# **QeuBox Low Level Design Document**

## **Screen Name: Box Cell Food Hist**

### **Database Table Name: box\_cell\_food\_hist**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| box\_cell\_food\_sno | bigint - (Numeric Precision: 64) | box\_cell\_food\_hist\_pkey (PRIMARY KEY), |
| box\_cell\_sno | smallint - (Numeric Precision: 16) | null |
| sku\_inventory\_sno | bigint - (Numeric Precision: 64) | null |
| entry\_time | timestamp without time zone | null |
| qbox\_entity\_sno | bigint - (Numeric Precision: 64) | null |
| description | text | null |
| active\_flag | boolean | null |
| partner\_sales\_order\_id | text | null |
| is\_current | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Box Cell Food Sno | hidden | number | true | Manual Entry |
| Box Cell Sno | textbox\_number | number | true | Manual Entry |
| Sku Inventory Sno | textbox\_number | number | true | Manual Entry |
| Entry Time | textbox | string | true | Manual Entry |
| Qbox Entity Sno | textbox\_number | number | true | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |
| Partner Sales Order Id | textbox | string | false | Manual Entry |
| Is Current | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_box\_cell\_food\_hists | get | /api/v1/get\_box\_cell\_food\_hists |
| create\_box\_cell\_food\_hist | post | /api/v1/create\_box\_cell\_food\_hist |
| update\_box\_cell\_food\_hist | put | /api/v1/update\_box\_cell\_food\_hist |
| delete\_box\_cell\_food\_hist | delete | /api/v1/delete\_box\_cell\_food\_hist |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_box\_cell\_food\_hists | {} | {"data":{"entry\_time":"timestamp without time zone","box\_cell\_food\_sno":"bigint","description":"text","qbox\_entity\_sno":"bigint","active\_flag":"boolean","box\_cell\_sno":"smallint","sku\_inventory\_sno":"bigint","is\_current":"boolean","partner\_sales\_order\_id":"text"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_box\_cell\_food\_hist | {"data":{"entry\_time":"timestamp without time zone","box\_cell\_food\_sno":"bigint","description":"text","qbox\_entity\_sno":"bigint","active\_flag":"boolean","box\_cell\_sno":"smallint","sku\_inventory\_sno":"bigint","is\_current":"boolean","partner\_sales\_order\_id":"text"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_box\_cell\_food\_hist | {"data":{"entry\_time":"timestamp without time zone","box\_cell\_food\_sno":"bigint","description":"text","qbox\_entity\_sno":"bigint","active\_flag":"boolean","box\_cell\_sno":"smallint","sku\_inventory\_sno":"bigint","is\_current":"boolean","partner\_sales\_order\_id":"text"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_box\_cell\_food\_hist | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| get\_all\_box\_cell\_food\_hists | SELECT t.box\_cell\_food\_sno, t.box\_cell\_sno, t.sku\_inventory\_sno, t.entry\_time, t.qbox\_entity\_sno, t.description, t.active\_flag, t.partner\_sales\_order\_id, t.is\_current FROM masters.box\_cell\_food\_hist t ; |
| delete\_box\_cell\_food\_hist\_by\_id | DELETE FROM masters.box\_cell\_food\_hist WHERE box\_cell\_food\_sno = quote\_nullable(i\_box\_cell\_food\_sno); |
| get\_box\_cell\_food\_hist\_by\_id | SELECT t.box\_cell\_food\_sno, t.box\_cell\_sno, t.sku\_inventory\_sno, t.entry\_time, t.qbox\_entity\_sno, t.description, t.active\_flag, t.partner\_sales\_order\_id, t.is\_current FROM masters.box\_cell\_food\_hist t WHERE t.box\_cell\_food\_sno = quote\_nullable(i\_box\_cell\_food\_sno); |
| create\_box\_cell\_food\_hist | INSERT INTO masters.box\_cell\_food\_hist (t.box\_cell\_food\_sno, t.box\_cell\_sno, t.sku\_inventory\_sno, t.entry\_time, t.qbox\_entity\_sno, t.description, t.active\_flag, t.partner\_sales\_order\_id, t.is\_current) VALUES (i\_box\_cell\_food\_sno, i\_box\_cell\_sno, i\_sku\_inventory\_sno, i\_entry\_time, i\_qbox\_entity\_sno, i\_description, i\_active\_flag, i\_partner\_sales\_order\_id, i\_is\_current); |
| custom\_delete\_box\_cell\_food\_hist\_by\_id | DELETE FROM masters.box\_cell\_food\_hist WHERE <custom\_conditions>; |
| custom\_search\_box\_cell\_food\_hist\_by\_id | SELECT t.box\_cell\_food\_sno, t.box\_cell\_sno, t.sku\_inventory\_sno, t.entry\_time, t.qbox\_entity\_sno, t.description, t.active\_flag, t.partner\_sales\_order\_id, t.is\_current FROM masters.box\_cell\_food\_hist t WHERE <custom\_search\_conditions>; |
| custom\_update\_box\_cell\_food\_hist | UPDATE masters.box\_cell\_food\_hist SET box\_cell\_food\_sno = quote\_nullable(i\_box\_cell\_food\_sno), box\_cell\_sno = quote\_nullable(i\_box\_cell\_sno), sku\_inventory\_sno = quote\_nullable(i\_sku\_inventory\_sno), entry\_time = quote\_nullable(i\_entry\_time), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), partner\_sales\_order\_id = quote\_nullable(i\_partner\_sales\_order\_id), is\_current = quote\_nullable(i\_is\_current) WHERE <custom\_conditions>; |
| update\_box\_cell\_food\_hist | UPDATE masters.box\_cell\_food\_hist SET box\_cell\_food\_sno = quote\_nullable(i\_box\_cell\_food\_sno), box\_cell\_sno = quote\_nullable(i\_box\_cell\_sno), sku\_inventory\_sno = quote\_nullable(i\_sku\_inventory\_sno), entry\_time = quote\_nullable(i\_entry\_time), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), partner\_sales\_order\_id = quote\_nullable(i\_partner\_sales\_order\_id), is\_current = quote\_nullable(i\_is\_current) WHERE box\_cell\_food\_sno = quote\_nullable(i\_box\_cell\_food\_sno); |
| unique\_box\_cell\_food\_hist\_exists | SELECT 1 FROM masters.box\_cell\_food\_hist WHERE <unique\_conditions> LIMIT 1; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Codes Hdr**

### **Database Table Name: codes\_hdr**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| codes\_hdr\_sno | smallint - (Numeric Precision: 16) | codes\_hdr\_pkey (PRIMARY KEY), |
| description | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Codes Hdr Sno | hidden | number | true | Manual Entry |
| Description | textbox | string | true | Manual Entry |
| Active Flag | checkbox | boolean | true | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_codes\_hdrs | get | /api/v1/get\_codes\_hdrs |
| create\_codes\_hdr | post | /api/v1/create\_codes\_hdr |
| update\_codes\_hdr | put | /api/v1/update\_codes\_hdr |
| delete\_codes\_hdr | delete | /api/v1/delete\_codes\_hdr |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_codes\_hdrs | {} | {"data":{"description":"text","active\_flag":"boolean","codes\_hdr\_sno":"smallint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_codes\_hdr | {"data":{"description":"text","active\_flag":"boolean","codes\_hdr\_sno":"smallint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_codes\_hdr | {"data":{"description":"text","active\_flag":"boolean","codes\_hdr\_sno":"smallint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_codes\_hdr | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| unique\_codes\_hdr\_exists | SELECT 1 FROM masters.codes\_hdr WHERE <unique\_conditions> LIMIT 1; |
| custom\_search\_codes\_hdr\_by\_id | SELECT t.codes\_hdr\_sno, t.description, t.active\_flag FROM masters.codes\_hdr t WHERE <custom\_search\_conditions>; |
| delete\_codes\_hdr\_by\_id | DELETE FROM masters.codes\_hdr WHERE codes\_hdr\_sno = quote\_nullable(i\_codes\_hdr\_sno); |
| get\_codes\_hdr\_by\_id | SELECT t.codes\_hdr\_sno, t.description, t.active\_flag FROM masters.codes\_hdr t WHERE t.codes\_hdr\_sno = quote\_nullable(i\_codes\_hdr\_sno); |
| custom\_update\_codes\_hdr | UPDATE masters.codes\_hdr SET codes\_hdr\_sno = quote\_nullable(i\_codes\_hdr\_sno), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| custom\_delete\_codes\_hdr\_by\_id | DELETE FROM masters.codes\_hdr WHERE <custom\_conditions>; |
| update\_codes\_hdr | UPDATE masters.codes\_hdr SET codes\_hdr\_sno = quote\_nullable(i\_codes\_hdr\_sno), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE codes\_hdr\_sno = quote\_nullable(i\_codes\_hdr\_sno); |
| create\_codes\_hdr | INSERT INTO masters.codes\_hdr (t.codes\_hdr\_sno, t.description, t.active\_flag) VALUES (i\_codes\_hdr\_sno, i\_description, i\_active\_flag); |
| get\_all\_codes\_hdrs | SELECT t.codes\_hdr\_sno, t.description, t.active\_flag FROM masters.codes\_hdr t ; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Order**

### **Database Table Name: order**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| order\_sno | bigint - (Numeric Precision: 64) | order\_sno\_pkey (PRIMARY KEY), |
| partner\_order\_id | text | null |
| order\_type | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Order Sno | hidden | number | true | Manual Entry |
| Partner Order Id | textbox | string | true | Manual Entry |
| Order Type | textbox | string | true | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_orders | get | /api/v1/get\_orders |
| create\_order | post | /api/v1/create\_order |
| update\_order | put | /api/v1/update\_order |
| delete\_order | delete | /api/v1/delete\_order |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_orders | {} | {"data":{"partner\_order\_id":"text","active\_flag":"boolean","order\_sno":"bigint","order\_type":"text"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_order | {"data":{"partner\_order\_id":"text","active\_flag":"boolean","order\_sno":"bigint","order\_type":"text"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_order | {"data":{"partner\_order\_id":"text","active\_flag":"boolean","order\_sno":"bigint","order\_type":"text"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_order | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| update\_order | UPDATE masters.order SET order\_sno = quote\_nullable(i\_order\_sno), partner\_order\_id = quote\_nullable(i\_partner\_order\_id), order\_type = quote\_nullable(i\_order\_type), active\_flag = quote\_nullable(i\_active\_flag) WHERE order\_sno = quote\_nullable(i\_order\_sno); |
| custom\_search\_order\_by\_id | SELECT t.order\_sno, t.partner\_order\_id, t.order\_type, t.active\_flag FROM masters.order t WHERE <custom\_search\_conditions>; |
| unique\_order\_exists | SELECT 1 FROM masters.order WHERE <unique\_conditions> LIMIT 1; |
| create\_order | INSERT INTO masters.order (t.order\_sno, t.partner\_order\_id, t.order\_type, t.active\_flag) VALUES (i\_order\_sno, i\_partner\_order\_id, i\_order\_type, i\_active\_flag); |
| delete\_order\_by\_id | DELETE FROM masters.order WHERE order\_sno = quote\_nullable(i\_order\_sno); |
| get\_all\_orders | SELECT t.order\_sno, t.partner\_order\_id, t.order\_type, t.active\_flag FROM masters.order t ; |
| custom\_update\_order | UPDATE masters.order SET order\_sno = quote\_nullable(i\_order\_sno), partner\_order\_id = quote\_nullable(i\_partner\_order\_id), order\_type = quote\_nullable(i\_order\_type), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| get\_order\_by\_id | SELECT t.order\_sno, t.partner\_order\_id, t.order\_type, t.active\_flag FROM masters.order t WHERE t.order\_sno = quote\_nullable(i\_order\_sno); |
| custom\_delete\_order\_by\_id | DELETE FROM masters.order WHERE <custom\_conditions>; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Reject Reason**

