else:
                        pages['next'] = pages['current']
                        pages['hasNext'] = False

                    if round(category_search_result['hits']['total']['value'] // parameters['perPage']) == 0:
                        pages['total'] = 1
                    else:
                        pages['total'] = round(category_search_result['hits']['total']['value'] // parameters['p

                    # Calculate Presentation number
                    items['begin'] = 1
                    items['end'] = category_search_result['hits']['total']['value']
                    items['total'] = category_search_result['hits']['total']['value']

            result['data'] = res['Info']
            result['pages'] = pages
            result['items'] = items

            return jsonify(result)

```

- **case 2: 가격만 선택하여 검색할 경우**

가격만 선택하여 검색을 진행했을 경우 code

```
        # only select price
        if parameters['price'] and (not parameters['main']) and (not parameters['moods']) and (not parameters['
            body = {
                'query': {
                    'bool': {
                        'must': [
                            {
                                'range': {
                                    'price': price_range # 초기에 Pre-processing에서 처리된 범위 사용
                                }
                            }
                        ],
                        'filter': [
                            {
                                "match": {
                                    "isSystem": parameters['isSystem']
                                }
                            },
                        ]
                    }
                },
                'from': from_range,
                'size': parameters['perPage'],
                'sort': [
                    Tosort
                ]
            }
            category_search_result = es.search(index='presentations', body=body)
            for i in category_search_result['hits']['hits']:
                if i['_id'] not in res['_id']:  # 중복제거
                    # Presentation Information
                    i['_source']['_id'] = i['_id']
                    res['_id'].append(i['_id'])
                    res['Info'].append(i['_source'])

                    # Calculate Presentation Page
                    pages['current'] = parameters['page']
                    if parameters['perPage'] == 1:
                        pages['prev'] = 0
                        parameters['hasPrev'] = False
                    else:
                        pages['prev'] = parameters['page'] - 1
                        parameters['hasPrev'] = True

                    if category_search_result['hits']['total']['value'] / (
                            parameters['perPage'] * parameters['page']) != 0:
                        pages['next'] = pages['current'] + 1
                        pages['hasNext'] = True
                    else:
                        pages['next'] = pages['current']
                        pages['hasNext'] = False

                    if round(category_search_result['hits']['total']['value'] // parameters['perPage']) == 0:
                        pages['total'] = 1
                    else:
```

```
                    pages['total'] = round(category_search_result['hits']['total']['value'] // parameters['p

                # Calculate Presentation number
                items['begin'] = 1
                items['end'] = category_search_result['hits']['total']['value']
                items['total'] = category_search_result['hits']['total']['value']

        result['data'] = res['Info']
        result['pages'] = pages
        result['items'] = items

        return jsonify(result)
```

- **case 3: 분위기만 선택하여 검색할 경우**

분위기만 선택하여 검색을 진행했을 경우 code

```
    # only select moods
    if parameters['moods'] and (not parameters['main']) and (not parameters['price']) and (not parameters['
        body = {
            'query': {
                'bool': {
                    'must': [
                        {
                            'match': {
                                    'moods.moodName': parameters['moods'], # 초기 pre-procssing에서 처리한 분위
                            }
                        }
                    ],
                    'filter': [
                        {
                            "match": {
                                "isSystem": parameters['isSystem']
                            }
                        },
                    ]
                }
            },
            'from': from_range,
            'size': parameters['perPage'],
            'sort': [
                Tosort
            ]
        }
        category_search_result = es.search(index='presentations', body=body)
        for i in category_search_result['hits']['hits']:
            if i['_id'] not in res['_id']:  # 중복제거
                # Presentation Information
                i['_source']['_id'] = i['_id']
                res['_id'].append(i['_id'])
                res['Info'].append(i['_source'])

                # Calculate Presentation Page
                pages['current'] = parameters['page']
                if parameters['perPage'] == 1:
                    pages['prev'] = 0
```

```
                    parameters['hasPrev'] = False
                else:
                    pages['prev'] = parameters['page'] - 1
                    parameters['hasPrev'] = True

                if category_search_result['hits']['total']['value'] / (
                        parameters['perPage'] * parameters['page']) != 0:
                    pages['next'] = pages['current'] + 1
                    pages['hasNext'] = True
                else:
                    pages['next'] = pages['current']
                    pages['hasNext'] = False

                if round(category_search_result['hits']['total']['value'] // parameters['perPage']) == 0:
                    pages['total'] = 1
                else:
                    pages['total'] = round(category_search_result['hits']['total']['value'] // parameters['p

                # Calculate Presentation number
                items['begin'] = 1
                items['end'] = category_search_result['hits']['total']['value']
                items['total'] = category_search_result['hits']['total']['value']

            result['data'] = res['Info']
            result['pages'] = pages
            result['items'] = items

            return jsonify(result)
```

- **case 4: 좋아요만 선택하여 검색할 경우**

좋아요만 선택하여 검색을 진행했을 경우 code

