

## DBMS\_DEVELOPER

The package `DBMS_DEVELOPER` provides simple and developer friendly methods to retrieve information about database objects.

This chapter contains the following topics:

- [DBMS\\_DEVELOPER Overview](#)
- [Summary of DBMS\\_DEVELOPER Subprograms](#)

### DBMS\_DEVELOPER Overview

The package `DBMS_DEVELOPER` provides subprograms to retrieve information about database objects.

The `DBMS_DEVELOPER` package provides simple methods to retrieve relevant information about the database objects. This package is useful for developer tools and developers who would like to get information about the objects in the database, their shape, their structure, etc.

`DBMS_DEVELOPER` has the following advantages:

- **Performance** : Information is retrieved within a few milliseconds.
- **Simplicity** : `DBMS_DEVELOPER` moves away from XML and utilizes JSON, which is simple and comprehensive. This package provides methods to get information on any object where a non-privileged user has read permissions.

### DBMS\_DEVELOPER Security

This section lists the security restrictions related to `DBMS_DEVELOPER` package.

The function is executed with invoker rights. You will be allowed to access metadata for a database object **ONLY** if you are authorized to do a "describe" on the object. If a you do not have such authorization, an error will be raised and no metadata will be retrieved.

Here's a summary of privileges required:

- **Tables and Views**: You need **SELECT** or **READ** privileges on the table or view.
- **Indexes**: You need **SELECT** or **READ** privileges on the table on which the index is defined.
- **Synonyms**: You need **SELECT** or **READ** privileges on the synonym.

If you're describing objects in your own schema, no additional privilege is needed. However, if you're describing objects in another schema, you'll need at least the **SELECT** privilege on those objects.

If you are a DBA or have the **SELECT ANY TABLE**, **READ ANY TABLE**, and **SELECT ANY VIEW** system privilege, you can call `DBMS_DEVELOPER.GET_METADATA` on any table, index, or view in the database without needing explicit grants.

## Summary of DBMS\_DEVELOPER Subprograms

The table lists the DBMS\_DEVELOPER subprograms and briefly describes them.

The following table specifies the subprograms for DBMS\_DEVELOPER package and its brief description.

Subprogram	Description
GET_METADATA	The function takes information about the object including : name, object_type, schema, etag and produces a JSON document containing the metadata for the corresponding object. The level of information that needs to be produced in the output can be specified in the input parameter - level .

### GET\_METADATA Function

This section describes the syntax, input and output formats of the GET\_METADATA function.

This function takes information about the object and produces a JSON document containing the corresponding metadata for that object.

#### Syntax

```
DBMS_DEVELOPER.GET_METADATA (
name           IN VARCHAR2,
schema         IN VARCHAR2 DEFAULT NULL,
object_type    IN VARCHAR2 DEFAULT NULL,
level          IN VARCHAR2 DEFAULT 'TYPICAL'
etag           IN RAW        DEFAULT NULL)
RETURN JSON;
```

**Table 76-1 Input Configuration Fields**

Field	Value
name	The object name. A synonym of the object name can also be provided. It is case sensitive and should be provided as it appears in the data dictionary.
schema	The schema name. The default is the current user. It is case sensitive and should be provided as it appears in the data dictionary.
object_type	The type of object that you want to be retrieved. This is optional as name resolution can also be done without specifying object_type. Supported values: TABLE, INDEX, and VIEW

**Table 76-1 (Cont.) Input Configuration Fields**

Field	Value
level	<p>The level of detail. The default is <code>TYPICAL</code>.</p> <p>Supported values:</p> <ol style="list-style-type: none"> <li><b>1.</b> <code>BASIC</code> : Describes essential information.</li> <li><b>2.</b> <code>TYPICAL</code> : Includes more information of specified objects.</li> <li><b>3.</b> <code>ALL</code> : Provides the most comprehensive level of information.</li> </ol>
etag	<p>A unique identifier for a specific version of the document. This <code>etag</code> value lets an application determine whether the content of a particular version of a document is the same as that of another version.</p> <ol style="list-style-type: none"> <li><b>1.</b> If an <code>etag</code> is provided and it matches with current <code>etag</code>, the function returns an empty document.</li> <li><b>2.</b> If the <code>etag</code> of the document in the database does not match the <code>etag</code> that was provided, a new document is returned (along with the new embedded <code>etag</code>).</li> </ol>

`GET_METADATA` function will not return a JSON schema, instead, it will return a JSON document. Depending on the level of metadata you would like to retrieve, you can use the appropriate parameters for `GET_METADATA` function to obtain the desired metadata JSON document. Refer to [JSON Schema for Different Object and Sub-Object Types](#) to understand which schema you would need for your application.

### Examples

This section provides example usage for `DBMS_DEVELOPER.GET_METADATA` for different values of `object_type` and `level`.

- 1.** Retrieving metadata for `level` set to `BASIC` and for `object_type` set to `TABLE` :

```
SQL> conn hr/hr
Connected.
SQL>
SQL> select DBMS_DEVELOPER.GET_METADATA(object_type => 'TABLE', name =>
'EMPLOYEES', level => 'BASIC');
```

Executing this query would produce the following JSON document:

```
{
  "etag": "113059749145936325402354257176981405696",
  "objectType": "TABLE",
  "objectInfo": {
    "schema": "HR",
    "name": "EMPLOYEES",
    "columns": [
```

```
{
  "name": "EMPLOYEE_ID",
  "dataType": {
    "type": "NUMBER",
    "precision": 5,
    "scale": 0
  },
  "notNull": true
},
{
  "name": "FIRST_NAME",
  "dataType": {
    "type": "VARCHAR2",
    "size": 20,
    "sizeUnits": "BYTE"
  },
  "notNull": false
},
{
  "name": "LAST_NAME",
  "dataType": {
    "type": "VARCHAR2",
    "size": 25,
    "sizeUnits": "BYTE"
  },
  "notNull": true
},
{
  "name": "EMAIL",
  "dataType": {
    "type": "VARCHAR2",
    "size": 25,
    "sizeUnits": "BYTE"
  },
  "notNull": true
},
{
  "name": "PHONE_NUMBER",
  "dataType": {
    "type": "VARCHAR2",
    "size": 20,
    "sizeUnits": "BYTE"
  },
  "notNull": false
},
{
  "name": "HIRE_DATE",
  "dataType": "DATE",
  "notNull": true
},
{
  "name": "JOB_ID",
  "dataType": {
    "type": "VARCHAR2",
    "size": 10,
    "sizeUnits": "BYTE"
  }
}
```

```

        },
        "notNull": true
    },
    {
        "name": "SALARY",
        "dataType": {
            "type": "NUMBER",
            "precision": 8,
            "scale": 2
        },
        "notNull": false
    },
    {
        "name": "COMMISSION_PCT",
        "dataType": {
            "type": "NUMBER",
            "precision": 2,
            "scale": 0
        },
        "notNull": false
    },
    {
        "name": "MANAGER_ID",
        "dataType": {
            "type": "NUMBER",
            "precision": 6,
            "scale": 0
        },
        "notNull": false
    },
    {
        "name": "DEPARTMENT_ID",
        "dataType": {
            "type": "NUMBER",
            "precision": 4,
            "scale": 0
        },
        "notNull": false
    }
]
}

```

## 2. Retrieving metadata for level set to TYPICAL and for object\_type set to VIEW:

```

SQL> conn hr/hr
Connected.
SQL>
SQL> select DBMS_DEVELOPER.GET_METADATA(object_type => 'VIEW', name =>
'EMP_DETAILS_VIEW', level => 'TYPICAL');

```

Executing this query would produce the following JSON document:

```

{
    "etag": "113059749145936325402354257176981405696",

```

```

"objectType": "VIEW",
"objectInfo": {
  "name": "EMP_DETAILS_VIEW",
  "schema": "HR",
  "readOnly": true,
  "dualityView": false
  "columns": [
    {
      "name": "EMPLOYEE_ID",
      "isPk": true,
      "dataType": {
        "type": "NUMBER",
        "precision": 5,
        "scale": 0
      },
      "notNull": true
    },
    {
      "name": "JOB_ID",
      "dataType": {
        "type": "VARCHAR2",
        "size": 10,
        "sizeUnits": "BYTE"
      },
      "notNull": true
    },
    {
      "name": "MANAGER_ID",
      "dataType": {
        "type": "NUMBER",
        "precision": 6,
        "scale": 0
      },
      "notNull": false
    },
    {
      "name": "DEPARTMENT_ID",
      "dataType": {
        "type": "NUMBER",
        "precision": 4,
        "scale": 0
      },
      "notNull": false
    },
    {
      "name": "LOCATION_ID",
      "dataType": {
        "type": "NUMBER",
        "precision": 4,
        "scale": 0
      },
      "notNull": false
    },
    {
      "name": "COUNTRY_ID",
      "dataType": {

```

```

        "type": "VARCHAR2",
        "size": 2,
        "sizeUnits": "BYTE"
    },
    "notNull": true
},
{
    "name": "FIRST_NAME",
    "dataType": {
        "type": "VARCHAR2",
        "size": 20,
        "sizeUnits": "BYTE"
    },
    "notNull": false
},
{
    "name": "LAST_NAME",
    "dataType": {
        "type": "VARCHAR2",
        "size": 25,
        "sizeUnits": "BYTE"
    },
    "notNull": true,
    "comments": "Last name."
},
{
    "name": "SALARY",
    "dataType": {
        "type": "NUMBER",
        "precision": 8,
        "scale": 2
    },
    "notNull": false
},
{
    "name": "COMISSION_PCT",
    "dataType": {
        "type": "NUMBER",
        "precision": 2,
        "scale": 0
    },
    "notNull": false
},
{
    "name": "DEPARTMENT_NAME",
    "dataType": {
        "type": "VARCHAR2",
        "size": 30,
        "sizeUnits": "BYTE"
    },
    "notNull": true
},
{
    "name": "JOB_TITLE",
    "dataType": {
        "type": "VARCHAR2",