### **Database Table Name: reject\_reason**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| reject\_reason\_sno | bigint - (Numeric Precision: 64) | reject\_reason\_pkey (PRIMARY KEY), |
| reject\_reason | text | null |
| description | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Reject Reason Sno | hidden | number | true | Manual Entry |
| Reject Reason | textbox | string | true | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_reject\_reasons | get | /api/v1/get\_reject\_reasons |
| create\_reject\_reason | post | /api/v1/create\_reject\_reason |
| update\_reject\_reason | put | /api/v1/update\_reject\_reason |
| delete\_reject\_reason | delete | /api/v1/delete\_reject\_reason |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_reject\_reasons | {} | {"data":{"reject\_reason\_sno":"bigint","description":"text","active\_flag":"boolean","reject\_reason":"text"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_reject\_reason | {"data":{"reject\_reason\_sno":"bigint","description":"text","active\_flag":"boolean","reject\_reason":"text"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_reject\_reason | {"data":{"reject\_reason\_sno":"bigint","description":"text","active\_flag":"boolean","reject\_reason":"text"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_reject\_reason | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| get\_reject\_reason\_by\_id | SELECT t.reject\_reason\_sno, t.reject\_reason, t.description, t.active\_flag FROM masters.reject\_reason t WHERE t.reject\_reason\_sno = quote\_nullable(i\_reject\_reason\_sno); |
| delete\_reject\_reason\_by\_id | DELETE FROM masters.reject\_reason WHERE reject\_reason\_sno = quote\_nullable(i\_reject\_reason\_sno); |
| create\_reject\_reason | INSERT INTO masters.reject\_reason (t.reject\_reason\_sno, t.reject\_reason, t.description, t.active\_flag) VALUES (i\_reject\_reason\_sno, i\_reject\_reason, i\_description, i\_active\_flag); |
| get\_all\_reject\_reasons | SELECT t.reject\_reason\_sno, t.reject\_reason, t.description, t.active\_flag FROM masters.reject\_reason t ; |
| custom\_update\_reject\_reason | UPDATE masters.reject\_reason SET reject\_reason\_sno = quote\_nullable(i\_reject\_reason\_sno), reject\_reason = quote\_nullable(i\_reject\_reason), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| custom\_delete\_reject\_reason\_by\_id | DELETE FROM masters.reject\_reason WHERE <custom\_conditions>; |
| unique\_reject\_reason\_exists | SELECT 1 FROM masters.reject\_reason WHERE <unique\_conditions> LIMIT 1; |
| custom\_search\_reject\_reason\_by\_id | SELECT t.reject\_reason\_sno, t.reject\_reason, t.description, t.active\_flag FROM masters.reject\_reason t WHERE <custom\_search\_conditions>; |
| update\_reject\_reason | UPDATE masters.reject\_reason SET reject\_reason\_sno = quote\_nullable(i\_reject\_reason\_sno), reject\_reason = quote\_nullable(i\_reject\_reason), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE reject\_reason\_sno = quote\_nullable(i\_reject\_reason\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: App Role**

### **Database Table Name: app\_role**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| app\_role\_sno | smallint - (Numeric Precision: 16) | app\_role\_pkey (PRIMARY KEY), |
| name | text | null |
| status | boolean | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| App Role Sno | hidden | number | true | Manual Entry |
| Name | textbox | string | false | Manual Entry |
| Status | checkbox | boolean | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_app\_roles | get | /api/v1/get\_app\_roles |
| create\_app\_role | post | /api/v1/create\_app\_role |
| update\_app\_role | put | /api/v1/update\_app\_role |
| delete\_app\_role | delete | /api/v1/delete\_app\_role |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_app\_roles | {} | {"data":{"name":"text","active\_flag":"boolean","app\_role\_sno":"smallint","status":"boolean"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_app\_role | {"data":{"name":"text","active\_flag":"boolean","app\_role\_sno":"smallint","status":"boolean"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_app\_role | {"data":{"name":"text","active\_flag":"boolean","app\_role\_sno":"smallint","status":"boolean"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_app\_role | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| delete\_app\_role\_by\_id | DELETE FROM masters.app\_role WHERE app\_role\_sno = quote\_nullable(i\_app\_role\_sno); |
| update\_app\_role | UPDATE masters.app\_role SET app\_role\_sno = quote\_nullable(i\_app\_role\_sno), name = quote\_nullable(i\_name), status = quote\_nullable(i\_status), active\_flag = quote\_nullable(i\_active\_flag) WHERE app\_role\_sno = quote\_nullable(i\_app\_role\_sno); |
| custom\_delete\_app\_role\_by\_id | DELETE FROM masters.app\_role WHERE <custom\_conditions>; |
| custom\_search\_app\_role\_by\_id | SELECT t.app\_role\_sno, t.name, t.status, t.active\_flag FROM masters.app\_role t WHERE <custom\_search\_conditions>; |
| custom\_update\_app\_role | UPDATE masters.app\_role SET app\_role\_sno = quote\_nullable(i\_app\_role\_sno), name = quote\_nullable(i\_name), status = quote\_nullable(i\_status), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| get\_app\_role\_by\_id | SELECT t.app\_role\_sno, t.name, t.status, t.active\_flag FROM masters.app\_role t WHERE t.app\_role\_sno = quote\_nullable(i\_app\_role\_sno); |
| unique\_app\_role\_exists | SELECT 1 FROM masters.app\_role WHERE <unique\_conditions> LIMIT 1; |
| get\_all\_app\_roles | SELECT t.app\_role\_sno, t.name, t.status, t.active\_flag FROM masters.app\_role t ; |
| create\_app\_role | INSERT INTO masters.app\_role (t.app\_role\_sno, t.name, t.status, t.active\_flag) VALUES (i\_app\_role\_sno, i\_name, i\_status, i\_active\_flag); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: App User Role**

### **Database Table Name: app\_user\_role**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| app\_user\_role\_sno | bigint - (Numeric Precision: 64) | app\_user\_role\_pkey (PRIMARY KEY), |
| app\_user\_sno | bigint - (Numeric Precision: 64) | app\_user\_role\_app\_user\_sno\_fkey (FOREIGN KEY), |
| app\_role\_sno | smallint - (Numeric Precision: 16) | app\_user\_role\_app\_role\_sno\_fkey (FOREIGN KEY), |
| description | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| App User Role Sno | hidden | number | true | Manual Entry |
| App User Sno | listbox | number | false | API feed list box selection |
| App Role Sno | listbox | number | true | API feed list box selection |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_app\_user\_roles | get | /api/v1/get\_app\_user\_roles |
| create\_app\_user\_role | post | /api/v1/create\_app\_user\_role |
| update\_app\_user\_role | put | /api/v1/update\_app\_user\_role |
| delete\_app\_user\_role | delete | /api/v1/delete\_app\_user\_role |
| get\_app\_roles | get | /api/v1/get\_app\_roles |
| get\_app\_users | get | /api/v1/get\_app\_users |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_app\_user\_roles | {} | {"data":{"description":"text","app\_user\_sno":"bigint","active\_flag":"boolean","app\_role\_sno":"smallint","app\_user\_role\_sno":"bigint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_app\_user\_role | {"data":{"description":"text","app\_user\_sno":"bigint","active\_flag":"boolean","app\_role\_sno":"smallint","app\_user\_role\_sno":"bigint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_app\_user\_role | {"data":{"description":"text","app\_user\_sno":"bigint","active\_flag":"boolean","app\_role\_sno":"smallint","app\_user\_role\_sno":"bigint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_app\_user\_role | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_app\_roles | {"id":"integer"} | {"data":"array of app\_role"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_app\_users | {"id":"integer"} | {"data":"array of app\_user"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| custom\_update\_app\_user\_role | UPDATE masters.app\_user\_role SET app\_user\_role\_sno = quote\_nullable(i\_app\_user\_role\_sno), app\_user\_sno = quote\_nullable(i\_app\_user\_sno), app\_role\_sno = quote\_nullable(i\_app\_role\_sno), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| custom\_delete\_app\_user\_role\_by\_id | DELETE FROM masters.app\_user\_role WHERE <custom\_conditions>; |
| custom\_search\_app\_user\_role\_by\_id | SELECT t.app\_user\_role\_sno, t.app\_user\_sno, t.app\_role\_sno, t.description, t.active\_flag, f\_app\_role.name, f\_app\_user.name FROM masters.app\_user\_role t LEFT JOIN masters.app\_role f\_app\_role ON t.app\_role\_sno = f\_app\_role.app\_role\_sno LEFT JOIN masters.app\_user f\_app\_user ON t.app\_user\_sno = f\_app\_user.app\_user\_sno WHERE <custom\_search\_conditions>; |
| delete\_app\_user\_role\_by\_id | DELETE FROM masters.app\_user\_role WHERE app\_user\_role\_sno = quote\_nullable(i\_app\_user\_role\_sno); |
| create\_app\_user\_role | INSERT INTO masters.app\_user\_role (t.app\_user\_role\_sno, t.app\_user\_sno, t.app\_role\_sno, t.description, t.active\_flag) VALUES (i\_app\_user\_role\_sno, i\_app\_user\_sno, i\_app\_role\_sno, i\_description, i\_active\_flag); |
| unique\_app\_user\_role\_exists | SELECT 1 FROM masters.app\_user\_role WHERE <unique\_conditions> LIMIT 1; |
| get\_app\_user\_role\_by\_id | SELECT t.app\_user\_role\_sno, t.app\_user\_sno, t.app\_role\_sno, t.description, t.active\_flag, f\_app\_role.name, f\_app\_user.name FROM masters.app\_user\_role t LEFT JOIN masters.app\_role f\_app\_role ON t.app\_role\_sno = f\_app\_role.app\_role\_sno LEFT JOIN masters.app\_user f\_app\_user ON t.app\_user\_sno = f\_app\_user.app\_user\_sno WHERE t.app\_user\_role\_sno = quote\_nullable(i\_app\_user\_role\_sno); |
| get\_all\_app\_user\_roles | SELECT t.app\_user\_role\_sno, t.app\_user\_sno, t.app\_role\_sno, t.description, t.active\_flag, f\_app\_role.name, f\_app\_user.name FROM masters.app\_user\_role t LEFT JOIN masters.app\_role f\_app\_role ON t.app\_role\_sno = f\_app\_role.app\_role\_sno LEFT JOIN masters.app\_user f\_app\_user ON t.app\_user\_sno = f\_app\_user.app\_user\_sno ; |
| update\_app\_user\_role | UPDATE masters.app\_user\_role SET app\_user\_role\_sno = quote\_nullable(i\_app\_user\_role\_sno), app\_user\_sno = quote\_nullable(i\_app\_user\_sno), app\_role\_sno = quote\_nullable(i\_app\_role\_sno), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE app\_user\_role\_sno = quote\_nullable(i\_app\_user\_role\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: App User**

