**Answer 1:**

```
GET product/_search
{
 "query": {
   "range": {
      "sold": {
        "lte": 10
      }
    }
```

**Answer 2:**

```
GET product/_search
{
  "query": {
    "range": {
      "sold": {
        "gte": 10,
        "lte": 30
      }
   }
  }
}
```

**Answer 3:**

```
GET product/_search
{
 "query":{
   "match":{
     "tags":"meat"
   }
 }
}
```

**Answer 4:**

```
GET /product/_search{
 "query":{
 "match": {
   "name":{
     "query": "Tomato Paste",
      "lenient": "true",
      "operator": "and"
    }
  }
}

}
```

**Answer 5:**

```
GET /product/_search
{
  "query":{
    "match_phrase_prefix": {
      "name":{
        "query":"pasta",
        "max_expansions":1
      }
    }
  }
}
```

**Answer 6:**

```
GET /product/_search
{
  "query": {
    "bool": {
     "must": {
       "exists":{
         "field":"tags"
       }
     }
    }
  }
}
```

**Answer 7:**

```
GET /shakespeare/_search
{
  "query":{
    "terms_set":{
      "text_entry":{
        "terms":
          ["men", "holy", "fight", "prince", "blessed", "conquest", "war", "knife"],
          "minimum_should_match_script":{
            "source":"params.num_terms/4"
          }

      }
    }
  }
}
```

**Answer 8:**

```
            GET /shakespeare/_search
            {
            "query":{
                "match_phrase": {
                         "text_entry": "some night-tripping fairy"
                         }
            }
}
```

**Answer 9:**

```
GET /shakespeare/_search
{
    "query": {
        "match": {
            "text_entry": {
                "query": "sape of likehood",
                "fuzziness": 2,
                "operator":"and"
            }
        }
    }
}
```

**Answer 10:**

```
GET /shakespeare/_search
{
  "query": {
    "multi_match": {
      "query": "Henry IV",
      "fields": ["play_name","speaker"]
      , "operator": "and"
    }
  }
}
```

**Answer 11:**

```
GET /shakespeare/_search
{
"query": {
 "match": {
  "text_entry": {
  "query": "my lady",
  "cutoff_frequency": 0.001

 }
}
}
}
```

**Answer 12:**

- That is currently active, has in_stock of at least 10 and has either wine or meat or both in tags.
- Filter the documents having price of at least 150.

- May or may not have 300 or more items sold.

```
GET product/_search
{
 "query": {
   "bool": {
    "must": [{
     "term": {
      "is_active": {
        "value": true
         }
        }
      }, {
        "range": {
          "in_stock": {
            "gte": 10
          }
        }
      }, {
        "query_string": {
          "default_field": "tags",
          "query": "(wine) or (meat)"
        }
      }],
      "filter": {
        "range": {
          "price": {
            "gte": 150
          }
        }
      },
      "should": [{
        "range": {
          "sold": {
            "gte": 300
          }
        }
      }]
    }
  }
}
```

**Answer 13:**

1. **Answer**

   - batch (integer type): example values, 2017, 2018
   - students (nested type, i.e. array of inner objects): each inner object can have three properties **id**, **name** and **age**.

```
PUT college2
{
  "mappings": {
    "_doc": {
      "dynamic": "false",
        "properties": {
          "batch": {
            "type": "integer"
            },
            "students": {
              "type": "nested",
              "properties": {
                "id": {
                  "type": "integer"
                },
                "name": {
                  "type": "text"
                },
                "age": {
                  "type": "integer"
                }
              }
            }
          }
        }
    }
}
```

2. **Answer**

```
        PUT college2/_doc/1
{
    "batch": 2019,
    "students": [
        {
          "id": 526,
          "name": "Sabina Shrestha",
          "age": 21
        },
        {
          "id": 503,
          "name": "Arika Joshi",
          "age": 21
        },
        {
          "id": 516,
          "name": "Nabina Khadka",
          "age": 21
        }
    ]
}
```

### 3. Answer

```
GET college2/_search
{
  "query": {
    "nested": {
      "path": "students",
      "ignore_unmapped": true,
      "score_mode": "sum",
      "inner_hits": {},
      "query": {
        "bool": {
          "should": [
            {
              "range": {
                "students.age": {
                  "gte": 10

                }
```

```
                   }
                  }
                 ]
                }
               }
              }
             }
            }
```

**Answer 14:**

```
POST /_aliases
{
 "actions" : [
  {
   "add" : {
    "indices":["product"],
     "alias" : "active_products",
     "filter" : {
      "term" : {
       "is_active": "true"
       }
       }
      }
    }
  ]
}
```