```

```

        "size": 35,
        "sizeUnits": "BYTE"
    },
    "notNull": true
},
{
    "name": "CITY",
    "dataType": {
        "type": "VARCHAR2",
        "size": 30,
        "sizeUnits": "BYTE"
    },
    "notNull": true
},
{
    "name": "STATE_PROVINCE",
    "dataType": {
        "type": "VARCHAR2",
        "size": 25,
        "sizeUnits": "BYTE"
    },
    "notNull": false
},
{
    "name": "COUNTRY_NAME",
    "dataType": {
        "type": "VARCHAR2",
        "size": 40,
        "sizeUnits": "BYTE"
    },
    "notNull": false
},
{
    "name": "REGION_NAME",
    "dataType": {
        "type": "VARCHAR2",
        "size": 25,
        "sizeUnits": "BYTE"
    },
    "notNull": false
}
],
"constraints": [
    {
        "name" : "SYS_C008649",
        "constraintType" : "VIEW READONLY",
        "status" : "ENABLE",
        "deferrable" : false,
        "validated" : "NON VALIDATED",
        "sysGeneratedName" : true
    }
],
"annotations": [
    {
        "Category": "EmpdetailsInfo"
    }
],

```



```

        {
            "Display": "Normal"
        }
    ]
}
}

```

### 3. Retrieving metadata for level set to ALL and for object\_type set to VIEW:

```

SQL> conn hr/hr
Connected.
SQL>
SQL> select DBMS_DEVELOPER.GET_METADATA(object_type => 'VIEW', name =>
'EMP_DETAILS_VIEW', level => 'ALL');

```

```

{
  "etag": "113059749145936325402354257176981405696",
  "objectType": "VIEW",
  "objectInfo": {
    "name": "EMP_DETAILS_VIEW",
    "schema": "HR",
    "readOnly": true,
    "dualityView": false,
    "editioningView" : false,
    "columns": [
      {
        "name": "EMPLOYEE_ID",
        "isPk": true,
        "dataType": {
          "type": "NUMBER",
          "precision": 5,
          "scale": 0
        },
        "notNull": true
      },
      {
        "name": "JOB_ID",
        "dataType": {
          "type": "VARCHAR2",
          "size": 10,
          "sizeUnits": "BYTE"
        },
        "notNull": true
      },
      {
        "name": "MANAGER_ID",
        "dataType": {
          "type": "NUMBER",
          "precision": 6,
          "scale": 0
        },
        "notNull": false
      },
      {
        "name": "DEPARTMENT_ID",

```

```

        "dataType": {
            "type": "NUMBER",
            "precision": 4,
            "scale": 0
        },
        "notNull": false
    },
    {
        "name": "LOCATION_ID",
        "dataType": {
            "type": "NUMBER",
            "precision": 4,
            "scale": 0
        },
        "notNull": false
    },
    {
        "name": "COUNTRY_ID",
        "dataType": {
            "type": "VARCHAR2",
            "size": 2,
            "sizeUnits": "BYTE"
        },
        "notNull": true
    },
    {
        "name": "FIRST_NAME",
        "dataType": {
            "type": "VARCHAR2",
            "size": 20,
            "sizeUnits": "BYTE"
        },
        "not_null": false
    },
    {
        "name": "LAST_NAME",
        "dataType": {
            "type": "VARCHAR2",
            "size": 25,
            "sizeUnits": "BYTE"
        },
        "notNull": true
    },
    {
        "name": "SALARY",
        "dataType": {
            "type": "NUMBER",
            "precision": 8,
            "scale": 2
        },
        "notNull": false
    },
    {
        "name": "COMISSION_PCT",
        "dataType": {
            "type": "NUMBER",

```

```

        "precision": 2,
        "scale": 0
    },
    "notNull": false
},
{
    "name": "DEPARTMENT_NAME",
    "dataType": {
        "type": "VARCHAR2",
        "size": 30,
        "sizeUnits": "BYTE"
    },
    "notNull": true
},
{
    "name": "JOB_TITLE",
    "dataType": {
        "type": "VARCHAR2",
        "size": 35,
        "sizeUnits": "BYTE"
    },
    "notNull": true
},
{
    "name": "CITY",
    "dataType": {
        "type": "VARCHAR2",
        "size": 30,
        "sizeUnits": "BYTE"
    },
    "notNull": true
},
{
    "name": "STATE_PROVINCE",
    "dataType": {
        "type": "VARCHAR2",
        "size": 25,
        "sizeUnits": "BYTE"
    },
    "notNull": false
},
{
    "name": "COUNTRY_NAME",
    "dataType": {
        "type": "VARCHAR2",
        "size": 40,
        "sizeUnits": "BYTE"
    },
    "notNull": false
},
{
    "name": "REGION_NAME",
    "dataType": {
        "type": "VARCHAR2",
        "size": 25,
        "sizeUnits": "BYTE"
    }
}

```

```

        },
        "notNull": false
    }
],
"constraints": [
    {
        "name" : "SYS_C008649",
        "constraintType" : "VIEW READONLY",
        "status" : "ENABLE",
        "deferrable" : false,
        "validated" : "NON VALIDATED",
        "sysGeneratedName" : true
    }
],
"annotations": [
    {
        "Category": "EmpdetailsInfo"
    },
    {
        "Display": "Normal"
    }
]
}
}

```

#### 4. Retrieving metadata for level set to ALL and for object\_type set to INDEX :

```

SQL> conn hr/hr
Connected.
SQL>
SQL> select DBMS_DEVELOPER.GET_METADATA(object_type => 'INDEX', name =>
'EMP_EMP_ID_PK', level => 'ALL');

```

Executing this query would produce the following JSON document:

```

{
  "etag": "113059749145936325402354257176981405696",
  "objectType": "INDEX",
  "objectInfo": {
    "name": "EMP_EMP_ID_PK",
    "indexType": "NORMAL",
    "Owner": "HR",
    "tableName": "EMPLOYEES",
    "uniqueness": "UNIQUE",
    "status": "VALID",
    "lastAnalyzed": "2024-06-19T18:25:43.511Z",
    "numRows": 107,
    "distinctKeys": 107,
    "sampleSize": 107,
    "compression": "DISABLE",
    "visibility": "VISIBLE",
    "toBeDropped": false,
    "columns": [
      {
        "name": "EMPLOYEE_ID",

```

```

        "isPk": true,
        "isUk": true,
        "isFk": false,
        "dataType": {
            "type": "NUMBER",
            "precision": 5,
            "scale": 0
        },
        "numDistinct": 107,
        "lowValue": "100",
        "highValue": "206",
        "density": 0.00934579439252336,
        "avgColLength": 4
    },
    ],
    "annotations": [
        {
            "Category": "EMP_EMP_ID_PKinfo"
        },
        {
            "Display": "Normal"
        }
    ]
}

```

**5. Retrieving metadata where level and object\_type are not explicitly specified:**

```

SQL> conn hr/hr
Connected.
SQL>
SQL> select dbms_developer.get_metadata(name => 'EMPLOYEES');

```

Even though the object\_type is not specified, the name resolution happens under the hood. The name object\_type corresponding to the name => 'EMPLOYEES' is determined to be table. So the JSON document is produced at the default typical level for the table object\_type. The output JSON document corresponding to the above example is shown here :

```

{
  "objectType" : "TABLE",
  "objectInfo" :
  {
    "name" : "EMPLOYEES",
    "schema" : "HR",
    "columns" :
    [
      {
        "name" : "EMPLOYEE_ID",
        "notNull" : true,
        "dataType" :
        {
          "type" : "NUMBER",
          "precision" : 6
        }
      },

```

```

        "isPk" : true,
        "isUk" : true,
        "isFk" : false
    },
    {
        "name" : "FIRST_NAME",
        "notNull" : false,
        "dataType" :
        {
            "type" : "VARCHAR2",
            "length" : 20,
            "sizeUnits" : "BYTE"
        },
        "isPk" : false,
        "isUk" : false,
        "isFk" : false
    },
    {
        "name" : "LAST_NAME",
        "notNull" : true,
        "dataType" :
        {
            "type" : "VARCHAR2",
            "length" : 25,
            "sizeUnits" : "BYTE"
        },
        "isPk" : false,
        "isUk" : false,
        "isFk" : false
    },
    {
        "name" : "EMAIL",
        "notNull" : true,
        "dataType" :
        {
            "type" : "VARCHAR2",
            "length" : 25,
            "sizeUnits" : "BYTE"
        },
        "isPk" : false,
        "isUk" : true,
        "isFk" : false
    },
    {
        "name" : "PHONE_NUMBER",
        "notNull" : false,
        "dataType" :
        {
            "type" : "VARCHAR2",
            "length" : 20,
            "sizeUnits" : "BYTE"
        },
        "isPk" : false,
        "isUk" : false,
        "isFk" : false
    },
    {

```

```
{
  "name" : "HIRE_DATE",
  "notNull" : true,
  "dataType" :
  {
    "type" : "DATE"
  },
  "isPk" : false,
  "isUk" : false,
  "isFk" : false
},
{
  "name" : "JOB_ID",
  "notNull" : true,
  "dataType" :
  {
    "type" : "VARCHAR2",
    "length" : 10,
    "sizeUnits" : "BYTE"
  },
  "isPk" : false,
  "isUk" : false,
  "isFk" : true
},
{
  "name" : "SALARY",
  "notNull" : false,
  "dataType" :
  {
    "type" : "NUMBER",
    "precision" : 8,
    "scale" : 2
  },
  "isPk" : false,
  "isUk" : false,
  "isFk" : false
},
{
  "name" : "COMMISSION_PCT",
  "notNull" : false,
  "dataType" :
  {
    "type" : "NUMBER",
    "precision" : 2,
    "scale" : 2
  },
  "isPk" : false,
  "isUk" : false,
  "isFk" : false
},
{
  "name" : "MANAGER_ID",
  "notNull" : false,
  "dataType" :
  {
    "type" : "NUMBER",
```

```

        "precision" : 6
    },
    "isPk" : false,
    "isUk" : false,
    "isFk" : true
},
{
    "name" : "DEPARTMENT_ID",
    "notNull" : false,
    "dataType" :
    {
        "type" : "NUMBER",
        "precision" : 4
    },
    "isPk" : false,
    "isUk" : false,
    "isFk" : true
}
],
"lastAnalyzed" : "2025-01-06T12:57:19",
"numRows" : 107,
"sampleSize" : 107,
"avgRowLen" : 69,
"indexes" :
[
    {
        "name" : "EMP_EMAIL_UK",
        "indexType" : "NORMAL",
        "uniqueness" : "UNIQUE",
        "status" : "VALID",
        "lastAnalyzed" : "2025-01-06T12:57:19",
        "numRows" : 107,
        "sampleSize" : 107,
        "columns" :
        [
            {
                "name" : "EMAIL"
            }
        ]
    },
    {
        "name" : "EMP_EMP_ID_PK",
        "indexType" : "NORMAL",
        "uniqueness" : "UNIQUE",
        "status" : "VALID",
        "lastAnalyzed" : "2025-01-06T12:57:20",
        "numRows" : 107,
        "sampleSize" : 107,
        "columns" :
        [
            {
                "name" : "EMPLOYEE_ID"
            }
        ]
    }
],
{