### **Database Table Name: app\_user**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| app\_user\_sno | bigint - (Numeric Precision: 64) | app\_user\_pkey (PRIMARY KEY), |
| user\_id | text | app\_user\_user\_id\_key (UNIQUE), |
| password | text | null |
| status | boolean | null |
| name | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| App User Sno | hidden | number | true | Manual Entry |
| User Id | textbox | string | true | Manual Entry |
| Password | textbox | string | false | Manual Entry |
| Status | checkbox | boolean | false | Manual Entry |
| Name | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_app\_users | get | /api/v1/get\_app\_users |
| create\_app\_user | post | /api/v1/create\_app\_user |
| update\_app\_user | put | /api/v1/update\_app\_user |
| delete\_app\_user | delete | /api/v1/delete\_app\_user |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_app\_users | {} | {"data":{"password":"text","user\_id":"text","name":"text","app\_user\_sno":"bigint","active\_flag":"boolean","status":"boolean"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_app\_user | {"data":{"password":"text","user\_id":"text","name":"text","app\_user\_sno":"bigint","active\_flag":"boolean","status":"boolean"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_app\_user | {"data":{"password":"text","user\_id":"text","name":"text","app\_user\_sno":"bigint","active\_flag":"boolean","status":"boolean"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_app\_user | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| get\_all\_app\_users | SELECT t.app\_user\_sno, t.user\_id, t.password, t.status, t.name, t.active\_flag FROM masters.app\_user t ; |
| delete\_app\_user\_by\_id | DELETE FROM masters.app\_user WHERE app\_user\_sno = quote\_nullable(i\_app\_user\_sno); |
| update\_app\_user | UPDATE masters.app\_user SET app\_user\_sno = quote\_nullable(i\_app\_user\_sno), user\_id = quote\_nullable(i\_user\_id), password = quote\_nullable(i\_password), status = quote\_nullable(i\_status), name = quote\_nullable(i\_name), active\_flag = quote\_nullable(i\_active\_flag) WHERE app\_user\_sno = quote\_nullable(i\_app\_user\_sno); |
| get\_app\_user\_by\_id | SELECT t.app\_user\_sno, t.user\_id, t.password, t.status, t.name, t.active\_flag FROM masters.app\_user t WHERE t.app\_user\_sno = quote\_nullable(i\_app\_user\_sno); |
| custom\_update\_app\_user | UPDATE masters.app\_user SET app\_user\_sno = quote\_nullable(i\_app\_user\_sno), user\_id = quote\_nullable(i\_user\_id), password = quote\_nullable(i\_password), status = quote\_nullable(i\_status), name = quote\_nullable(i\_name), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| create\_app\_user | INSERT INTO masters.app\_user (t.app\_user\_sno, t.user\_id, t.password, t.status, t.name, t.active\_flag) VALUES (i\_app\_user\_sno, i\_user\_id, i\_password, i\_status, i\_name, i\_active\_flag); |
| custom\_delete\_app\_user\_by\_id | DELETE FROM masters.app\_user WHERE <custom\_conditions>; |
| custom\_search\_app\_user\_by\_id | SELECT t.app\_user\_sno, t.user\_id, t.password, t.status, t.name, t.active\_flag FROM masters.app\_user t WHERE <custom\_search\_conditions>; |
| unique\_app\_user\_exists | SELECT 1 FROM masters.app\_user WHERE <unique\_conditions> LIMIT 1; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: City**

### **Database Table Name: city**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| city\_sno | bigint - (Numeric Precision: 64) | city\_pkey (PRIMARY KEY), |
| country\_sno | smallint - (Numeric Precision: 16) | city\_country\_sno\_fkey (FOREIGN KEY), |
| state\_sno | smallint - (Numeric Precision: 16) | city\_state\_sno\_fkey (FOREIGN KEY), |
| name | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| City Sno | hidden | number | true | Manual Entry |
| Country Sno | listbox | number | true | API feed list box selection |
| State Sno | listbox | number | true | API feed list box selection |
| Name | textbox | string | true | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_citys | get | /api/v1/get\_citys |
| create\_city | post | /api/v1/create\_city |
| update\_city | put | /api/v1/update\_city |
| delete\_city | delete | /api/v1/delete\_city |
| get\_countrys | get | /api/v1/get\_countrys |
| get\_states | get | /api/v1/get\_states |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_citys | {} | {"data":{"state\_sno":"smallint","name":"text","country\_sno":"smallint","city\_sno":"bigint","active\_flag":"boolean"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_city | {"data":{"state\_sno":"smallint","name":"text","country\_sno":"smallint","city\_sno":"bigint","active\_flag":"boolean"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_city | {"data":{"state\_sno":"smallint","name":"text","country\_sno":"smallint","city\_sno":"bigint","active\_flag":"boolean"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_city | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_countrys | {"id":"integer"} | {"data":"array of country"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_states | {"id":"integer"} | {"data":"array of state"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| create\_city | INSERT INTO masters.city (t.city\_sno, t.country\_sno, t.state\_sno, t.name, t.active\_flag) VALUES (i\_city\_sno, i\_country\_sno, i\_state\_sno, i\_name, i\_active\_flag); |
| custom\_update\_city | UPDATE masters.city SET city\_sno = quote\_nullable(i\_city\_sno), country\_sno = quote\_nullable(i\_country\_sno), state\_sno = quote\_nullable(i\_state\_sno), name = quote\_nullable(i\_name), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| update\_city | UPDATE masters.city SET city\_sno = quote\_nullable(i\_city\_sno), country\_sno = quote\_nullable(i\_country\_sno), state\_sno = quote\_nullable(i\_state\_sno), name = quote\_nullable(i\_name), active\_flag = quote\_nullable(i\_active\_flag) WHERE city\_sno = quote\_nullable(i\_city\_sno); |
| delete\_city\_by\_id | DELETE FROM masters.city WHERE city\_sno = quote\_nullable(i\_city\_sno); |
| unique\_city\_exists | SELECT 1 FROM masters.city WHERE <unique\_conditions> LIMIT 1; |
| custom\_search\_city\_by\_id | SELECT t.city\_sno, t.country\_sno, t.state\_sno, t.name, t.active\_flag, f\_country.name, f\_state.name FROM masters.city t LEFT JOIN masters.country f\_country ON t.country\_sno = f\_country.country\_sno LEFT JOIN masters.state f\_state ON t.state\_sno = f\_state.state\_sno WHERE <custom\_search\_conditions>; |
| get\_all\_citys | SELECT t.city\_sno, t.country\_sno, t.state\_sno, t.name, t.active\_flag, f\_country.name, f\_state.name FROM masters.city t LEFT JOIN masters.country f\_country ON t.country\_sno = f\_country.country\_sno LEFT JOIN masters.state f\_state ON t.state\_sno = f\_state.state\_sno ; |
| get\_city\_by\_id | SELECT t.city\_sno, t.country\_sno, t.state\_sno, t.name, t.active\_flag, f\_country.name, f\_state.name FROM masters.city t LEFT JOIN masters.country f\_country ON t.country\_sno = f\_country.country\_sno LEFT JOIN masters.state f\_state ON t.state\_sno = f\_state.state\_sno WHERE t.city\_sno = quote\_nullable(i\_city\_sno); |
| custom\_delete\_city\_by\_id | DELETE FROM masters.city WHERE <custom\_conditions>; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Area**

### **Database Table Name: area**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| area\_sno | bigint - (Numeric Precision: 64) | area\_pkey (PRIMARY KEY), |
| country\_sno | smallint - (Numeric Precision: 16) | area\_country\_sno\_fkey (FOREIGN KEY), |
| state\_sno | smallint - (Numeric Precision: 16) | area\_state\_sno\_fkey (FOREIGN KEY), |
| city\_sno | integer | area\_city\_sno\_fkey (FOREIGN KEY), |
| name | text | null |
| pincode | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Area Sno | hidden | number | true | Manual Entry |
| Country Sno | listbox | number | true | API feed list box selection |
| State Sno | listbox | number | true | API feed list box selection |
| City Sno | listbox | number | true | API feed list box selection |
| Name | textbox | string | true | Manual Entry |
| Pincode | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_areas | get | /api/v1/get\_areas |
| create\_area | post | /api/v1/create\_area |
| update\_area | put | /api/v1/update\_area |
| delete\_area | delete | /api/v1/delete\_area |
| get\_citys | get | /api/v1/get\_citys |
| get\_countrys | get | /api/v1/get\_countrys |
| get\_states | get | /api/v1/get\_states |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_areas | {} | {"data":{"pincode":"text","area\_sno":"bigint","state\_sno":"smallint","name":"text","country\_sno":"smallint","city\_sno":"integer","active\_flag":"boolean"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_area | {"data":{"pincode":"text","area\_sno":"bigint","state\_sno":"smallint","name":"text","country\_sno":"smallint","city\_sno":"integer","active\_flag":"boolean"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_area | {"data":{"pincode":"text","area\_sno":"bigint","state\_sno":"smallint","name":"text","country\_sno":"smallint","city\_sno":"integer","active\_flag":"boolean"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_area | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_citys | {"id":"integer"} | {"data":"array of city"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_countrys | {"id":"integer"} | {"data":"array of country"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_states | {"id":"integer"} | {"data":"array of state"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| delete\_area\_by\_id | DELETE FROM masters.area WHERE area\_sno = quote\_nullable(i\_area\_sno); |
| unique\_area\_exists | SELECT 1 FROM masters.area WHERE <unique\_conditions> LIMIT 1; |
| get\_all\_areas | SELECT t.area\_sno, t.country\_sno, t.state\_sno, t.city\_sno, t.name, t.pincode, t.active\_flag, f\_city.name, f\_country.name, f\_state.name FROM masters.area t LEFT JOIN masters.city f\_city ON t.city\_sno = f\_city.city\_sno LEFT JOIN masters.country f\_country ON t.country\_sno = f\_country.country\_sno LEFT JOIN masters.state f\_state ON t.state\_sno = f\_state.state\_sno ; |
| custom\_search\_area\_by\_id | SELECT t.area\_sno, t.country\_sno, t.state\_sno, t.city\_sno, t.name, t.pincode, t.active\_flag, f\_city.name, f\_country.name, f\_state.name FROM masters.area t LEFT JOIN masters.city f\_city ON t.city\_sno = f\_city.city\_sno LEFT JOIN masters.country f\_country ON t.country\_sno = f\_country.country\_sno LEFT JOIN masters.state f\_state ON t.state\_sno = f\_state.state\_sno WHERE <custom\_search\_conditions>; |
| update\_area | UPDATE masters.area SET area\_sno = quote\_nullable(i\_area\_sno), country\_sno = quote\_nullable(i\_country\_sno), state\_sno = quote\_nullable(i\_state\_sno), city\_sno = quote\_nullable(i\_city\_sno), name = quote\_nullable(i\_name), pincode = quote\_nullable(i\_pincode), active\_flag = quote\_nullable(i\_active\_flag) WHERE area\_sno = quote\_nullable(i\_area\_sno); |
| create\_area | INSERT INTO masters.area (t.area\_sno, t.country\_sno, t.state\_sno, t.city\_sno, t.name, t.pincode, t.active\_flag) VALUES (i\_area\_sno, i\_country\_sno, i\_state\_sno, i\_city\_sno, i\_name, i\_pincode, i\_active\_flag); |
| custom\_update\_area | UPDATE masters.area SET area\_sno = quote\_nullable(i\_area\_sno), country\_sno = quote\_nullable(i\_country\_sno), state\_sno = quote\_nullable(i\_state\_sno), city\_sno = quote\_nullable(i\_city\_sno), name = quote\_nullable(i\_name), pincode = quote\_nullable(i\_pincode), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| get\_area\_by\_id | SELECT t.area\_sno, t.country\_sno, t.state\_sno, t.city\_sno, t.name, t.pincode, t.active\_flag, f\_city.name, f\_country.name, f\_state.name FROM masters.area t LEFT JOIN masters.city f\_city ON t.city\_sno = f\_city.city\_sno LEFT JOIN masters.country f\_country ON t.country\_sno = f\_country.country\_sno LEFT JOIN masters.state f\_state ON t.state\_sno = f\_state.state\_sno WHERE t.area\_sno = quote\_nullable(i\_area\_sno); |
| custom\_delete\_area\_by\_id | DELETE FROM masters.area WHERE <custom\_conditions>; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Country**