```
        # only select likes
        if parameters['likes'] and (not parameters['main']) and (not parameters['price']) and (not parameters['m
            body = {
                'query': {
                    'bool': {
                        'must': [
                            {
                                'range': {
                                    'price': likes_range
                                }
                            }
                        ],
                        'filter': [
                            {
                                "match": {
                                    "isSystem": parameters['isSystem']
                                }
                            },
                        ]
                    }
                },
                'from': from_range,
                'size': parameters['perPage'],
```

```
                'sort': [
                    Tosort
                ]
            }

            category_search_result = es.search(index='presentations', body=body)
            for i in category_search_result['hits']['hits']:
                if i['_id'] not in res['_id']:  # 중복제거
                    # Presentation Information
                    i['_source']['_id'] = i['_id']
                    res['_id'].append(i['_id'])
                    res['Info'].append(i['_source'])

                    # Calculate Presentation Page
                    pages['current'] = parameters['page']
                    if parameters['perPage'] == 1:
                        pages['prev'] = 0
                        parameters['hasPrev'] = False
                    else:
                        pages['prev'] = parameters['page'] - 1
                        parameters['hasPrev'] = True

                    if category_search_result['hits']['total']['value'] / (
                            parameters['perPage'] * parameters['page']) != 0:
                        pages['next'] = pages['current'] + 1
                        pages['hasNext'] = True
                    else:
                        pages['next'] = pages['current']
                        pages['hasNext'] = False

                    if round(category_search_result['hits']['total']['value'] // parameters['perPage']) == 0:
                        pages['total'] = 1
                    else:
                        pages['total'] = round(category_search_result['hits']['total']['value'] // parameters['p

                    # Calculate Presentation number
                    items['begin'] = 1
                    items['end'] = category_search_result['hits']['total']['value']
                    items['total'] = category_search_result['hits']['total']['value']

            result['data'] = res['Info']
            result['pages'] = pages
            result['items'] = items

            return jsonify(result)
```

- **case 5: 정렬만 선택하여 검색할 경우**

이미 나온 결과를 정렬을 선택하여 정렬할 code

```
        # setting only sort
        if parameters['sort'] and (not parameters['main']) and (not parameters['price']) and (not parameters['mo
            body = {
                'bool': {
                    'must': [
                        {
```

```
                            'query': {
                                'match_all': {}
                            },
                        }
                    ],
                    'filter': [
                        {
                            "match": {
                                "isSystem": parameters['isSystem']
                            }
                        },
                    ]
                },
                'from': from_range,
                'size': parameters['perPage'],
                'sort': [
                    Tosort
                ]
            }
            category_search_result = es.search(index='presentations', body=body)
            for i in category_search_result['hits']['hits']:
                if i['_id'] not in res['_id']:  # 중복제거
                    # Presentation Information
                    i['_source']['_id'] = i['_id']
                    res['_id'].append(i['_id'])
                    res['Info'].append(i['_source'])

                    # Calculate Presentation Page
                    pages['current'] = parameters['page']
                    if parameters['perPage'] == 1:
                        pages['prev'] = 0
                        parameters['hasPrev'] = False
                    else:
                        pages['prev'] = parameters['page'] - 1
                        parameters['hasPrev'] = True

                    if category_search_result['hits']['total']['value'] / (
                            parameters['perPage'] * parameters['page']) != 0:
                        pages['next'] = pages['current'] + 1
                        pages['hasNext'] = True
                    else:
                        pages['next'] = pages['current']
                        pages['hasNext'] = False

                    if round(category_search_result['hits']['total']['value'] // parameters['perPage']) == 0:
                        pages['total'] = 1
                    else:
                        pages['total'] = round(category_search_result['hits']['total']['value'] // parameters['p

                    # Calculate Presentation number
                    items['begin'] = 1
                    items['end'] = category_search_result['hits']['total']['value']
                    items['total'] = category_search_result['hits']['total']['value']

            result['data'] = res['Info']
            result['pages'] = pages
            result['items'] = items
```

```
64
65          return jsonify(result)
```

- **case 6: 스타일만 선택하여 검색할 경우**

스타일만 선택하여 검색을 진행했을 경우 code