```



```

    "name" : "EMP_DEPARTMENT_IX",
    "indexType" : "NORMAL",
    "uniqueness" : "NONUNIQUE",
    "status" : "VALID",
    "lastAnalyzed" : "2025-01-06T12:57:20",
    "numRows" : 106,
    "sampleSize" : 106,
    "columns" :
    [
      {
        "name" : "DEPARTMENT_ID"
      }
    ]
  },
  {
    "name" : "EMP_JOB_IX",
    "indexType" : "NORMAL",
    "uniqueness" : "NONUNIQUE",
    "status" : "VALID",
    "lastAnalyzed" : "2025-01-06T12:57:20",
    "numRows" : 107,
    "sampleSize" : 107,
    "columns" :
    [
      {
        "name" : "JOB_ID"
      }
    ]
  },
  {
    "name" : "EMP_MANAGER_IX",
    "indexType" : "NORMAL",
    "uniqueness" : "NONUNIQUE",
    "status" : "VALID",
    "lastAnalyzed" : "2025-01-06T12:57:20",
    "numRows" : 106,
    "sampleSize" : 106,
    "columns" :
    [
      {
        "name" : "MANAGER_ID"
      }
    ]
  },
  {
    "name" : "EMP_NAME_IX",
    "indexType" : "NORMAL",
    "uniqueness" : "NONUNIQUE",
    "status" : "VALID",
    "lastAnalyzed" : "2025-01-06T12:57:20",
    "numRows" : 107,
    "sampleSize" : 107,
    "columns" :
    [
      {
        "name" : "LAST_NAME"
      }
    ]
  }
}

```

```

        },
        {
            "name" : "FIRST_NAME"
        }
    ]
}
],
"constraints" :
[
    {
        "name" : "EMP_LAST_NAME_NN",
        "constraintType" : "CHECK - NOT NULL",
        "searchCondition" : "\"LAST_NAME\" IS NOT NULL",
        "columns" :
        [
            {
                "name" : "LAST_NAME"
            }
        ],
        "status" : "ENABLE",
        "deferrable" : false,
        "validated" : "VALIDATED",
        "sysGeneratedName" : false
    },
    {
        "name" : "EMP_EMAIL_NN",
        "constraintType" : "CHECK - NOT NULL",
        "searchCondition" : "\"EMAIL\" IS NOT NULL",
        "columns" :
        [
            {
                "name" : "EMAIL"
            }
        ],
        "status" : "ENABLE",
        "deferrable" : false,
        "validated" : "VALIDATED",
        "sysGeneratedName" : false
    },
    {
        "name" : "EMP_HIRE_DATE_NN",
        "constraintType" : "CHECK - NOT NULL",
        "searchCondition" : "\"HIRE_DATE\" IS NOT NULL",
        "columns" :
        [
            {
                "name" : "HIRE_DATE"
            }
        ],
        "status" : "ENABLE",
        "deferrable" : false,
        "validated" : "VALIDATED",
        "sysGeneratedName" : false
    },
    {
        "name" : "EMP_JOB_NN",

```

```

"constraintType" : "CHECK - NOT NULL",
"searchCondition" : "\"JOB_ID\" IS NOT NULL",
"columns" :
[
  {
    "name" : "JOB_ID"
  }
],
"status" : "ENABLE",
"deferrable" : false,
"validated" : "VALIDATED",
"sysGeneratedName" : false
},
{
  "name" : "EMP_SALARY_MIN",
  "constraintType" : "CHECK",
  "searchCondition" : "salary > 0",
  "columns" :
  [
    {
      "name" : "SALARY"
    }
  ],
  "status" : "ENABLE",
  "deferrable" : false,
  "validated" : "VALIDATED",
  "sysGeneratedName" : false
},
{
  "name" : "EMP_EMAIL_UK",
  "constraintType" : "UNIQUE",
  "columns" :
  [
    {
      "name" : "EMAIL"
    }
  ],
  "status" : "ENABLE",
  "deferrable" : false,
  "validated" : "VALIDATED",
  "sysGeneratedName" : false
},
{
  "name" : "EMP_EMP_ID_PK",
  "constraintType" : "PRIMARY KEY",
  "columns" :
  [
    {
      "name" : "EMPLOYEE_ID"
    }
  ],
  "status" : "ENABLE",
  "deferrable" : false,
  "validated" : "VALIDATED",
  "sysGeneratedName" : false
},

```

```
{
  "Action" : "RESTRICT",
  "referencedConstraintName" : "DEPT_ID_PK",
  "constraintType" : "REFERENTIAL INTEGRITY",
  "columns" :
  [
    {
      "name" : "DEPARTMENT_ID"
    }
  ],
  "referencedColumns" :
  [
    {
      "name" : "DEPARTMENT_ID"
    }
  ],
  "sysGeneratedName" : false,
  "deferrable" : false,
  "referencedOwner" : "HR",
  "referencedTable" : "DEPARTMENTS",
  "name" : "EMP_DEPT_FK",
  "status" : "ENABLE",
  "validated" : "VALIDATED"
},
{
  "Action" : "RESTRICT",
  "referencedConstraintName" : "JOB_ID_PK",
  "constraintType" : "REFERENTIAL INTEGRITY",
  "columns" :
  [
    {
      "name" : "JOB_ID"
    }
  ],
  "referencedColumns" :
  [
    {
      "name" : "JOB_ID"
    }
  ],
  "sysGeneratedName" : false,
  "deferrable" : false,
  "referencedOwner" : "HR",
  "referencedTable" : "JOBS",
  "name" : "EMP_JOB_FK",
  "status" : "ENABLE",
  "validated" : "VALIDATED"
},
{
  "Action" : "RESTRICT",
  "referencedConstraintName" : "EMP_EMP_ID_PK",
  "constraintType" : "REFERENTIAL INTEGRITY",
  "columns" :
  [
    {
      "name" : "MANAGER_ID"
```

```

    }
  ],
  "referencedColumns" :
  [
    {
      "name" : "EMPLOYEE_ID"
    }
  ],
  "sysGeneratedName" : false,
  "deferrable" : false,
  "referencedOwner" : "HR",
  "referencedTable" : "EMPLOYEES",
  "name" : "EMP_MANAGER_FK",
  "status" : "ENABLE",
  "validated" : "VALIDATED"
}
]
},
"etag" : "CBBE5B271EEBB4C0C97FCFB047C233DD"
}

```

6. Retrieving metadata when the specified etag matches the current etag would return an empty document :

```

SQL> conn hr/hr
Connected.
SQL>
SQL> select dbms_developer.get_metadata(name => 'EMPLOYEES', etag =>
'CBBE5B271EEBB4C0C97FCFB047C233DD');

```

Executing this query would produce an empty JSON document :

```

{
}

```

## JSON Schema for Different Object and Sub-Object Types

This section provides an overview to the different schema possible for various object types, sub-object types and levels.

The JSON Schema can be used as a reference by applications to parse the JSON document returned. If you are programming against a specific version of the database program, you use that specific version of the JSON schema for that database version, and that schema will always be the same. But, if you upgrade or downgrade to a different release, you will be using a different schema version and then you need to use that correct schema. If there is any change in the schema, the version number in the \$id field will be changed. Applications can rely on this number in case Oracle plans to revise the JSON schema in the future. Existing versions will remain unchanged, so applications will remain fully functional and unaffected by the updates.

- The detailed description of the JSON schema for the `object_type` TABLE at all levels is described in the section : [JSON Schema for Object Type : TABLE](#)
- The detailed description of the JSON schema for the `object_type` INDEX at all levels is described in the section : [JSON Schema for Object Type : INDEX](#)

- The detailed description of the JSON schema for the `object_type` VIEW at all levels is described in the section : [JSON Schema for Object Type : VIEW](#)
- The detailed description of the JSON schema for the sub-object type `COLUMNS` at all levels is described in the section : [JSON Schema for Sub-Object Type : columns](#)
- The detailed description of the JSON schema for the sub-object type `CONSTRAINTS` at all levels is described in the section : [JSON Schema for Sub-Object Type : constraints](#)

## JSON Schema for Object Type : `TABLE`

This section describes the JSON schema for the object type `TABLE` at all levels.

### JSON Schema for `TABLE` Object Type in Tabular Form

**Table 76-2 JSON schema for object type : `TABLE`**

FIELDS	LEVELS		
	BASIC	TYPICAL	ALL
name			
schema			
columns			
lastAnalyzed			
hasBeenAnalyzed			
numRows			
sampleSize			
avgRowLen			
indexes			
constraints			
annotations			
segmentCreated			
inMemory			
compression			



#### Note:

The fields of the sub-objects `columns` and `constraints` varies depending on the choice of the `level`. The detailed information of the fields present in the sub-object `columns` is described in the section : [JSON Schema for Sub-Object Type : columns](#). The detailed information of the fields present in the sub-object `constraints` is described in the section : [JSON Schema for Sub-Object Type : constraints](#).