### **Database Table Name: country**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| country\_sno | smallint - (Numeric Precision: 16) | country\_pkey (PRIMARY KEY), |
| name | text | null |
| iso2 | character varying | null |
| phone\_code | character varying | null |
| numeric\_code | character varying | null |
| currency\_code | character varying | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Country Sno | hidden | number | true | Manual Entry |
| Name | textbox | string | true | Manual Entry |
| Iso2 | textbox | string | true | Manual Entry |
| Phone Code | textbox | string | true | Manual Entry |
| Numeric Code | textbox | string | false | Manual Entry |
| Currency Code | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_countrys | get | /api/v1/get\_countrys |
| create\_country | post | /api/v1/create\_country |
| update\_country | put | /api/v1/update\_country |
| delete\_country | delete | /api/v1/delete\_country |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_countrys | {} | {"data":{"numeric\_code":"character varying","name":"text","country\_sno":"smallint","active\_flag":"boolean","iso2":"character varying","currency\_code":"character varying","phone\_code":"character varying"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_country | {"data":{"numeric\_code":"character varying","name":"text","country\_sno":"smallint","active\_flag":"boolean","iso2":"character varying","currency\_code":"character varying","phone\_code":"character varying"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_country | {"data":{"numeric\_code":"character varying","name":"text","country\_sno":"smallint","active\_flag":"boolean","iso2":"character varying","currency\_code":"character varying","phone\_code":"character varying"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_country | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| unique\_country\_exists | SELECT 1 FROM masters.country WHERE <unique\_conditions> LIMIT 1; |
| delete\_country\_by\_id | DELETE FROM masters.country WHERE country\_sno = quote\_nullable(i\_country\_sno); |
| custom\_update\_country | UPDATE masters.country SET country\_sno = quote\_nullable(i\_country\_sno), name = quote\_nullable(i\_name), iso2 = quote\_nullable(i\_iso2), phone\_code = quote\_nullable(i\_phone\_code), numeric\_code = quote\_nullable(i\_numeric\_code), currency\_code = quote\_nullable(i\_currency\_code), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| custom\_search\_country\_by\_id | SELECT t.country\_sno, t.name, t.iso2, t.phone\_code, t.numeric\_code, t.currency\_code, t.active\_flag FROM masters.country t WHERE <custom\_search\_conditions>; |
| get\_country\_by\_id | SELECT t.country\_sno, t.name, t.iso2, t.phone\_code, t.numeric\_code, t.currency\_code, t.active\_flag FROM masters.country t WHERE t.country\_sno = quote\_nullable(i\_country\_sno); |
| create\_country | INSERT INTO masters.country (t.country\_sno, t.name, t.iso2, t.phone\_code, t.numeric\_code, t.currency\_code, t.active\_flag) VALUES (i\_country\_sno, i\_name, i\_iso2, i\_phone\_code, i\_numeric\_code, i\_currency\_code, i\_active\_flag); |
| update\_country | UPDATE masters.country SET country\_sno = quote\_nullable(i\_country\_sno), name = quote\_nullable(i\_name), iso2 = quote\_nullable(i\_iso2), phone\_code = quote\_nullable(i\_phone\_code), numeric\_code = quote\_nullable(i\_numeric\_code), currency\_code = quote\_nullable(i\_currency\_code), active\_flag = quote\_nullable(i\_active\_flag) WHERE country\_sno = quote\_nullable(i\_country\_sno); |
| get\_all\_countrys | SELECT t.country\_sno, t.name, t.iso2, t.phone\_code, t.numeric\_code, t.currency\_code, t.active\_flag FROM masters.country t ; |
| custom\_delete\_country\_by\_id | DELETE FROM masters.country WHERE <custom\_conditions>; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: State**

### **Database Table Name: state**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| state\_sno | smallint - (Numeric Precision: 16) | state\_pkey (PRIMARY KEY), |
| country\_sno | smallint - (Numeric Precision: 16) | state\_country\_sno\_fkey (FOREIGN KEY), |
| state\_code | character varying | null |
| name | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| State Sno | hidden | number | true | Manual Entry |
| Country Sno | listbox | number | true | API feed list box selection |
| State Code | textbox | string | false | Manual Entry |
| Name | textbox | string | true | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_states | get | /api/v1/get\_states |
| create\_state | post | /api/v1/create\_state |
| update\_state | put | /api/v1/update\_state |
| delete\_state | delete | /api/v1/delete\_state |
| get\_countrys | get | /api/v1/get\_countrys |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_states | {} | {"data":{"state\_sno":"smallint","name":"text","country\_sno":"smallint","active\_flag":"boolean","state\_code":"character varying"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_state | {"data":{"state\_sno":"smallint","name":"text","country\_sno":"smallint","active\_flag":"boolean","state\_code":"character varying"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_state | {"data":{"state\_sno":"smallint","name":"text","country\_sno":"smallint","active\_flag":"boolean","state\_code":"character varying"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_state | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_countrys | {"id":"integer"} | {"data":"array of country"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| get\_state\_by\_id | SELECT t.state\_sno, t.country\_sno, t.state\_code, t.name, t.active\_flag, f\_country.name FROM masters.state t LEFT JOIN masters.country f\_country ON t.country\_sno = f\_country.country\_sno WHERE t.state\_sno = quote\_nullable(i\_state\_sno); |
| delete\_state\_by\_id | DELETE FROM masters.state WHERE state\_sno = quote\_nullable(i\_state\_sno); |
| get\_all\_states | SELECT t.state\_sno, t.country\_sno, t.state\_code, t.name, t.active\_flag, f\_country.name FROM masters.state t LEFT JOIN masters.country f\_country ON t.country\_sno = f\_country.country\_sno ; |
| custom\_update\_state | UPDATE masters.state SET state\_sno = quote\_nullable(i\_state\_sno), country\_sno = quote\_nullable(i\_country\_sno), state\_code = quote\_nullable(i\_state\_code), name = quote\_nullable(i\_name), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| custom\_delete\_state\_by\_id | DELETE FROM masters.state WHERE <custom\_conditions>; |
| create\_state | INSERT INTO masters.state (t.state\_sno, t.country\_sno, t.state\_code, t.name, t.active\_flag) VALUES (i\_state\_sno, i\_country\_sno, i\_state\_code, i\_name, i\_active\_flag); |
| update\_state | UPDATE masters.state SET state\_sno = quote\_nullable(i\_state\_sno), country\_sno = quote\_nullable(i\_country\_sno), state\_code = quote\_nullable(i\_state\_code), name = quote\_nullable(i\_name), active\_flag = quote\_nullable(i\_active\_flag) WHERE state\_sno = quote\_nullable(i\_state\_sno); |
| custom\_search\_state\_by\_id | SELECT t.state\_sno, t.country\_sno, t.state\_code, t.name, t.active\_flag, f\_country.name FROM masters.state t LEFT JOIN masters.country f\_country ON t.country\_sno = f\_country.country\_sno WHERE <custom\_search\_conditions>; |
| unique\_state\_exists | SELECT 1 FROM masters.state WHERE <unique\_conditions> LIMIT 1; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Attendance Records**