```
1       # only select styles
2       if parameters['styles'] and (not parameters['main']) and (not parameters['price']) and (not parameters['
3           body = {
4               'query': {
5                   'bool': {
6                       'must': [
7                           {
8                               'match': {
9                                   'styles.styleName': parameters['styles'],
10                              }
11                          }
12                      ],
13                      'filter': [
14                          {
15                              "match": {
16                                  "isSystem": parameters['isSystem']
17                              }
18                          },
19                      ]
20                  }
21              },
22              'from': from_range,
23              'size': parameters['perPage'],
24              'sort': [
25                  Tosort
26              ]
27          }
28          category_search_result = es.search(index='presentations', body=body)
29          for i in category_search_result['hits']['hits']:
30              if i['_id'] not in res['_id']:  # 중복제거
31                  # Presentation Information
32                  i['_source']['_id'] = i['_id']
33                  res['_id'].append(i['_id'])
34                  res['Info'].append(i['_source'])
35
36                  # Calculate Presentation Page
37                  pages['current'] = parameters['page']
38                  if parameters['perPage'] == 1:
39                      pages['prev'] = 0
40                      parameters['hasPrev'] = False
41                  else:
42                      pages['prev'] = parameters['page'] - 1
43                      parameters['hasPrev'] = True
44
45                  if category_search_result['hits']['total']['value'] / (
46                          parameters['perPage'] * parameters['page']) != 0:
47                      pages['next'] = pages['current'] + 1
48                      pages['hasNext'] = True
49                  else:
```

```
                    pages['next'] = pages['current']
                    pages['hasNext'] = False

                if round(category_search_result['hits']['total']['value'] // parameters['perPage']) == 0:
                    pages['total'] = 1
                else:
                    pages['total'] = round(category_search_result['hits']['total']['value'] // parameters['p

                # Calculate Presentation number
                items['begin'] = 1
                items['end'] = category_search_result['hits']['total']['value']
                items['total'] = category_search_result['hits']['total']['value']

            result['data'] = res['Info']
            result['pages'] = pages
            result['items'] = items

            return jsonify(result)
```

- **case 7: 카테고리, 분위기 등 여러개를 선택하여 검색을 진행할 경우**

여러개의 카테고리를 선택하여 검색을 진행했을 경우 code(현재는 모두 이것을 사용)

```
        # only select category
        if parameters['main']:
            if parameters['middle']:
                category_body = {
                    'match': { # category middle설정
                        'categoryInfo.middle.categoryName': {
                            'query': parameters['middle']
                        }
                    }
                }
            else:
                category_body = {
                    'match': { # category main설정
                        'categoryInfo.main.categoryName': {
                            'query': parameters['main']
                        }
                    }
                }
        else:
            category_body = {
                'match_all': {}
            }

        # moods
        if parameters['moods']:
            moods_body = {
                'match': {
                    'moods.moodName': { # mood 설정
                        'query': parameters['moods']
                    }
                }
            }
        else:
```

```
            moods_body = {
                'match_all': {}
            }

        # styles
        if parameters['styles']:
            styles_body = {
                'match': {
                    'styles.styleName': { # styles 설정
                        'query': parameters['styles']
                    }
                }
            }
        else:
            styles_body = {
                'match_all': {}
            }

        # category main query body
        m_category_search = {
            'query': {
                'bool': {
                    'must': [ # 해당하는 범위를 가지고 search
                        category_body,
                        moods_body,
                        styles_body
                    ],
                    'filter': [
                        {
                            'range': {
                                'price': price_range
                            },
                        },
                        # {
                        #     'range': {
                        #         'likes': likes_range
                        #     }
                        # },
                        {
                            "match": {
                                "isSystem": parameters['isSystem']
                            }
                        },
                    ]
                }
            },
            'from': from_range,
            'size': parameters['perPage'],
            'sort': [
                Tosort
            ]
        }
    except: # 오류가 발생했을 경우, 모든 프레젠테이션 검색
        m_category_search = {
            'query': {
                'bool': {
                    'must': {
                        'match_all': {},
```

```
                    },
                    'filter': [
                        {
                            "match": {
                                "isSystem": parameters['isSystem']
                            }
                        },
                    ]
                }
            },
            'from': from_range,
            'size': parameters['perPage'],
            'sort': [
                Tosort
            ]
        }

    category_search_result = es.search(index='presentations', body=m_category_search)

    for i in category_search_result['hits']['hits']:
        if i['_id'] not in res['_id']:  #
            # Presentation Information
            i['_source']['_id'] = i['_id']
            res['_id'].append(i['_id'])
            res['Info'].append(i['_source'])

            # Calculate Presentation Page
            pages['current'] = parameters['page']
            if parameters['perPage'] == 1:
                pages['prev'] = 0
                parameters['hasPrev'] = False
            else:
                pages['prev'] = parameters['page'] - 1
                parameters['hasPrev'] = True

            if category_search_result['hits']['total']['value'] / (parameters['perPage'] * parameters['page'])
                pages['next'] = pages['current'] + 1
                pages['hasNext'] = True
            else:
                pages['next'] = pages['current']
                pages['hasNext'] = False

            if round(category_search_result['hits']['total']['value'] // parameters['perPage']) == 0:
                pages['total'] = 1
            else:
                pages['total'] = round(category_search_result['hits']['total']['value'] // parameters['perPage'

            # Calculate Presentation number
            items['begin'] = 1
            items['end'] = category_search_result['hits']['total']['value']
            items['total'] = category_search_result['hits']['total']['value']

    result['data'] = res['Info']
    result['pages'] = pages
    result['items'] = items

    return jsonify(result)
```