### JSON Schema for `TABLE` object type at level `BASIC`

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
```

```

    "$id": "https://oracle.com/schema/23.7/DBMS_DEVELOPER.GET_METADATA/BASIC/
object_info/table",
    "title": "DBMS_DEVELOPER.GET_METADATA/BASIC/OBJECT_INFO/TABLE",
    "description": "Information for a table object",
    "type": "object",
    "properties": {
      "name": {
        "description": "Table name",
        "type": "string"
      },
      "schema": {
        "description": "Table schema",
        "type": "string"
      },
      "columns": {
        "description": "Table columns",
        "type": "array",
        "items": {
          "$ref": "../column"
        }
      }
    },
    "required": [
      "name",
      "schema",
      "columns"
    ]
  }
}

```

#### JSON Schema for TABLE object type at level TYPICAL

```

{
  "etag": "113059749145936325402354257176981405696",
  "objectType": "TABLE",
  "objectInfo": {
    "schema": "HR",
    "name": "EMPLOYEES",
    "lastAnalyzed": "2024-06-14T18:25:43.511Z",
    "numRows": 107,
    "sampleSize": 107,
    "columns": [
      {
        "name": "EMPLOYEE_ID",
        "isPk": true,
        "isUk": true,
        "isFk": false,
        "dataType": {
          "type": "NUMBER",
          "precision": 5,
          "scale": 0
        },
        "notNull": true
      },
      {
        "name": "FIRST_NAME",

```

```

        "isPk": false,
        "isUk": false,
        "isFk": false,
        "dataType": {
            "type": "VARCHAR2",
            "size": 20,
            "sizeUnits": "BYTE"
        },
        "notNull": false
    },
    {
        "name": "LAST_NAME",
        "isPk": false,
        "isUk": false,
        "isFk": false,
        "dataType": {
            "type": "VARCHAR2",
            "size": 25,
            "sizeUnits": "BYTE"
        },
        "notNull": true
    },
    {
        "name": "EMAIL",
        "isPk": false,
        "isUk": true,
        "isFk": false,
        "dataType": {
            "type": "VARCHAR2",
            "size": 25,
            "sizeUnits": "BYTE"
        },
        "notNull": true
    },
    {
        "name": "PHONE_NUMBER",
        "isPk": false,
        "isUk": false,
        "isFk": false,
        "dataType": {
            "type": "VARCHAR2",
            "size": 20,
            "sizeUnits": "BYTE"
        },
        "notNull": false
    },
    {
        "name": "HIRE_DATE",
        "isPk": false,
        "isUk": false,
        "isFk": false,
        "dataType": "DATE",
        "notNull": true
    },
    {
        "name": "JOB_ID",

```



```

        "isPk": false,
        "isUk": false,
        "isFk": true,
        "dataType": {
            "type": "VARCHAR2",
            "size": 10,
            "sizeUnits": "BYTE"
        },
        "notNull": true
    },
    {
        "name": "SALARY",
        "isPk": false,
        "isUk": false,
        "isFk": false,
        "dataType": {
            "type": "NUMBER",
            "precision": 8,
            "scale": 2
        },
        "notNull": false
    },
    {
        "name": "COMISSION_PCT",
        "isPk": false,
        "isUk": false,
        "isFk": false,
        "dataType": {
            "type": "NUMBER",
            "precision": 2,
            "scale": 0
        },
        "notNull": false
    },
    {
        "name": "MANAGER_ID",
        "isPk": false,
        "isUk": false,
        "isFk": true,
        "dataType": {
            "type": "NUMBER",
            "precision": 6,
            "scale": 0
        },
        "notNull": false
    },
    {
        "name": "DEPARTMENT_ID",
        "isPk": false,
        "isUk": false,
        "isFk": true,
        "dataType": {
            "type": "NUMBER",
            "precision": 4,
            "scale": 0
        },
    },

```

```

        "notNull": false
    }
],
"indexes": [
    {
        "name": "HR.EMP_JOB_IX",
        "uniqueness": "NONUNIQUE",
        "status": "VALID",
        "columns": [
            {
                "name": "JOB_ID"
            }
        ]
    },
    {
        "name": "HR.EMP_NAME_IX",
        "uniqueness": "NONUNIQUE",
        "status": "VALID",
        "columns": [
            {
                "name": "FIRST_NAME"
            },
            {
                "name": "LAST_NAME"
            }
        ]
    },
    {
        "name": "HR.EMP_EMAIL_UK",
        "uniqueness": "UNIQUE",
        "status": "VALID",
        "columns": [
            {
                "name": "EMAIL"
            }
        ]
    },
    {
        "name": "HR.EMP_EMP_ID_PK",
        "uniqueness": "UNIQUE",
        "status": "VALID",
        "columns": [
            {
                "name": "EMPLOYEE_ID"
            }
        ]
    },
    {
        "name": "HR.EMP_MANAGER_IX",
        "uniqueness": "NONUNIQUE",
        "status": "VALID",
        "columns": [
            {
                "name": "MANAGER_ID"
            }
        ]
    }
]

```

```

    },
    {
        "name": "HR.EMP_DEPARTMENT_IX",
        "uniqueness": "NONUNIQUE",
        "status": "VALID",
        "columns": [
            {
                "name": "DEPARTMENT_ID"
            }
        ]
    }
],
"constraints":
[
    {
        "name" : "EMP_LAST_NAME_NN",
        "constraintType" : "CHECK - NOT NULL",
        "searchCondition" : "\"LAST_NAME\" IS NOT NULL",
        "columns" :
        [
            {
                "name" : "LAST_NAME"
            }
        ],
        "status" : "ENABLE",
        "deferrable" : false,
        "validated" : "VALIDATED",
        "sysGeneratedName" : false
    },
    {
        "name" : "EMP_EMAIL_NN",
        "constraintType" : "CHECK - NOT NULL",
        "searchCondition" : "\"EMAIL\" IS NOT NULL",
        "columns" :
        [
            {
                "name" : "EMAIL"
            }
        ],
        "status" : "ENABLE",
        "deferrable" : false,
        "validated" : "VALIDATED",
        "sysGeneratedName" : false
    },
    {
        "name" : "EMP_HIRE_DATE_NN",
        "constraintType" : "CHECK - NOT NULL",
        "searchCondition" : "\"HIRE_DATE\" IS NOT NULL",
        "columns" :
        [
            {
                "name" : "HIRE_DATE"
            }
        ],
        "status" : "ENABLE",
        "deferrable" : false,

```

```

        "validated" : "VALIDATED",
        "sysGeneratedName" : false
    },
    {
        "name" : "EMP_JOB_NN",
        "constraintType" : "CHECK - NOT NULL",
        "searchCondition" : "\"JOB_ID\" IS NOT NULL",
        "columns" :
        [
            {
                "name" : "JOB_ID"
            }
        ],
        "status" : "ENABLE",
        "deferrable" : false,
        "validated" : "VALIDATED",
        "sysGeneratedName" : false
    },
    {
        "name" : "EMP_SALARY_MIN",
        "constraintType" : "CHECK",
        "searchCondition" : "salary > 0",
        "columns" :
        [
            {
                "name" : "SALARY"
            }
        ],
        "status" : "ENABLE",
        "deferrable" : false,
        "validated" : "VALIDATED",
        "sysGeneratedName" : false
    },
    {
        "name" : "EMP_EMAIL_UK",
        "constraintType" : "UNIQUE",
        "columns" :
        [
            {
                "name" : "EMAIL"
            }
        ],
        "status" : "ENABLE",
        "deferrable" : false,
        "validated" : "VALIDATED",
        "sysGeneratedName" : false
    },
    {
        "name" : "EMP_EMP_ID_PK",
        "constraintType" : "PRIMARY KEY",
        "columns" :
        [
            {
                "name" : "EMPLOYEE_ID"
            }
        ],

```

```

        "status" : "ENABLE",
        "deferrable" : false,
        "validated" : "VALIDATED",
        "sysGeneratedName" : false
    },
    {
        "Action" : "RESTRICT",
        "referencedConstraintName" : "DEPT_ID_PK",
        "constraintType" : "REFERENTIAL INTEGRITY",
        "columns" :
        [
            {
                "name" : "DEPARTMENT_ID"
            }
        ],
        "referencedColumns" :
        [
            {
                "name" : "DEPARTMENT_ID"
            }
        ],
        "sysGeneratedName" : false,
        "deferrable" : false,
        "referencedOwner" : "HR",
        "referencedTable" : "DEPARTMENTS",
        "name" : "EMP_DEPT_FK",
        "status" : "ENABLE",
        "validated" : "VALIDATED"
    },
    {
        "Action" : "RESTRICT",
        "referencedConstraintName" : "JOB_ID_PK",
        "constraintType" : "REFERENTIAL INTEGRITY",
        "columns" :
        [
            {
                "name" : "JOB_ID"
            }
        ],
        "referencedColumns" :
        [
            {
                "name" : "JOB_ID"
            }
        ],
        "sysGeneratedName" : false,
        "deferrable" : false,
        "referencedOwner" : "HR",
        "referencedTable" : "JOBS",
        "name" : "EMP_JOB_FK",
        "status" : "ENABLE",
        "validated" : "VALIDATED"
    },
    {
        "Action" : "RESTRICT",
        "referencedConstraintName" : "EMP_EMP_ID_PK",

```

```

        "constraintType" : "REFERENTIAL INTEGRITY",
        "columns" :
        [
            {
                "name" : "MANAGER_ID"
            }
        ],
        "referencedColumns" :
        [
            {
                "name" : "EMPLOYEE_ID"
            }
        ],
        "sysGeneratedName" : false,
        "deferrable" : false,
        "referencedOwner" : "HR",
        "referencedTable" : "EMPLOYEES",
        "name" : "EMP_MANAGER_FK",
        "status" : "ENABLE",
        "validated" : "VALIDATED"
    }
],
"annotations": [
    {
        "Category": "EmpInfo"
    },
    {
        "Display": "Normal"
    }
]
}
}

```

#### JSON Schema for TABLE object type at level ALL

```

{
    "etag": "98D8203DF79ECF4207BCCC91FEF81515",
    "objectType": "TABLE",
    "objectInfo": {
        "schema": "HR",
        "name": "EMPLOYEES",
        "lastAnalyzed": "2024-06-14T18:25:43.511Z",
        "numRows": 107,
        "sampleSize": 107,
        "compression": false,
        "avgRowLen": 69,
        "segmentCreated": "YES",
        "inMemory": "ENABLED",
        "columns": [
            {
                "name": "EMPLOYEE_ID",
                "isPk": true,
                "isUk": true,
                "isFk": false,
                "dataType": {

```

```

        "type": "NUMBER",
        "precision": 5,
        "scale": 0
    },
    "notNull": true,
    "numDistinct": 107,
    "lowValue": "100",
    "highValue": "206",
    "density": 0.00934579439252336,
    "avgCollLength": 4
},
{
    "name": "FIRST_NAME",
    "isPk": false,
    "isUk": false,
    "isFk": false,
    "dataType": {
        "type": "VARCHAR2",
        "size": 20,
        "sizeUnits": "BYTE"
    },
    "notNull": true,
    "numDistinct": 91,
    "lowValue": "Adam",
    "highValue": "Winston",
    "density": 0.010989010989011,
    "avgCollLength": 7
},
{
    "name": "LAST_NAME",
    "isPk": false,
    "isUk": false,
    "isFk": false,
    "dataType": {
        "type": "VARCHAR2",
        "size": 25,
        "sizeUnits": "BYTE"
    },
    "notNull": true,
    "numDistinct": 102,
    "lowValue": "Abel",
    "highValue": "Zlotkey",
    "density": 0.00980392156862745,
    "avgCollLength": 8
},
{
    "name": "EMAIL",
    "isPk": false,
    "isUk": true,
    "isFk": false,
    "dataType": {
        "type": "VARCHAR2",
        "size": 25,
        "sizeUnits": "BYTE"
    },
    "notNull": true,

```