### **Database Table Name: attendance\_records**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| attendance\_sno | bigint - (Numeric Precision: 64) | attendance\_records\_pkey (PRIMARY KEY), |
| auth\_user\_id | bigint - (Numeric Precision: 64) | attendance\_records\_auth\_user\_id\_fkey (FOREIGN KEY), |
| date | date | null |
| status | text | null |
| check\_in\_time | time without time zone | null |
| check\_out\_time | time without time zone | null |
| remarks | text | null |
| barcode\_value | text | null |
| scanned\_by | bigint - (Numeric Precision: 64) | null |
| scanned\_at | timestamp without time zone | null |
| active\_flag | boolean | null |
| qbox\_entity\_sno | bigint - (Numeric Precision: 64) | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Attendance Sno | hidden | number | true | Manual Entry |
| Auth User Id | listbox | number | false | API feed list box selection |
| Date | timestamp | string | true | Manual Entry |
| Status | textbox | string | true | Manual Entry |
| Check In Time | textbox | unknown | false | Manual Entry |
| Check Out Time | textbox | unknown | false | Manual Entry |
| Remarks | textbox | string | false | Manual Entry |
| Barcode Value | textbox | string | false | Manual Entry |
| Scanned By | textbox\_number | number | false | Manual Entry |
| Scanned At | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |
| Qbox Entity Sno | textbox\_number | number | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_attendance\_recordss | get | /api/v1/get\_attendance\_recordss |
| create\_attendance\_records | post | /api/v1/create\_attendance\_records |
| update\_attendance\_records | put | /api/v1/update\_attendance\_records |
| delete\_attendance\_records | delete | /api/v1/delete\_attendance\_records |
| get\_auth\_users | get | /api/v1/get\_auth\_users |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_attendance\_recordss | {} | {"data":{"date":"date","scanned\_by":"bigint","barcode\_value":"text","scanned\_at":"timestamp without time zone","check\_in\_time":"time without time zone","check\_out\_time":"time without time zone","attendance\_sno":"bigint","qbox\_entity\_sno":"bigint","active\_flag":"boolean","auth\_user\_id":"bigint","remarks":"text","status":"text"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_attendance\_records | {"data":{"date":"date","scanned\_by":"bigint","barcode\_value":"text","scanned\_at":"timestamp without time zone","check\_in\_time":"time without time zone","check\_out\_time":"time without time zone","attendance\_sno":"bigint","qbox\_entity\_sno":"bigint","active\_flag":"boolean","auth\_user\_id":"bigint","remarks":"text","status":"text"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_attendance\_records | {"data":{"date":"date","scanned\_by":"bigint","barcode\_value":"text","scanned\_at":"timestamp without time zone","check\_in\_time":"time without time zone","check\_out\_time":"time without time zone","attendance\_sno":"bigint","qbox\_entity\_sno":"bigint","active\_flag":"boolean","auth\_user\_id":"bigint","remarks":"text","status":"text"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_attendance\_records | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_auth\_users | {"id":"integer"} | {"data":"array of auth\_user"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| update\_attendance\_records | UPDATE masters.attendance\_records SET attendance\_sno = quote\_nullable(i\_attendance\_sno), auth\_user\_id = quote\_nullable(i\_auth\_user\_id), date = quote\_nullable(i\_date), status = quote\_nullable(i\_status), check\_in\_time = quote\_nullable(i\_check\_in\_time), check\_out\_time = quote\_nullable(i\_check\_out\_time), remarks = quote\_nullable(i\_remarks), barcode\_value = quote\_nullable(i\_barcode\_value), scanned\_by = quote\_nullable(i\_scanned\_by), scanned\_at = quote\_nullable(i\_scanned\_at), active\_flag = quote\_nullable(i\_active\_flag), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno) WHERE attendance\_sno = quote\_nullable(i\_attendance\_sno); |
| custom\_search\_attendance\_records\_by\_id | SELECT t.attendance\_sno, t.auth\_user\_id, t.date, t.status, t.check\_in\_time, t.check\_out\_time, t.remarks, t.barcode\_value, t.scanned\_by, t.scanned\_at, t.active\_flag, t.qbox\_entity\_sno FROM masters.attendance\_records t WHERE <custom\_search\_conditions>; |
| get\_attendance\_records\_by\_id | SELECT t.attendance\_sno, t.auth\_user\_id, t.date, t.status, t.check\_in\_time, t.check\_out\_time, t.remarks, t.barcode\_value, t.scanned\_by, t.scanned\_at, t.active\_flag, t.qbox\_entity\_sno FROM masters.attendance\_records t WHERE t.attendance\_sno = quote\_nullable(i\_attendance\_sno); |
| custom\_delete\_attendance\_records\_by\_id | DELETE FROM masters.attendance\_records WHERE <custom\_conditions>; |
| delete\_attendance\_records\_by\_id | DELETE FROM masters.attendance\_records WHERE attendance\_sno = quote\_nullable(i\_attendance\_sno); |
| custom\_update\_attendance\_records | UPDATE masters.attendance\_records SET attendance\_sno = quote\_nullable(i\_attendance\_sno), auth\_user\_id = quote\_nullable(i\_auth\_user\_id), date = quote\_nullable(i\_date), status = quote\_nullable(i\_status), check\_in\_time = quote\_nullable(i\_check\_in\_time), check\_out\_time = quote\_nullable(i\_check\_out\_time), remarks = quote\_nullable(i\_remarks), barcode\_value = quote\_nullable(i\_barcode\_value), scanned\_by = quote\_nullable(i\_scanned\_by), scanned\_at = quote\_nullable(i\_scanned\_at), active\_flag = quote\_nullable(i\_active\_flag), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno) WHERE <custom\_conditions>; |
| unique\_attendance\_records\_exists | SELECT 1 FROM masters.attendance\_records WHERE <unique\_conditions> LIMIT 1; |
| create\_attendance\_records | INSERT INTO masters.attendance\_records (t.attendance\_sno, t.auth\_user\_id, t.date, t.status, t.check\_in\_time, t.check\_out\_time, t.remarks, t.barcode\_value, t.scanned\_by, t.scanned\_at, t.active\_flag, t.qbox\_entity\_sno) VALUES (i\_attendance\_sno, i\_auth\_user\_id, i\_date, i\_status, i\_check\_in\_time, i\_check\_out\_time, i\_remarks, i\_barcode\_value, i\_scanned\_by, i\_scanned\_at, i\_active\_flag, i\_qbox\_entity\_sno); |
| get\_all\_attendance\_recordss | SELECT t.attendance\_sno, t.auth\_user\_id, t.date, t.status, t.check\_in\_time, t.check\_out\_time, t.remarks, t.barcode\_value, t.scanned\_by, t.scanned\_at, t.active\_flag, t.qbox\_entity\_sno FROM masters.attendance\_records t ; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Entity Infra**

### **Database Table Name: entity\_infra**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| entity\_infra\_sno | bigint - (Numeric Precision: 64) | entity\_infra\_pkey (PRIMARY KEY), |
| qbox\_entity\_sno | bigint - (Numeric Precision: 64) | entity\_infra\_qbox\_entity\_sno\_fkey (FOREIGN KEY), |
| infra\_sno | bigint - (Numeric Precision: 64) | entity\_infra\_infra\_sno\_fkey (FOREIGN KEY), |
| description | text | null |
| active\_flag | boolean | null |
| created\_on | timestamp without time zone | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Entity Infra Sno | hidden | number | true | Manual Entry |
| Qbox Entity Sno | listbox | number | true | API feed list box selection |
| Infra Sno | listbox | number | true | API feed list box selection |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |
| Created On | textbox | string | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_entity\_infras | get | /api/v1/get\_entity\_infras |
| create\_entity\_infra | post | /api/v1/create\_entity\_infra |
| update\_entity\_infra | put | /api/v1/update\_entity\_infra |
| delete\_entity\_infra | delete | /api/v1/delete\_entity\_infra |
| get\_infras | get | /api/v1/get\_infras |
| get\_qbox\_entitys | get | /api/v1/get\_qbox\_entitys |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_entity\_infras | {} | {"data":{"created\_on":"timestamp without time zone","description":"text","qbox\_entity\_sno":"bigint","active\_flag":"boolean","entity\_infra\_sno":"bigint","infra\_sno":"bigint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_entity\_infra | {"data":{"created\_on":"timestamp without time zone","description":"text","qbox\_entity\_sno":"bigint","active\_flag":"boolean","entity\_infra\_sno":"bigint","infra\_sno":"bigint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_entity\_infra | {"data":{"created\_on":"timestamp without time zone","description":"text","qbox\_entity\_sno":"bigint","active\_flag":"boolean","entity\_infra\_sno":"bigint","infra\_sno":"bigint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_entity\_infra | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_infras | {"id":"integer"} | {"data":"array of infra"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_qbox\_entitys | {"id":"integer"} | {"data":"array of qbox\_entity"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| create\_entity\_infra | INSERT INTO masters.entity\_infra (t.entity\_infra\_sno, t.qbox\_entity\_sno, t.infra\_sno, t.description, t.active\_flag, t.created\_on) VALUES (i\_entity\_infra\_sno, i\_qbox\_entity\_sno, i\_infra\_sno, i\_description, i\_active\_flag, i\_created\_on); |
| custom\_update\_entity\_infra | UPDATE masters.entity\_infra SET entity\_infra\_sno = quote\_nullable(i\_entity\_infra\_sno), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), infra\_sno = quote\_nullable(i\_infra\_sno), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), created\_on = quote\_nullable(i\_created\_on) WHERE <custom\_conditions>; |
| unique\_entity\_infra\_exists | SELECT 1 FROM masters.entity\_infra WHERE <unique\_conditions> LIMIT 1; |
| update\_entity\_infra | UPDATE masters.entity\_infra SET entity\_infra\_sno = quote\_nullable(i\_entity\_infra\_sno), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), infra\_sno = quote\_nullable(i\_infra\_sno), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), created\_on = quote\_nullable(i\_created\_on) WHERE entity\_infra\_sno = quote\_nullable(i\_entity\_infra\_sno); |
| delete\_entity\_infra\_by\_id | DELETE FROM masters.entity\_infra WHERE entity\_infra\_sno = quote\_nullable(i\_entity\_infra\_sno); |
| get\_all\_entity\_infras | SELECT t.entity\_infra\_sno, t.qbox\_entity\_sno, t.infra\_sno, t.description, t.active\_flag, t.created\_on, f\_infra.name, f\_qbox\_entity.name FROM masters.entity\_infra t LEFT JOIN masters.infra f\_infra ON t.infra\_sno = f\_infra.infra\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno ; |
| custom\_delete\_entity\_infra\_by\_id | DELETE FROM masters.entity\_infra WHERE <custom\_conditions>; |
| get\_entity\_infra\_by\_id | SELECT t.entity\_infra\_sno, t.qbox\_entity\_sno, t.infra\_sno, t.description, t.active\_flag, t.created\_on, f\_infra.name, f\_qbox\_entity.name FROM masters.entity\_infra t LEFT JOIN masters.infra f\_infra ON t.infra\_sno = f\_infra.infra\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno WHERE t.entity\_infra\_sno = quote\_nullable(i\_entity\_infra\_sno); |
| custom\_search\_entity\_infra\_by\_id | SELECT t.entity\_infra\_sno, t.qbox\_entity\_sno, t.infra\_sno, t.description, t.active\_flag, t.created\_on, f\_infra.name, f\_qbox\_entity.name FROM masters.entity\_infra t LEFT JOIN masters.infra f\_infra ON t.infra\_sno = f\_infra.infra\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno WHERE <custom\_search\_conditions>; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Box Cell**