```

        "numDistinct": 107,
        "lowValue": "ABANDA",
        "highValue": "WTAYLOR",
        "density": 0.00934579439252336,
        "avgCollLength": 8
    },
    {
        "name": "PHONE_NUMBER",
        "isPk": false,
        "isUk": false,
        "isFk": false,
        "dataType": {
            "type": "VARCHAR2",
            "size": 20,
            "sizeUnits": "BYTE"
        },
        "notNull": false,
        "numDistinct": 107,
        "lowValue": "011.44.1343.329268",
        "highValue": "650.509.4876",
        "density": 0.00934579439252336,
        "avgCollLength": 15
    },
    {
        "name": "HIRE_DATE",
        "isPk": false,
        "isUk": false,
        "isFk": false,
        "dataType": "DATE",
        "notNull": true,
        "numDistinct": 98,
        "lowValue": "2001.01.13.00.00.00",
        "highValue": "2008.04.21.00.00.00",
        "density": 0.0102040816326531,
        "avgCollLength": 8
    },
    {
        "name": "JOB_ID",
        "isPk": false,
        "isUk": false,
        "isFk": true,
        "dataType": {
            "type": "VARCHAR2",
            "size": 10,
            "sizeUnits": "BYTE"
        },
        "notNull": true,
        "numDistinct": 19,
        "lowValue": "AC_ACCOUNT",
        "highValue": "ST_MAN",
        "density": 0.00467289719626168,
        "avgCollLength": 9
    },
    {
        "name": "SALARY",
        "isPk": false,

```



```

        "isUk": false,
        "isFk": false,
        "dataType": {
            "type": "NUMBER",
            "precision": 8,
            "scale": 2
        },
        "notNull": false,
        "numDistinct": 58,
        "lowValue": "2100",
        "highValue": "24000",
        "density": 0.0172413793103448,
        "avgColLength": 4
    },
    {
        "name": "COMISSION_PCT",
        "isPk": false,
        "isUk": false,
        "isFk": false,
        "dataType": {
            "type": "NUMBER",
            "precision": 2,
            "scale": 0
        },
        "notNull": false,
        "numDistinct": 7,
        "lowValue": ".1",
        "highValue": ".4",
        "density": 0.142857142857143,
        "avgColLength": 2
    },
    {
        "name": "MANAGER_ID",
        "isPk": false,
        "isUk": false,
        "isFk": true,
        "dataType": {
            "type": "NUMBER",
            "precision": 6,
            "scale": 0
        },
        "notNull": false,
        "numDistinct": 18,
        "lowValue": "100",
        "highValue": "205",
        "density": 0.00471698113207547,
        "avgColLength": 4
    },
    {
        "name": "DEPARTMENT_ID",
        "isPk": false,
        "isUk": false,
        "isFk": true,
        "dataType": {
            "type": "NUMBER",
            "precision": 4,

```

```

        "scale": 0
    },
    "notNull": false,
    "numDistinct": 11,
    "lowValue": "10",
    "highValue": "110",
    "density": "0.00471698113207547",
    "avgColLength": 3
}
],
"indexes": [
{
    "name": "HR.EMP_JOB_IX",
    "indexType": "NORMAL",
    "uniqueness": "NONUNIQUE",
    "status": "VALID",
    "numRows": 107,
    "distinctKeys": 19,
    "lastAnalyzed": "2024-06-14T18:25:43.511Z",
    "compression": "DISABLED",
    "segmentCreated": "YES",
    "visibility": "VISIBLE",
    "columns": [
        {
            "name": "JOB_ID"
        }
    ]
},
{
    "name": "HR.EMP_NAME_IX",
    "indexType": "NORMAL",
    "uniqueness": "NONUNIQUE",
    "status": "VALID",
    "numRows": 107,
    "distinctKeys": 107,
    "lastAnalyzed": "2024-06-14T18:25:43.511Z",
    "compression": "DISABLED",
    "segmentCreated": "YES",
    "visibility": "VISIBLE",
    "columns": [
        {
            "name": "FIRST_NAME"
        },
        {
            "name": "LAST_NAME"
        }
    ]
},
{
    "name": "HR.EMP_EMAIL_UK",
    "indexType": "NORMAL",
    "uniqueness": "UNIQUE",
    "status": "VALID",
    "numRows": 107,
    "distinctKeys": 107,
    "lastAnalyzed": "2024-06-14T18:25:43.511Z",

```

```

        "compression": "DISABLED",
        "segmentCreated": "YES",
        "visibility": "VISIBLE",
        "columns": [
            {
                "name": "EMAIL"
            }
        ]
    },
    {
        "name": "HR.EMP_EMP_ID_PK",
        "indexType": "NORMAL",
        "uniqueness": "UNIQUE",
        "status": "VALID",
        "numRows": 107,
        "distinctKeys": 107,
        "lastAnalyzed": "2024-06-14T18:25:43.511Z",
        "compression": "DISABLED",
        "segmentCreated": "YES",
        "visibility": "VISIBLE",
        "columns": [
            {
                "name": "EMPLOYEE_ID"
            }
        ]
    },
    {
        "name": "HR.EMP_MANAGER_IX",
        "indexType": "NORMAL",
        "uniqueness": "NONUNIQUE",
        "status": "VALID",
        "numRows": 106,
        "distinctKeys": 18,
        "lastAnalyzed": "2024-06-14T18:25:43.511Z",
        "compression": "DISABLED",
        "segmentCreated": "YES",
        "visibility": "VISIBLE",
        "columns": [
            {
                "name": "MANAGER_ID"
            }
        ]
    },
    {
        "name": "HR.EMP_DEPARTMENT_IX",
        "indexType": "NORMAL",
        "uniqueness": "NONUNIQUE",
        "status": "VALID",
        "numRows": 106,
        "distinctKeys": 11,
        "lastAnalyzed": "2024-06-14T18:25:43.511Z",
        "compression": "DISABLED",
        "segmentCreated": "YES",
        "visibility": "VISIBLE",
        "columns": [
            {

```

```

        "name": "DEPARTMENT_ID"
    }
    ]
}
],
"constraints":
[
{
    "name" : "EMP_LAST_NAME_NN",
    "constraintType" : "CHECK - NOT NULL",
    "searchCondition" : "\"LAST_NAME\" IS NOT NULL",
    "columns" :
    [
        {
            "name" : "LAST_NAME"
        }
    ],
    "status" : "ENABLE",
    "deferrable" : false,
    "validated" : "VALIDATED",
    "sysGeneratedName" : false
},
{
    "name" : "EMP_EMAIL_NN",
    "constraintType" : "CHECK - NOT NULL",
    "searchCondition" : "\"EMAIL\" IS NOT NULL",
    "columns" :
    [
        {
            "name" : "EMAIL"
        }
    ],
    "status" : "ENABLE",
    "deferrable" : false,
    "validated" : "VALIDATED",
    "sysGeneratedName" : false
},
{
    "name" : "EMP_HIRE_DATE_NN",
    "constraintType" : "CHECK - NOT NULL",
    "searchCondition" : "\"HIRE_DATE\" IS NOT NULL",
    "columns" :
    [
        {
            "name" : "HIRE_DATE"
        }
    ],
    "status" : "ENABLE",
    "deferrable" : false,
    "validated" : "VALIDATED",
    "sysGeneratedName" : false
},
{
    "name" : "EMP_JOB_NN",
    "constraintType" : "CHECK - NOT NULL",
    "searchCondition" : "\"JOB_ID\" IS NOT NULL",

```

```

"columns" :
[
  {
    "name" : "JOB_ID"
  }
],
"status" : "ENABLE",
"deferrable" : false,
"validated" : "VALIDATED",
"sysGeneratedName" : false
},
{
  "name" : "EMP_SALARY_MIN",
  "constraintType" : "CHECK",
  "searchCondition" : "salary > 0",
  "columns" :
  [
    {
      "name" : "SALARY"
    }
  ],
  "status" : "ENABLE",
  "deferrable" : false,
  "validated" : "VALIDATED",
  "sysGeneratedName" : false
},
{
  "name" : "EMP_EMAIL_UK",
  "constraintType" : "UNIQUE",
  "columns" :
  [
    {
      "name" : "EMAIL"
    }
  ],
  "status" : "ENABLE",
  "deferrable" : false,
  "validated" : "VALIDATED",
  "sysGeneratedName" : false
},
{
  "name" : "EMP_EMP_ID_PK",
  "constraintType" : "PRIMARY KEY",
  "columns" :
  [
    {
      "name" : "EMPLOYEE_ID"
    }
  ],
  "status" : "ENABLE",
  "deferrable" : false,
  "validated" : "VALIDATED",
  "sysGeneratedName" : false
},
{
  "Action" : "RESTRICT",

```

```

"referencedConstraintName" : "DEPT_ID_PK",
"constraintType" : "REFERENTIAL INTEGRITY",
"columns" :
[
  {
    "name" : "DEPARTMENT_ID"
  }
],
"referencedColumns" :
[
  {
    "name" : "DEPARTMENT_ID"
  }
],
"sysGeneratedName" : false,
"deferrable" : false,
"referencedOwner" : "HR",
"referencedTable" : "DEPARTMENTS",
"name" : "EMP_DEPT_FK",
"status" : "ENABLE",
"validated" : "VALIDATED"
},
{
  "Action" : "RESTRICT",
  "referencedConstraintName" : "JOB_ID_PK",
  "constraintType" : "REFERENTIAL INTEGRITY",
  "columns" :
  [
    {
      "name" : "JOB_ID"
    }
  ],
  "referencedColumns" :
  [
    {
      "name" : "JOB_ID"
    }
  ],
  "sysGeneratedName" : false,
  "deferrable" : false,
  "referencedOwner" : "HR",
  "referencedTable" : "JOBS",
  "name" : "EMP_JOB_FK",
  "status" : "ENABLE",
  "validated" : "VALIDATED"
},
{
  "Action" : "RESTRICT",
  "referencedConstraintName" : "EMP_EMP_ID_PK",
  "constraintType" : "REFERENTIAL INTEGRITY",
  "columns" :
  [
    {
      "name" : "MANAGER_ID"
    }
  ],

```

```

        "referencedColumns" :
        [
          {
            "name" : "EMPLOYEE_ID"
          }
        ],
        "sysGeneratedName" : false,
        "deferrable" : false,
        "referencedOwner" : "HR",
        "referencedTable" : "EMPLOYEES",
        "name" : "EMP_MANAGER_FK",
        "status" : "ENABLE",
        "validated" : "VALIDATED"
      }
    ],
    "annotations": [
      {
        "Category": "EmpInfo"
      },
      {
        "Display": "Normal"
      }
    ]
  }
}

```

## JSON Schema for Object Type : VIEW

This section describes the JSON schema for the object type `VIEW` at all levels.

### JSON Schema for `VIEW` Object Type in Tabular Form

**Table 76-3 JSON Schema for `VIEW` Object Type**

Fields	LEVELS		
	BASIC	TYPICAL	ALL
name			
schema			
columns			
readOnly			
dualityView			
constraints			
annotations			
editioningView			

**Note:**

The fields of the sub-objects `columns` and `constraints` varies depending on the choice of the `level`. The detailed information of the fields present in the sub-object `columns` is described in the section : [JSON Schema for Sub-Object Type : columns](#). The detailed information of the fields present in the sub-object `constraints` is described in the section : [JSON Schema for Sub-Object Type : constraints](#).

**JSON Schema for VIEW object type at level BASIC**

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "$id": "https://oracle.com/schema/23.7/DBMS_DEVELOPER.GET_METADATA/BASIC/
object_info/view",
  "title": "DBMS_DEVELOPER.GET_METADATA/BASIC/OBJECT_INFO/VIEW",
  "description": "Information for a View object",
  "type": "object",
  "properties": {
    "name": {
      "description": "View name",
      "type": "string"
    },
    "schema": {
      "description": "View schema",
      "type": "string"
    },
    "columns": {
      "description": "View columns",
      "type": "array",
      "items": {
        "$ref": "../column"
      }
    },
    "required": [
      "name",
      "schema",
      "columns"
    ]
  }
}
```

**JSON Schema for VIEW object type at level TYPICAL**

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "$id": "https://oracle.com/schema/23.7/DBMS_DEVELOPER.GET_METADATA/
TYPICAL/object_info/view",
  "title": "DBMS_DEVELOPER.GET_METADATA/TYPICAL/OBJECT_INFO/VIEW",
  "description": "Information for a View object",
  "type": "object",
  "properties": {
    "name": {
      "description": "View name",
```



```

        "type": "string"
    },
    "schema": {
        "description": "View schema",
        "type": "string"
    },
    "readOnly": {
        "description": "An indicator of whether the view is a Read Only
View",
        "type": "boolean",
        "default": true
    },
    "dualityView": {
        "description": "Whether the view is a JSON Duality View or not",
        "type": "boolean",
        "default": false
    },
    "columns": {
        "description": "View columns",
        "type": "array",
        "items": {
            "$ref": "../column"
        }
    },
    "annotations": {
        "description": "View annotations",
        "type": "array",
        "items": {
            "$ref": "../annotation"
        }
    },
    "constraints": {
        "description": "View constraints",
        "type": "array",
        "items": {
            "$ref": "../constraint"
        }
    },
    "required": [
        "name",
        "schema",
        "readOnly",
        "dualityView",
        "columns"
    ]
}
}

```

#### JSON Schema for VIEW object type at level ALL

```

{
    "etag": "113059749145936325402354257176981405696",
    "objectType": "VIEW",
    "objectInfo": {
        "name": "EMP_DETAILS_VIEW",

```

```

"schema": "HR",
"readOnly": true,
"dualityView": false,
"editioningView" : false,
"columns": [
  {
    "name": "EMPLOYEE_ID",
    "isPk": true,
    "dataType": {
      "type": "NUMBER",
      "precision": 5,
      "scale": 0
    },
    "notNull": true
  },
  {
    "name": "JOB_ID",
    "dataType": {
      "type": "VARCHAR2",
      "size": 10,
      "sizeUnits": "BYTE"
    },
    "notNull": true
  },
  {
    "name": "MANAGER_ID",
    "dataType": {
      "type": "NUMBER",
      "precision": 6,
      "scale": 0
    },
    "notNull": false
  },
  {
    "name": "DEPARTMENT_ID",
    "dataType": {
      "type": "NUMBER",
      "precision": 4,
      "scale": 0
    },
    "notNull": false
  },
  {
    "name": "LOCATION_ID",
    "dataType": {
      "type": "NUMBER",
      "precision": 4,
      "scale": 0
    },
    "notNull": false
  },
  {
    "name": "COUNTRY_ID",
    "dataType": {
      "type": "VARCHAR2",
      "size": 2,

```

```

        "sizeUnits": "BYTE"
    },
    "notNull": true
},
{
    "name": "FIRST_NAME",
    "dataType": {
        "type": "VARCHAR2",
        "size": 20,
        "sizeUnits": "BYTE"
    },
    "not_null": false
},
{
    "name": "LAST_NAME",
    "dataType": {
        "type": "VARCHAR2",
        "size": 25,
        "sizeUnits": "BYTE"
    },
    "notNull": true
},
{
    "name": "SALARY",
    "dataType": {
        "type": "NUMBER",
        "precision": 8,
        "scale": 2
    },
    "notNull": false
},
{
    "name": "COMISSION_PCT",
    "dataType": {
        "type": "NUMBER",
        "precision": 2,
        "scale": 0
    },
    "notNull": false
},
{
    "name": "DEPARTMENT_NAME",
    "dataType": {
        "type": "VARCHAR2",
        "size": 30,
        "sizeUnits": "BYTE"
    },
    "notNull": true
},
{
    "name": "JOB_TITLE",
    "dataType": {
        "type": "VARCHAR2",
        "size": 35,
        "sizeUnits": "BYTE"
    },

```

```

        "notNull": true
    },
    {
        "name": "CITY",
        "dataType": {
            "type": "VARCHAR2",
            "size": 30,
            "sizeUnits": "BYTE"
        },
        "notNull": true
    },
    {
        "name": "STATE_PROVINCE",
        "dataType": {
            "type": "VARCHAR2",
            "size": 25,
            "sizeUnits": "BYTE"
        },
        "notNull": false
    },
    {
        "name": "COUNTRY_NAME",
        "dataType": {
            "type": "VARCHAR2",
            "size": 40,
            "sizeUnits": "BYTE"
        },
        "notNull": false
    },
    {
        "name": "REGION_NAME",
        "dataType": {
            "type": "VARCHAR2",
            "size": 25,
            "sizeUnits": "BYTE"
        },
        "notNull": false
    }
],
"constraints": [
    {
        "name" : "SYS_C008649",
        "constraintType" : "VIEW READONLY",
        "status" : "ENABLE",
        "deferrable" : false,
        "validated" : "NON VALIDATED",
        "sysGeneratedName" : true
    }
],
"annotations": [
    {
        "Category": "EmpdetailsInfo"
    },
    {
        "Display": "Normal"
    }
]

```

```

    ]
  }
}

```

## JSON Schema for Object Type : INDEX

This section describes the JSON schema for the object type `INDEX` at all levels.

### JSON Schema for INDEX Object Type in Tabular Form

**Table 76-4 JSON Schema for INDEX Object Type**

Fields	LEVELS		
	BASIC	TYPICAL	ALL
name			
indexType			
Owner			
tableName			
status			
columns			
uniqueness			
funcIdxStatus			
hasBeenAnalyz ed			
lastAnalyzed			
numRows			
sampleSize			
annotations			
distinctKeys			
compression			
segmentCreate d			
visiblity			
toBeDropped			



#### Note:

The fields of the sub-object `columns` varies depending on the choice of `level`. The detailed information of the fields present in the sub-object `columns` is described in the section : [JSON Schema for Sub-Object Type : columns](#).

### JSON Schema for INDEX object type at level BASIC

```

{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "$id": "https://oracle.com/schema/23.7/DBMS_DEVELOPER.GET_METADATA/BASIC/
object_info/index",
  "title": "DBMS_DEVELOPER.GET_METADATA/BASIC/OBJECT_INFO/INDEX",
  "description": "Information for index",

```

```

    "type": "object",
    "properties": {
      "name": {
        "description": "Index name",
        "type": "string"
      },
      "indexType": {
        "description": "Index Types i.e. CLUSTER, NORMAL, NORMAL/REV,
BITMAP, DOMAIN, LOB, IOT - TOP,
                                FUNCTION-BASED NORMAL, FUNCTION-BASED NORMAL/REV,
                                FUNCTION-BASED BITMAP, FUNCTION-BASED DOMAIN,
VECTOR",
        "type": "string"
      },
      "Owner": {
        "description": "Owner of the indexed object",
        "type": "string"
      },
      "tableName": {
        "description": "Name of the indexed object",
        "type": "string"
      },
      "status": {
        "description": "Status",
        "type": {
          "enum": [
            "VALID",
            "UNUSABLE",
            "INPROGRS",
            "N/A"
          ]
        }
      },
      "columns": {
        "description": "Index column(s)",
        "type": "array",
        "items": {
          "$ref": "../column"
        }
      },
      "required": [
        "name",
        "indexType",
        "Owner",
        "tableName",
        "status",
        "columns"
      ]
    }
  }
}

```

#### JSON Schema for INDEX object type at level TYPICAL

```

{
  "etag": "113059749145936325402354257176981405696",

```

```

"objectType": "INDEX",
"objectInfo": {
  "name": "EMP_EMP_ID_PK",
  "indexType": "NORMAL",
  "Owner": "HR",
  "tableName": "EMPLOYEES",
  "uniqueness": "UNIQUE",
  "status": "VALID",
  "lastAnalyzed": "2024-06-14T18:25:43.511Z",
  "numRows": 107,
  "sampleSize": 107,
  "columns": [
    {
      "name": "EMPLOYEE_ID",
      "isPk": true,
      "isUk": true,
      "isFk": false,
      "dataType": {
        "type": "NUMBER",
        "precision": 5,
        "scale": 0
      },
      "notNull": true
    }
  ],
  "annotations": [
    {
      "Category": "EMP_EMP_ID_PKinfo"
    },
    {
      "Display": "Normal"
    }
  ]
}
}

```

#### JSON Schema for INDEX object type at level ALL