### **Database Table Name: box\_cell**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| box\_cell\_sno | bigint - (Numeric Precision: 64) | box\_cell\_pkey (PRIMARY KEY), |
| qbox\_entity\_sno | bigint - (Numeric Precision: 64) | box\_cell\_qbox\_entity\_sno\_fkey (FOREIGN KEY), |
| entity\_infra\_sno | bigint - (Numeric Precision: 64) | box\_cell\_entity\_infra\_sno\_fkey (FOREIGN KEY), |
| row\_no | smallint - (Numeric Precision: 16) | null |
| column\_no | smallint - (Numeric Precision: 16) | null |
| box\_cell\_status\_cd | smallint - (Numeric Precision: 16) | null |
| description | text | null |
| active\_flag | boolean | null |
| open\_hex | character varying | null |
| close\_hex | character varying | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Box Cell Sno | hidden | number | true | Manual Entry |
| Qbox Entity Sno | listbox | number | true | API feed list box selection |
| Entity Infra Sno | listbox | number | true | API feed list box selection |
| Row No | textbox\_number | number | true | Manual Entry |
| Column No | textbox\_number | number | true | Manual Entry |
| Box Cell Status Cd | textbox\_number | number | true | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |
| Open Hex | textbox | string | false | Manual Entry |
| Close Hex | textbox | string | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_box\_cells | get | /api/v1/get\_box\_cells |
| create\_box\_cell | post | /api/v1/create\_box\_cell |
| update\_box\_cell | put | /api/v1/update\_box\_cell |
| delete\_box\_cell | delete | /api/v1/delete\_box\_cell |
| get\_entity\_infras | get | /api/v1/get\_entity\_infras |
| get\_qbox\_entitys | get | /api/v1/get\_qbox\_entitys |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_box\_cells | {} | {"data":{"open\_hex":"character varying","close\_hex":"character varying","description":"text","qbox\_entity\_sno":"bigint","box\_cell\_status\_cd":"smallint","row\_no":"smallint","active\_flag":"boolean","box\_cell\_sno":"bigint","entity\_infra\_sno":"bigint","column\_no":"smallint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_box\_cell | {"data":{"open\_hex":"character varying","close\_hex":"character varying","description":"text","qbox\_entity\_sno":"bigint","box\_cell\_status\_cd":"smallint","row\_no":"smallint","active\_flag":"boolean","box\_cell\_sno":"bigint","entity\_infra\_sno":"bigint","column\_no":"smallint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_box\_cell | {"data":{"open\_hex":"character varying","close\_hex":"character varying","description":"text","qbox\_entity\_sno":"bigint","box\_cell\_status\_cd":"smallint","row\_no":"smallint","active\_flag":"boolean","box\_cell\_sno":"bigint","entity\_infra\_sno":"bigint","column\_no":"smallint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_box\_cell | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_entity\_infras | {"id":"integer"} | {"data":"array of entity\_infra"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_qbox\_entitys | {"id":"integer"} | {"data":"array of qbox\_entity"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| custom\_search\_box\_cell\_by\_id | SELECT t.box\_cell\_sno, t.qbox\_entity\_sno, t.entity\_infra\_sno, t.row\_no, t.column\_no, t.box\_cell\_status\_cd, t.description, t.active\_flag, t.open\_hex, t.close\_hex, f\_qbox\_entity.name FROM masters.box\_cell t LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno WHERE <custom\_search\_conditions>; |
| get\_all\_box\_cells | SELECT t.box\_cell\_sno, t.qbox\_entity\_sno, t.entity\_infra\_sno, t.row\_no, t.column\_no, t.box\_cell\_status\_cd, t.description, t.active\_flag, t.open\_hex, t.close\_hex, f\_qbox\_entity.name FROM masters.box\_cell t LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno ; |
| get\_box\_cell\_by\_id | SELECT t.box\_cell\_sno, t.qbox\_entity\_sno, t.entity\_infra\_sno, t.row\_no, t.column\_no, t.box\_cell\_status\_cd, t.description, t.active\_flag, t.open\_hex, t.close\_hex, f\_qbox\_entity.name FROM masters.box\_cell t LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno WHERE t.box\_cell\_sno = quote\_nullable(i\_box\_cell\_sno); |
| custom\_delete\_box\_cell\_by\_id | DELETE FROM masters.box\_cell WHERE <custom\_conditions>; |
| unique\_box\_cell\_exists | SELECT 1 FROM masters.box\_cell WHERE <unique\_conditions> LIMIT 1; |
| delete\_box\_cell\_by\_id | DELETE FROM masters.box\_cell WHERE box\_cell\_sno = quote\_nullable(i\_box\_cell\_sno); |
| create\_box\_cell | INSERT INTO masters.box\_cell (t.box\_cell\_sno, t.qbox\_entity\_sno, t.entity\_infra\_sno, t.row\_no, t.column\_no, t.box\_cell\_status\_cd, t.description, t.active\_flag, t.open\_hex, t.close\_hex) VALUES (i\_box\_cell\_sno, i\_qbox\_entity\_sno, i\_entity\_infra\_sno, i\_row\_no, i\_column\_no, i\_box\_cell\_status\_cd, i\_description, i\_active\_flag, i\_open\_hex, i\_close\_hex); |
| custom\_update\_box\_cell | UPDATE masters.box\_cell SET box\_cell\_sno = quote\_nullable(i\_box\_cell\_sno), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), entity\_infra\_sno = quote\_nullable(i\_entity\_infra\_sno), row\_no = quote\_nullable(i\_row\_no), column\_no = quote\_nullable(i\_column\_no), box\_cell\_status\_cd = quote\_nullable(i\_box\_cell\_status\_cd), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), open\_hex = quote\_nullable(i\_open\_hex), close\_hex = quote\_nullable(i\_close\_hex) WHERE <custom\_conditions>; |
| update\_box\_cell | UPDATE masters.box\_cell SET box\_cell\_sno = quote\_nullable(i\_box\_cell\_sno), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), entity\_infra\_sno = quote\_nullable(i\_entity\_infra\_sno), row\_no = quote\_nullable(i\_row\_no), column\_no = quote\_nullable(i\_column\_no), box\_cell\_status\_cd = quote\_nullable(i\_box\_cell\_status\_cd), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), open\_hex = quote\_nullable(i\_open\_hex), close\_hex = quote\_nullable(i\_close\_hex) WHERE box\_cell\_sno = quote\_nullable(i\_box\_cell\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Box Cell Food**

### **Database Table Name: box\_cell\_food**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| box\_cell\_food\_sno | bigint - (Numeric Precision: 64) | box\_cell\_food\_pkey (PRIMARY KEY), |
| box\_cell\_sno | smallint - (Numeric Precision: 16) | box\_cell\_food\_box\_cell\_sno\_fkey (FOREIGN KEY), |
| sku\_inventory\_sno | bigint - (Numeric Precision: 64) | box\_cell\_food\_sku\_inventory\_sno\_fkey (FOREIGN KEY), |
| entry\_time | timestamp without time zone | null |
| qbox\_entity\_sno | bigint - (Numeric Precision: 64) | box\_cell\_food\_qbox\_entity\_sno\_fkey (FOREIGN KEY), |
| active | boolean | null |
| description | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Box Cell Food Sno | hidden | number | true | Manual Entry |
| Box Cell Sno | listbox | number | true | API feed list box selection |
| Sku Inventory Sno | listbox | number | true | API feed list box selection |
| Entry Time | textbox | string | true | Manual Entry |
| Qbox Entity Sno | listbox | number | true | API feed list box selection |
| Active | checkbox | boolean | false | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_box\_cell\_foods | get | /api/v1/get\_box\_cell\_foods |
| create\_box\_cell\_food | post | /api/v1/create\_box\_cell\_food |
| update\_box\_cell\_food | put | /api/v1/update\_box\_cell\_food |
| delete\_box\_cell\_food | delete | /api/v1/delete\_box\_cell\_food |
| get\_box\_cells | get | /api/v1/get\_box\_cells |
| get\_qbox\_entitys | get | /api/v1/get\_qbox\_entitys |
| get\_sku\_inventorys | get | /api/v1/get\_sku\_inventorys |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_box\_cell\_foods | {} | {"data":{"entry\_time":"timestamp without time zone","box\_cell\_food\_sno":"bigint","active":"boolean","description":"text","qbox\_entity\_sno":"bigint","active\_flag":"boolean","box\_cell\_sno":"smallint","sku\_inventory\_sno":"bigint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_box\_cell\_food | {"data":{"entry\_time":"timestamp without time zone","box\_cell\_food\_sno":"bigint","active":"boolean","description":"text","qbox\_entity\_sno":"bigint","active\_flag":"boolean","box\_cell\_sno":"smallint","sku\_inventory\_sno":"bigint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_box\_cell\_food | {"data":{"entry\_time":"timestamp without time zone","box\_cell\_food\_sno":"bigint","active":"boolean","description":"text","qbox\_entity\_sno":"bigint","active\_flag":"boolean","box\_cell\_sno":"smallint","sku\_inventory\_sno":"bigint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_box\_cell\_food | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_box\_cells | {"id":"integer"} | {"data":"array of box\_cell"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_qbox\_entitys | {"id":"integer"} | {"data":"array of qbox\_entity"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_sku\_inventorys | {"id":"integer"} | {"data":"array of sku\_inventory"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| delete\_box\_cell\_food\_by\_id | DELETE FROM masters.box\_cell\_food WHERE box\_cell\_food\_sno = quote\_nullable(i\_box\_cell\_food\_sno); |
| custom\_search\_box\_cell\_food\_by\_id | SELECT t.box\_cell\_food\_sno, t.box\_cell\_sno, t.sku\_inventory\_sno, t.entry\_time, t.qbox\_entity\_sno, t.active, t.description, t.active\_flag, f\_qbox\_entity.name FROM masters.box\_cell\_food t LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno WHERE <custom\_search\_conditions>; |
| create\_box\_cell\_food | INSERT INTO masters.box\_cell\_food (t.box\_cell\_food\_sno, t.box\_cell\_sno, t.sku\_inventory\_sno, t.entry\_time, t.qbox\_entity\_sno, t.active, t.description, t.active\_flag) VALUES (i\_box\_cell\_food\_sno, i\_box\_cell\_sno, i\_sku\_inventory\_sno, i\_entry\_time, i\_qbox\_entity\_sno, i\_active, i\_description, i\_active\_flag); |
| get\_all\_box\_cell\_foods | SELECT t.box\_cell\_food\_sno, t.box\_cell\_sno, t.sku\_inventory\_sno, t.entry\_time, t.qbox\_entity\_sno, t.active, t.description, t.active\_flag, f\_qbox\_entity.name FROM masters.box\_cell\_food t LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno ; |
| unique\_box\_cell\_food\_exists | SELECT 1 FROM masters.box\_cell\_food WHERE <unique\_conditions> LIMIT 1; |
| update\_box\_cell\_food | UPDATE masters.box\_cell\_food SET box\_cell\_food\_sno = quote\_nullable(i\_box\_cell\_food\_sno), box\_cell\_sno = quote\_nullable(i\_box\_cell\_sno), sku\_inventory\_sno = quote\_nullable(i\_sku\_inventory\_sno), entry\_time = quote\_nullable(i\_entry\_time), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), active = quote\_nullable(i\_active), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE box\_cell\_food\_sno = quote\_nullable(i\_box\_cell\_food\_sno); |
| custom\_update\_box\_cell\_food | UPDATE masters.box\_cell\_food SET box\_cell\_food\_sno = quote\_nullable(i\_box\_cell\_food\_sno), box\_cell\_sno = quote\_nullable(i\_box\_cell\_sno), sku\_inventory\_sno = quote\_nullable(i\_sku\_inventory\_sno), entry\_time = quote\_nullable(i\_entry\_time), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), active = quote\_nullable(i\_active), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| custom\_delete\_box\_cell\_food\_by\_id | DELETE FROM masters.box\_cell\_food WHERE <custom\_conditions>; |
| get\_box\_cell\_food\_by\_id | SELECT t.box\_cell\_food\_sno, t.box\_cell\_sno, t.sku\_inventory\_sno, t.entry\_time, t.qbox\_entity\_sno, t.active, t.description, t.active\_flag, f\_qbox\_entity.name FROM masters.box\_cell\_food t LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno WHERE t.box\_cell\_food\_sno = quote\_nullable(i\_box\_cell\_food\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Qbox Entity**