```

{
  "etag": "113059749145936325402354257176981405696",
  "objectType": "INDEX",
  "objectInfo": {
    "name": "EMP_EMP_ID_PK",
    "indexType": "NORMAL",
    "Owner": "HR",
    "tableName": "EMPLOYEES",
    "uniqueness": "UNIQUE",
    "status": "VALID",
    "lastAnalyzed": "2024-06-19T18:25:43.511Z",
    "numRows": 107,
    "distinctKeys": 107,
    "sampleSize": 107,
    "compression": "DISABLE",
    "visibility": "VISIBLE",
    "toBeDropped": false,

```

```

    "columns": [
      {
        "name": "EMPLOYEE_ID",
        "isPk": true,
        "isUk": true,
        "isFk": false,
        "dataType": {
          "type": "NUMBER",
          "precision": 5,
          "scale": 0
        },
        "numDistinct": 107,
        "lowValue": "100",
        "highValue": "206",
        "density": 0.00934579439252336,
        "avgColLength": 4
      },
    ],
    "annotations": [
      {
        "Category": "EMP_EMP_ID_PKinfo"
      },
      {
        "Display": "Normal"
      }
    ]
  }
}

```

## JSON Schema for Sub-Object Type : columns

This section describes the JSON schema for the sub-object `columns` which belongs to all the object types.

### JSON Schema for `columns` Sub-Object Type in Tabular Form

**Table 76-5 JSON Schema for COLUMNS Sub-Object**

Fields	Levels		
	BASIC	TYPICAL	ALL
name			
default			
notNull			
dataType	type		
	precision		
	scale		
	length		
	sizeUnits		
	fractionalSecondsP		
	recision		
	yearPrecision		
	dayPrecision		



**Table 76-5 (Cont.) JSON Schema for COLUMNS Sub-Object**

Fields	Levels
	size
isPk	
isUk	
isFk	
domain	name
	type
	display
	order
	annotations
annotations	
numDistinct	
lowValue	
highValue	
density	
avgCollen	
hiddenColumn	

**JSON Schema for COLUMN sub-object type at level BASIC**

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "$id": "https://oracle.com/schema/23.7/DBMS_DEVELOPER.GET_METADATA/BASIC/
object_info/column",
  "title": "DBMS_DEVELOPER.GET_METADATA/BASIC/OBJECT_INFO/COLUMN",
  "description": "Information for column",
  "type": "object",
  "properties": {
    "name": {
      "description": "Column name",
      "type": "string"
    },
    "dataType": {
      "description": "Datatype",
      "type": "object",
      "properties": {
        "type": {
          "description": "dataType",
          "type": "string"
        },
        "precision": {
          "description": "precision",
          "type": "integer",
          "minimum": 1
        },
        "scale": {
          "description": "scale",
          "type": "integer",
          "minimum": 0
        }
      }
    }
  }
}
```

```

    },
    "size": {
      "description": "size",
      "type": "integer",
      "minimum": 1
    },
    "sizeUnits": {
      "description": "varchar size units",
      "type": "string",
      "enum": [
        "BYTE",
        "CHAR"
      ]
    },
    "fractionalSecondsPrecision": {
      "description": "fractionalSecondsPrecision",
      "type": "integer"
    },
    "yearPrecision": {
      "description": "yearPrecision",
      "type": "integer"
    },
    "dayPrecision": {
      "description": "dayPrecision",
      "type": "integer"
    }
  },
  "anyOf": [
    {
      "if": {
        "properties": {
          "type": {
            "const": "NUMBER"
          }
        }
      },
      "then": {
        "required": [
          "precision",
          "scale"
        ]
      }
    },
    {
      "if": {
        "properties": {
          "type": {
            "const": "FLOAT"
          }
        }
      },
      "then": {
        "required": [
          "precision"
        ]
      }
    }
  ]
}

```

```

    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "VARCHAR2"
                }
            }
        },
        "then": {
            "required": [
                "size",
                "sizeUnits"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "RAW"
                }
            }
        },
        "then": {
            "required": [
                "size"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "CHAR"
                }
            }
        },
        "then": {
            "required": [
                "size",
                "sizeUnits"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "NCHAR"
                }
            }
        },
        "then": {
            "required": [
                "size"
            ]
        }
    }
}

```

```

    ]
  }
},
{
  "if": {
    "properties": {
      "type": {
        "const": "UROWID"
      }
    }
  },
  "then": {
    "required": [
      "size"
    ]
  }
},
{
  "if": {
    "properties": {
      "type": {
        "const": "NVARCHAR2"
      }
    }
  },
  "then": {
    "required": [
      "size"
    ]
  }
},
{
  "if": {
    "properties": {
      "type": {
        "enum": [
          "TIMESTAMP",
          "TIMESTAMP WITH TIME ZONE",
          "TIMESTAMP WITH LOCAL TIME ZONE"
        ]
      }
    }
  },
  "then": {
    "required": [
      "fractionalSecondsPrecision"
    ]
  }
},
{
  "if": {
    "properties": {
      "type": {
        "const": "INTERVAL YEAR TO MONTH"
      }
    }
  }
}

```

```

        },
        "then": {
            "required": [
                "yearPrecision"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "INTERVAL DAY TO SECOND"
                }
            }
        },
        "then": {
            "required": [
                "yearPrecision",
                "fractionalSecondsPrecision"
            ]
        }
    }
]
},
"notNull": {
    "description": "Not null",
    "type": "boolean",
    "default": false
},
"default": {
    "description": "Default value",
    "anyOf": [
        {
            "type": "string"
        },
        {
            "type": "number"
        }
    ]
},
"required": [
    "name",
    "dataType",
    "notNull"
]
}
}

```

#### JSON Schema for COLUMN sub-object type at level TYPICAL

```

{
    "$schema": "https://json-schema.org/draft/2020-12/schema",
    "$id": "https://oracle.com/schema/23.7/DBMS_DEVELOPER.GET_METADATA/
TYPICAL/object_info/column",
    "title": "DBMS_DEVELOPER.GET_METADATA/TYPICAL/OBJECT_INFO/COLUMN",

```

```

"description": "Information for column",
"type": "object",
"properties": {
  "name": {
    "description": "Column name",
    "type": "string"
  },
  "isPk": {
    "description": "Is Primary Key",
    "type": "boolean",
    "default": false
  },
  "isUk": {
    "description": "Is Unique Key",
    "type": "boolean",
    "default": false
  },
  "isFk": {
    "description": "Is Foreign Key",
    "type": "boolean",
    "default": false
  },
  "dataType": {
    "description": "Datatype",
    "type": "object",
    "properties": {
      "type": {
        "description": "dataType",
        "type": "string"
      },
      "precision": {
        "description": "precision",
        "type": "integer",
        "minimum": 1
      },
      "scale": {
        "description": "scale",
        "type": "integer",
        "minimum": 0
      },
      "size": {
        "description": "size",
        "type": "integer",
        "minimum": 1
      },
      "sizeUnits": {
        "description": "varchar size units",
        "type": "string",
        "enum": [
          "BYTE",
          "CHAR"
        ]
      },
      "fractionalSecondsPrecision": {
        "description": "fractionalSecondsPrecision",
        "type": "integer"
      }
    }
  }
}

```

```

    },
    "yearPrecision": {
        "description": "yearPrecision",
        "type": "integer"
    },
    "dayPrecision": {
        "description": "dayPrecision",
        "type": "integer"
    }
},
"anyOf": [
    {
        "if": {
            "properties": {
                "type": {
                    "const": "NUMBER"
                }
            }
        },
        "then": {
            "required": [
                "precision",
                "scale"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "FLOAT"
                }
            }
        },
        "then": {
            "required": [
                "precision"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "VARCHAR2"
                }
            }
        },
        "then": {
            "required": [
                "size",
                "sizeUnits"
            ]
        }
    }
],
{

```

```

        "if": {
            "properties": {
                "type": {
                    "const": "RAW"
                }
            }
        },
        "then": {
            "required": [
                "size"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "CHAR"
                }
            }
        },
        "then": {
            "required": [
                "size",
                "sizeUnits"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "NCHAR"
                }
            }
        },
        "then": {
            "required": [
                "size"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "UROWID"
                }
            }
        },
        "then": {
            "required": [
                "size"
            ]
        }
    }
},

```



```

{
  "if": {
    "properties": {
      "type": {
        "const": "NVARCHAR2"
      }
    }
  },
  "then": {
    "required": [
      "size"
    ]
  }
},
{
  "if": {
    "properties": {
      "type": {
        "enum": [
          "TIMESTAMP",
          "TIMESTAMP WITH TIME ZONE",
          "TIMESTAMP WITH LOCAL TIME ZONE"
        ]
      }
    }
  },
  "then": {
    "required": [
      "fractionalSecondsPrecision"
    ]
  }
},
{
  "if": {
    "properties": {
      "type": {
        "const": "INTERVAL YEAR TO MONTH"
      }
    }
  },
  "then": {
    "required": [
      "yearPrecision"
    ]
  }
},
{
  "if": {
    "properties": {
      "type": {
        "const": "INTERVAL DAY TO SECOND"
      }
    }
  },
  "then": {
    "required": [

```

```

        "yearPrecision",
        "fractionalSecondsPrecision"
    ]
    }
}
]
},
"notNull": {
    "description": "Not null",
    "type": "boolean",
    "default": false
},
"default": {
    "description": "Default value",
    "anyOf": [
        {
            "type": "string"
        },
        {
            "type": "number"
        }
    ]
},
},
"domain": {
    "description": "Column domain",
    "type": "object",
    "properties": {
        "name": {
            "description": "Domain name",
            "type": "string"
        },
        "type": {
            "description": "Domain type",
            "type": "string"
        },
        "display": {
            "description": "Domain display",
            "type": "string"
        },
        "order": {
            "description": "Domain order",
            "type": "string"
        }
    }
},
"annotations": {
    "description": "Column annotations",
    "type": "array",
    "items": {
        "$ref": "annotation"
    }
},
"required": [
    "name",
    "isPk",
    "isUk",

```

```

        "isFk",
        "dataType",
        "notNull"
    ]
}
}

```

### JSON Schema for COLUMN sub-object type at level ALL

```

{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "$id": "https://oracle.com/schema/23.7/DBMS_DEVELOPER.GET_METADATA/ALL/object_info/column",
  "title": "DBMS_DEVELOPER.GET_METADATA/ALL/OBJECT_INFO/COLUMN",
  "description": "Information for column",
  "type": "object",
  "properties": {
    "name": {
      "description": "Column name",
      "type": "string"
    },
    "isPk": {
      "description": "Is Primary Key",
      "type": "boolean",
      "default": false
    },
    "isUk": {
      "description": "Is Unique Key",
      "type": "boolean",
      "default": false
    },
    "isFk": {
      "description": "Is Foreign Key",
      "type": "boolean",
      "default": false
    },
    "dataType": {
      "description": "Datatype",
      "type": "object",
      "properties": {
        "type": {
          "description": "dataType",
          "type": "string"
        },
        "precision": {
          "description": "precision",
          "type": "integer",
          "minimum": 1
        },
        "scale": {
          "description": "scale",
          "type": "integer",
          "minimum": 0
        },
        "size": {

```

```

        "description": "size",
        "type": "integer",
        "minimum": 1
    },
    "sizeUnits": {
        "description": "varchar size units",
        "type": "string",
        "enum": [
            "BYTE",
            "CHAR"
        ]
    },
    "fractionalSecondsPrecision": {
        "description": "fractionalSecondsPrecision",
        "type": "integer"
    },
    "yearPrecision": {
        "description": "yearPrecision",
        "type": "integer"
    },
    "dayPrecision": {
        "description": "dayPrecision",
        "type": "integer"
    }
},
"anyOf": [
    {
        "if": {
            "properties": {
                "type": {
                    "const": "NUMBER"
                }
            }
        },
        "then": {
            "required": [
                "precision",
                "scale"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "FLOAT"
                }
            }
        },
        "then": {
            "required": [
                "precision"
            ]
        }
    }
],
{

```

```

        "if": {
            "properties": {
                "type": {
                    "const": "VARCHAR2"
                }
            }
        },
        "then": {
            "required": [
                "size",
                "sizeUnits"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "RAW"
                }
            }
        },
        "then": {
            "required": [
                "size"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "CHAR"
                }
            }
        },
        "then": {
            "required": [
                "size",
                "sizeUnits"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "NCHAR"
                }
            }
        },
        "then": {
            "required": [
                "size"
            ]
        }
    }
}

```

```

    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "UROWID"
                }
            }
        },
        "then": {
            "required": [
                "size"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "NVARCHAR2"
                }
            }
        },
        "then": {
            "required": [
                "size"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "enum": [
                        "TIMESTAMP",
                        "TIMESTAMP WITH TIME ZONE",
                        "TIMESTAMP WITH LOCAL TIME ZONE"
                    ]
                }
            }
        },
        "then": {
            "required": [
                "fractionalSecondsPrecision"
            ]
        }
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "INTERVAL YEAR TO MONTH"
                }
            }
        },
        "then": {

```

```

        "required": [
            "yearPrecision"
        ]
    },
    {
        "if": {
            "properties": {
                "type": {
                    "const": "INTERVAL DAY TO SECOND"
                }
            }
        },
        "then": {
            "required": [
                "yearPrecision",
                "fractionalSecondsPrecision"
            ]
        }
    }
]
},
"notNull": {
    "description": "Not null",
    "type": "boolean",
    "default": false
},
"numDistinct": {
    "description": "The number of distinct values in the column",
    "type": "integer"
},
"lowValue": {
    "description": "The low value in the column",
    "type": "string"
},
"highValue": {
    "description": "The high value in the column",
    "type": "string"
},
"density": {
    "description": "The average length of the column in bytes",
    "type": "integer"
},
"avgCollLength": {
    "description": "The number of distinct values in the column",
    "type": "integer"
},
"hiddenColumn": {
    "description": "Is this a hidden column?",
    "type": "boolean"
},
"default": {
    "description": "Default value",
    "anyOf": [
        {
            "type": "string"
        }
    ]
}

```

```

        },
        {
            "type": "number"
        }
    ]
},
"domain": {
    "description": "Column domain",
    "type": "object",
    "properties": {
        "name": {
            "description": "Domain name",
            "type": "string"
        },
        "type": {
            "description": "Domain type",
            "type": "string"
        },
        "display": {
            "description": "Domain display",
            "type": "string"
        },
        "order": {
            "description": "Domain order",
            "type": "string"
        }
    },
}
},
"annotations": {
    "description": "Column annotations",
    "type": "array",
    "items": {
        "$ref": "annotation"
    }
},
"required": [
    "name",
    "isPk",
    "isUk",
    "isFk",
    "dataType",
    "notNull",
    "hiddenColumn"
]
}
}

```



## JSON Schema for Sub-Object Type : `constraints`

This section describes the JSON schema for the sub-object `constraints` which belongs to all the object types.

### JSON Schema for `constraints` Sub-Object Type in Tabular Form

**Table 76-6 JSON Schema for Sub-Object Type : CONSTRAINTS**

Fields			
	BASIC	TYPICAL	ALL
<code>name</code>			
<code>constraintType</code>			
<code>searchCondition</code>			
<code>columns</code> <code>name</code>			
<code>referencedConstraintName</code>			
<code>referencedTable</code>			
<code>Action</code>			
<code>referencedOwner</code>			
<code>referencedColumns</code> <code>name</code>			
<code>status</code>			
<code>deferrable</code>			
<code>validated</code>			
<code>sysGeneratedName</code>			

### JSON Schema for `COLUMN` sub-object type at level `TYPICAL`

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "$id": "https://oracle.com/schema/23.7/DBMS_DEVELOPER.GET_METADATA/
TYPICAL/object_info/constraint",
  "title": "DBMS_DEVELOPER.GET_METADATA/TYPICAL/OBJECT_INFO/CONSTRAINT",
  "description": "Information for constraint",
  "type": "object",
  "properties": {
    "name": {
      "description": "Constraint name",
      "type": "string"
    },
    "constraintType": {
      "description": "Type of constraint definition: CHECK, CHECK - NOT
NULL, PRIMARY KEY, UNIQUE, REFERENTIAL INTEGRITY, VIEW CHECK, VIEW READONLY",
      "type": "string"
    },
    "searchCondition": {
      "description": "Text of search condition for a check constraint",
      "type": "string"
    },
    "columns": {
```

```

        "description": "Table columns",
        "type": "array",
        "items": {
            "type": "object",
            "properties": {
                "name": {
                    "description": "Name associated with column or
attribute of object column specified in the constraint definition",
                    "type": "string"
                }
            }
        }
    },
    "referencedConstraintName": {
        "description": "Name of unique constraint definition for
refernced table",
        "type": "string"
    },
    "referencedOwner": {
        "description": "Owner of table used in referential constraint",
        "type": "string"
    },
    "referencedTable": {
        "description": "Name of table used in referential constraint",
        "type": "string"
    },
    "referencedColumns": {
        "description": "Refernced table columns",
        "type": "array",
        "items": {
            "type": "object",
            "properties": {
                "name": {
                    "description": "Refernced table column name",
                    "type": "string"
                }
            }
        }
    },
    "Action": {
        "description": "On delete rule",
        "type": {
            "enum": [
                "RESTRICT",
                "CASCADE",
                "SET NULL",
                "SET DEFAULT"
            ]
        }
    },
    "status": {
        "description": "Constraint status",
        "type": {
            "enum": [
                "ENABLED",
                "DISABLED"
            ]
        }
    }
}

```

```

        ]
      }
    },
    "deferrable": {
      "description": "Deferrable",
      "type": "boolean",
      "default": false
    },
    "validated": {
      "description": "Validated",
      "type": {
        "enum": [
          "VALIDATED",
          "NONVALIDATED"
        ]
      }
    },
    "sysGeneratedName": {
      "description": "System Generated constraint",
      "type": "boolean"
    },
    "required": [
      "name",
      "constarintType",
      "status",
      "deferrable",
      "validated",
      "sysGeneratedName"
    ],
    "if": {
      "properties": {
        "constarintType": {
          "const": "REFERENTIAL INTEGRITY"
        }
      }
    },
    "then": {
      "required": [
        "referencedConstraintName",
        "referencedOwner",
        "referencedTable",
        "referencedColumns"
      ]
    }
  }
}

```

#### JSON Schema for COLUMN sub-object type at level ALL

```

{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "$id": "https://oracle.com/schema/23.7/DBMS_DEVELOPER.GET_METADATA/ALL/object_info/constraint",
  "title": "DBMS_DEVELOPER.GET_METADATA/ALL/OBJECT_INFO/CONSTRAINT",
  "description": "Information for constraint",

```

```

"type": "object",
"properties": {
  "name": {
    "description": "Constraint name",
    "type": "string"
  },
  "constraintType": {
    "description": "Type of constraint definition: CHECK, CHECK - NOT
NULL, PRIMARY KEY, UNIQUE, REFERENTIAL INTEGRITY, VIEW CHECK, VIEW READONLY",
    "type": "string"
  },
  "searchCondition": {
    "description": "Text of search condition for a check constraint",
    "type": "string"
  },
  "columns": {
    "description": "Table columns",
    "type": "array",
    "items": {
      "type": "object",
      "properties": {
        "name": {
          "description": "Name associated with column or
attribute of object column specified in the constraint definition",
          "type": "string"
        }
      }
    }
  },
  "referencedConstraintName": {
    "description": "Name of unique constraint definition for
refernced table",
    "type": "string"
  },
  "referencedOwner": {
    "description": "Owner of table used in referential constraint",
    "type": "string"
  },
  "referencedTable": {
    "description": "Name of table used in referential constraint",
    "type": "string"
  },
  "referencedColumns": {
    "description": "Refernced table columns",
    "type": "array",
    "items": {
      "type": "object",
      "properties": {
        "name": {
          "description": "Refernced table column name",
          "type": "string"
        }
      }
    }
  },
  "Action": {

```

```

        "description": "On delete rule",
        "type": {
            "enum": [
                "RESTRICT",
                "CASCADE",
                "SET NULL",
                "SET DEFAULT"
            ]
        }
    },
    "status": {
        "description": "Constraint status",
        "type": {
            "enum": [
                "ENABLED",
                "DISABLED"
            ]
        }
    },
    "deferrable": {
        "description": "Deferrable",
        "type": "boolean",
        "default": false
    },
    "validated": {
        "description": "Validated",
        "type": {
            "enum": [
                "VALIDATED",
                "NONVALIDATED"
            ]
        }
    },
    "sysGeneratedName": {
        "description": "System Generated constraint",
        "type": "boolean"
    }
},
"required": [
    "name",
    "constarintType",
    "status",
    "deferrable",
    "validated",
    "sysGeneratedName"
],
"if": {
    "properties": {
        "constarintType": {
            "const": "REFERENTIAL INTEGRITY"
        }
    }
},
"then": {
    "required": [
        "referencedConstraintName",
        "referencedOwner",

```

```
        "referencedTable",  
        "referencedColumns"  
    ]  
}  
}
```