### **Database Table Name: qbox\_entity**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| qbox\_entity\_sno | bigint - (Numeric Precision: 64) | qbox\_entity\_pkey (PRIMARY KEY), |
| name | text | null |
| address\_sno | bigint - (Numeric Precision: 64) | qbox\_entity\_address\_sno\_fkey (FOREIGN KEY), |
| area\_sno | bigint - (Numeric Precision: 64) | qbox\_entity\_area\_sno\_fkey (FOREIGN KEY), |
| qbox\_entity\_status\_cd | smallint - (Numeric Precision: 16) | qbox\_entity\_qbox\_entity\_status\_cd\_fkey (FOREIGN KEY), |
| created\_on | timestamp without time zone | null |
| entity\_code | character varying | null |
| active\_flag | boolean | null |
| state\_sno | integer | null |
| city\_sno | integer | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Qbox Entity Sno | hidden | number | true | Manual Entry |
| Name | textbox | string | true | Manual Entry |
| Address Sno | listbox | number | false | API feed list box selection |
| Area Sno | listbox | number | true | API feed list box selection |
| Qbox Entity Status Cd | listbox | number | true | API feed list box selection |
| Created On | textbox | string | false | Manual Entry |
| Entity Code | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |
| State Sno | textbox | number | false | Manual Entry |
| City Sno | textbox | number | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_qbox\_entitys | get | /api/v1/get\_qbox\_entitys |
| create\_qbox\_entity | post | /api/v1/create\_qbox\_entity |
| update\_qbox\_entity | put | /api/v1/update\_qbox\_entity |
| delete\_qbox\_entity | delete | /api/v1/delete\_qbox\_entity |
| get\_addresss | get | /api/v1/get\_addresss |
| get\_areas | get | /api/v1/get\_areas |
| get\_codes\_dtls | get | /api/v1/get\_codes\_dtls |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_qbox\_entitys | {} | {"data":{"address\_sno":"bigint","entity\_code":"character varying","created\_on":"timestamp without time zone","area\_sno":"bigint","state\_sno":"integer","name":"text","qbox\_entity\_sno":"bigint","city\_sno":"integer","active\_flag":"boolean","qbox\_entity\_status\_cd":"smallint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_qbox\_entity | {"data":{"address\_sno":"bigint","entity\_code":"character varying","created\_on":"timestamp without time zone","area\_sno":"bigint","state\_sno":"integer","name":"text","qbox\_entity\_sno":"bigint","city\_sno":"integer","active\_flag":"boolean","qbox\_entity\_status\_cd":"smallint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_qbox\_entity | {"data":{"address\_sno":"bigint","entity\_code":"character varying","created\_on":"timestamp without time zone","area\_sno":"bigint","state\_sno":"integer","name":"text","qbox\_entity\_sno":"bigint","city\_sno":"integer","active\_flag":"boolean","qbox\_entity\_status\_cd":"smallint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_qbox\_entity | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_addresss | {"id":"integer"} | {"data":"array of address"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_areas | {"id":"integer"} | {"data":"array of area"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_codes\_dtls | {"id":"integer"} | {"data":"array of codes\_dtl"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| get\_all\_qbox\_entitys | SELECT t.qbox\_entity\_sno, t.name, t.address\_sno, t.area\_sno, t.qbox\_entity\_status\_cd, t.created\_on, t.entity\_code, t.active\_flag, t.state\_sno, t.city\_sno, f\_area.name FROM masters.qbox\_entity t LEFT JOIN masters.area f\_area ON t.area\_sno = f\_area.area\_sno ; |
| custom\_search\_qbox\_entity\_by\_id | SELECT t.qbox\_entity\_sno, t.name, t.address\_sno, t.area\_sno, t.qbox\_entity\_status\_cd, t.created\_on, t.entity\_code, t.active\_flag, t.state\_sno, t.city\_sno, f\_area.name FROM masters.qbox\_entity t LEFT JOIN masters.area f\_area ON t.area\_sno = f\_area.area\_sno WHERE <custom\_search\_conditions>; |
| create\_qbox\_entity | INSERT INTO masters.qbox\_entity (t.qbox\_entity\_sno, t.name, t.address\_sno, t.area\_sno, t.qbox\_entity\_status\_cd, t.created\_on, t.entity\_code, t.active\_flag, t.state\_sno, t.city\_sno) VALUES (i\_qbox\_entity\_sno, i\_name, i\_address\_sno, i\_area\_sno, i\_qbox\_entity\_status\_cd, i\_created\_on, i\_entity\_code, i\_active\_flag, i\_state\_sno, i\_city\_sno); |
| unique\_qbox\_entity\_exists | SELECT 1 FROM masters.qbox\_entity WHERE <unique\_conditions> LIMIT 1; |
| custom\_update\_qbox\_entity | UPDATE masters.qbox\_entity SET qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), name = quote\_nullable(i\_name), address\_sno = quote\_nullable(i\_address\_sno), area\_sno = quote\_nullable(i\_area\_sno), qbox\_entity\_status\_cd = quote\_nullable(i\_qbox\_entity\_status\_cd), created\_on = quote\_nullable(i\_created\_on), entity\_code = quote\_nullable(i\_entity\_code), active\_flag = quote\_nullable(i\_active\_flag), state\_sno = quote\_nullable(i\_state\_sno), city\_sno = quote\_nullable(i\_city\_sno) WHERE <custom\_conditions>; |
| delete\_qbox\_entity\_by\_id | DELETE FROM masters.qbox\_entity WHERE qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno); |
| custom\_delete\_qbox\_entity\_by\_id | DELETE FROM masters.qbox\_entity WHERE <custom\_conditions>; |
| update\_qbox\_entity | UPDATE masters.qbox\_entity SET qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), name = quote\_nullable(i\_name), address\_sno = quote\_nullable(i\_address\_sno), area\_sno = quote\_nullable(i\_area\_sno), qbox\_entity\_status\_cd = quote\_nullable(i\_qbox\_entity\_status\_cd), created\_on = quote\_nullable(i\_created\_on), entity\_code = quote\_nullable(i\_entity\_code), active\_flag = quote\_nullable(i\_active\_flag), state\_sno = quote\_nullable(i\_state\_sno), city\_sno = quote\_nullable(i\_city\_sno) WHERE qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno); |
| get\_qbox\_entity\_by\_id | SELECT t.qbox\_entity\_sno, t.name, t.address\_sno, t.area\_sno, t.qbox\_entity\_status\_cd, t.created\_on, t.entity\_code, t.active\_flag, t.state\_sno, t.city\_sno, f\_area.name FROM masters.qbox\_entity t LEFT JOIN masters.area f\_area ON t.area\_sno = f\_area.area\_sno WHERE t.qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Sku Inventory**

### **Database Table Name: sku\_inventory**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| sku\_inventory\_sno | bigint - (Numeric Precision: 64) | sku\_inventory\_pkey (PRIMARY KEY), |
| purchase\_order\_dtl\_sno | bigint - (Numeric Precision: 64) | sku\_inventory\_purchase\_order\_dtl\_sno\_fkey (FOREIGN KEY), |
| unique\_code | text | null |
| sales\_order\_dtl\_sno | bigint - (Numeric Precision: 64) | sku\_inventory\_sales\_order\_dtl\_sno\_fkey (FOREIGN KEY), |
| wf\_stage\_cd | smallint - (Numeric Precision: 16) | null |
| description | text | null |
| active\_flag | boolean | null |
| transaction\_date | date | null |
| restaurant\_food\_sku\_sno | bigint - (Numeric Precision: 64) | fk\_restaurant\_food\_sku (FOREIGN KEY), |
| qbox\_entity\_sno | integer | fk\_qbox\_entity (FOREIGN KEY), |
| delivery\_partner\_sno | smallint - (Numeric Precision: 16) | fk\_delivery\_partner\_sno (FOREIGN KEY), |
| partner\_food\_sku\_sno | smallint - (Numeric Precision: 16) | fk\_partner\_food\_sku\_sno (FOREIGN KEY), |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Sku Inventory Sno | hidden | number | true | Manual Entry |
| Purchase Order Dtl Sno | listbox | number | false | API feed list box selection |
| Unique Code | textbox | string | true | Manual Entry |
| Sales Order Dtl Sno | listbox | number | false | API feed list box selection |
| Wf Stage Cd | textbox\_number | number | false | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |
| Transaction Date | timestamp | string | false | Manual Entry |
| Restaurant Food Sku Sno | listbox | number | false | API feed list box selection |
| Qbox Entity Sno | listbox | number | false | API feed list box selection |
| Delivery Partner Sno | listbox | number | false | API feed list box selection |
| Partner Food Sku Sno | listbox | number | false | API feed list box selection |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_sku\_inventorys | get | /api/v1/get\_sku\_inventorys |
| create\_sku\_inventory | post | /api/v1/create\_sku\_inventory |
| update\_sku\_inventory | put | /api/v1/update\_sku\_inventory |
| delete\_sku\_inventory | delete | /api/v1/delete\_sku\_inventory |
| get\_delivery\_partners | get | /api/v1/get\_delivery\_partners |
| get\_partner\_food\_skus | get | /api/v1/get\_partner\_food\_skus |
| get\_qbox\_entitys | get | /api/v1/get\_qbox\_entitys |
| get\_qbox\_entitys | get | /api/v1/get\_qbox\_entitys |
| get\_qbox\_entitys | get | /api/v1/get\_qbox\_entitys |
| get\_qbox\_entitys | get | /api/v1/get\_qbox\_entitys |
| get\_restaurant\_food\_skus | get | /api/v1/get\_restaurant\_food\_skus |
| get\_purchase\_order\_dtls | get | /api/v1/get\_purchase\_order\_dtls |
| get\_sales\_order\_dtls | get | /api/v1/get\_sales\_order\_dtls |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_sku\_inventorys | {} | {"data":{"wf\_stage\_cd":"smallint","transaction\_date":"date","delivery\_partner\_sno":"smallint","sales\_order\_dtl\_sno":"bigint","restaurant\_food\_sku\_sno":"bigint","description":"text","qbox\_entity\_sno":"integer","active\_flag":"boolean","partner\_food\_sku\_sno":"smallint","sku\_inventory\_sno":"bigint","purchase\_order\_dtl\_sno":"bigint","unique\_code":"text"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_sku\_inventory | {"data":{"wf\_stage\_cd":"smallint","transaction\_date":"date","delivery\_partner\_sno":"smallint","sales\_order\_dtl\_sno":"bigint","restaurant\_food\_sku\_sno":"bigint","description":"text","qbox\_entity\_sno":"integer","active\_flag":"boolean","partner\_food\_sku\_sno":"smallint","sku\_inventory\_sno":"bigint","purchase\_order\_dtl\_sno":"bigint","unique\_code":"text"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_sku\_inventory | {"data":{"wf\_stage\_cd":"smallint","transaction\_date":"date","delivery\_partner\_sno":"smallint","sales\_order\_dtl\_sno":"bigint","restaurant\_food\_sku\_sno":"bigint","description":"text","qbox\_entity\_sno":"integer","active\_flag":"boolean","partner\_food\_sku\_sno":"smallint","sku\_inventory\_sno":"bigint","purchase\_order\_dtl\_sno":"bigint","unique\_code":"text"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_sku\_inventory | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_delivery\_partners | {"id":"integer"} | {"data":"array of delivery\_partner"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_partner\_food\_skus | {"id":"integer"} | {"data":"array of partner\_food\_sku"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_qbox\_entitys | {"id":"integer"} | {"data":"array of qbox\_entity"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_qbox\_entitys | {"id":"integer"} | {"data":"array of qbox\_entity"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_qbox\_entitys | {"id":"integer"} | {"data":"array of qbox\_entity"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_qbox\_entitys | {"id":"integer"} | {"data":"array of qbox\_entity"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_restaurant\_food\_skus | {"id":"integer"} | {"data":"array of restaurant\_food\_sku"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_purchase\_order\_dtls | {"id":"integer"} | {"data":"array of purchase\_order\_dtl"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_sales\_order\_dtls | {"id":"integer"} | {"data":"array of sales\_order\_dtl"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| custom\_delete\_sku\_inventory\_by\_id | DELETE FROM masters.sku\_inventory WHERE <custom\_conditions>; |
| update\_sku\_inventory | UPDATE masters.sku\_inventory SET sku\_inventory\_sno = quote\_nullable(i\_sku\_inventory\_sno), purchase\_order\_dtl\_sno = quote\_nullable(i\_purchase\_order\_dtl\_sno), unique\_code = quote\_nullable(i\_unique\_code), sales\_order\_dtl\_sno = quote\_nullable(i\_sales\_order\_dtl\_sno), wf\_stage\_cd = quote\_nullable(i\_wf\_stage\_cd), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), transaction\_date = quote\_nullable(i\_transaction\_date), restaurant\_food\_sku\_sno = quote\_nullable(i\_restaurant\_food\_sku\_sno), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno), partner\_food\_sku\_sno = quote\_nullable(i\_partner\_food\_sku\_sno) WHERE sku\_inventory\_sno = quote\_nullable(i\_sku\_inventory\_sno); |
| get\_all\_sku\_inventorys | SELECT t.sku\_inventory\_sno, t.purchase\_order\_dtl\_sno, t.unique\_code, t.sales\_order\_dtl\_sno, t.wf\_stage\_cd, t.description, t.active\_flag, t.transaction\_date, t.restaurant\_food\_sku\_sno, t.qbox\_entity\_sno, t.delivery\_partner\_sno, t.partner\_food\_sku\_sno, f\_delivery\_partner.name, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_restaurant\_food\_sku.restaurant\_sku\_name FROM masters.sku\_inventory t LEFT JOIN masters.delivery\_partner f\_delivery\_partner ON t.delivery\_partner\_sno = f\_delivery\_partner.delivery\_partner\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.restaurant\_food\_sku f\_restaurant\_food\_sku ON t.restaurant\_food\_sku\_sno = f\_restaurant\_food\_sku.restaurant\_food\_sku\_sno ; |
| custom\_search\_sku\_inventory\_by\_id | SELECT t.sku\_inventory\_sno, t.purchase\_order\_dtl\_sno, t.unique\_code, t.sales\_order\_dtl\_sno, t.wf\_stage\_cd, t.description, t.active\_flag, t.transaction\_date, t.restaurant\_food\_sku\_sno, t.qbox\_entity\_sno, t.delivery\_partner\_sno, t.partner\_food\_sku\_sno, f\_delivery\_partner.name, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_restaurant\_food\_sku.restaurant\_sku\_name FROM masters.sku\_inventory t LEFT JOIN masters.delivery\_partner f\_delivery\_partner ON t.delivery\_partner\_sno = f\_delivery\_partner.delivery\_partner\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.restaurant\_food\_sku f\_restaurant\_food\_sku ON t.restaurant\_food\_sku\_sno = f\_restaurant\_food\_sku.restaurant\_food\_sku\_sno WHERE <custom\_search\_conditions>; |
| get\_sku\_inventory\_by\_id | SELECT t.sku\_inventory\_sno, t.purchase\_order\_dtl\_sno, t.unique\_code, t.sales\_order\_dtl\_sno, t.wf\_stage\_cd, t.description, t.active\_flag, t.transaction\_date, t.restaurant\_food\_sku\_sno, t.qbox\_entity\_sno, t.delivery\_partner\_sno, t.partner\_food\_sku\_sno, f\_delivery\_partner.name, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_restaurant\_food\_sku.restaurant\_sku\_name FROM masters.sku\_inventory t LEFT JOIN masters.delivery\_partner f\_delivery\_partner ON t.delivery\_partner\_sno = f\_delivery\_partner.delivery\_partner\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.restaurant\_food\_sku f\_restaurant\_food\_sku ON t.restaurant\_food\_sku\_sno = f\_restaurant\_food\_sku.restaurant\_food\_sku\_sno WHERE t.sku\_inventory\_sno = quote\_nullable(i\_sku\_inventory\_sno); |
| custom\_update\_sku\_inventory | UPDATE masters.sku\_inventory SET sku\_inventory\_sno = quote\_nullable(i\_sku\_inventory\_sno), purchase\_order\_dtl\_sno = quote\_nullable(i\_purchase\_order\_dtl\_sno), unique\_code = quote\_nullable(i\_unique\_code), sales\_order\_dtl\_sno = quote\_nullable(i\_sales\_order\_dtl\_sno), wf\_stage\_cd = quote\_nullable(i\_wf\_stage\_cd), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), transaction\_date = quote\_nullable(i\_transaction\_date), restaurant\_food\_sku\_sno = quote\_nullable(i\_restaurant\_food\_sku\_sno), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno), partner\_food\_sku\_sno = quote\_nullable(i\_partner\_food\_sku\_sno) WHERE <custom\_conditions>; |
| create\_sku\_inventory | INSERT INTO masters.sku\_inventory (t.sku\_inventory\_sno, t.purchase\_order\_dtl\_sno, t.unique\_code, t.sales\_order\_dtl\_sno, t.wf\_stage\_cd, t.description, t.active\_flag, t.transaction\_date, t.restaurant\_food\_sku\_sno, t.qbox\_entity\_sno, t.delivery\_partner\_sno, t.partner\_food\_sku\_sno) VALUES (i\_sku\_inventory\_sno, i\_purchase\_order\_dtl\_sno, i\_unique\_code, i\_sales\_order\_dtl\_sno, i\_wf\_stage\_cd, i\_description, i\_active\_flag, i\_transaction\_date, i\_restaurant\_food\_sku\_sno, i\_qbox\_entity\_sno, i\_delivery\_partner\_sno, i\_partner\_food\_sku\_sno); |
| unique\_sku\_inventory\_exists | SELECT 1 FROM masters.sku\_inventory WHERE <unique\_conditions> LIMIT 1; |
| delete\_sku\_inventory\_by\_id | DELETE FROM masters.sku\_inventory WHERE sku\_inventory\_sno = quote\_nullable(i\_sku\_inventory\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Codes Dtl**

### **Database Table Name: codes\_dtl**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| codes\_dtl\_sno | smallint - (Numeric Precision: 16) | codes\_dtl\_pkey (PRIMARY KEY), |
| codes\_hdr\_sno | smallint - (Numeric Precision: 16) | null |
| description | text | null |
| seqno | integer | null |
| filter\_1 | text | null |
| filter\_2 | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Codes Dtl Sno | hidden | number | true | Manual Entry |
| Codes Hdr Sno | textbox\_number | number | true | Manual Entry |
| Description | textbox | string | true | Manual Entry |
| Seqno | textbox | number | false | Manual Entry |
| Filter 1 | textbox | string | false | Manual Entry |
| Filter 2 | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | true | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_codes\_dtls | get | /api/v1/get\_codes\_dtls |
| create\_codes\_dtl | post | /api/v1/create\_codes\_dtl |
| update\_codes\_dtl | put | /api/v1/update\_codes\_dtl |
| delete\_codes\_dtl | delete | /api/v1/delete\_codes\_dtl |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_codes\_dtls | {} | {"data":{"seqno":"integer","codes\_dtl\_sno":"smallint","filter\_2":"text","filter\_1":"text","description":"text","active\_flag":"boolean","codes\_hdr\_sno":"smallint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_codes\_dtl | {"data":{"seqno":"integer","codes\_dtl\_sno":"smallint","filter\_2":"text","filter\_1":"text","description":"text","active\_flag":"boolean","codes\_hdr\_sno":"smallint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_codes\_dtl | {"data":{"seqno":"integer","codes\_dtl\_sno":"smallint","filter\_2":"text","filter\_1":"text","description":"text","active\_flag":"boolean","codes\_hdr\_sno":"smallint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_codes\_dtl | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| create\_codes\_dtl | INSERT INTO masters.codes\_dtl (t.codes\_dtl\_sno, t.codes\_hdr\_sno, t.description, t.seqno, t.filter\_1, t.filter\_2, t.active\_flag) VALUES (i\_codes\_dtl\_sno, i\_codes\_hdr\_sno, i\_description, i\_seqno, i\_filter\_1, i\_filter\_2, i\_active\_flag); |
| unique\_codes\_dtl\_exists | SELECT 1 FROM masters.codes\_dtl WHERE <unique\_conditions> LIMIT 1; |
| update\_codes\_dtl | UPDATE masters.codes\_dtl SET codes\_dtl\_sno = quote\_nullable(i\_codes\_dtl\_sno), codes\_hdr\_sno = quote\_nullable(i\_codes\_hdr\_sno), description = quote\_nullable(i\_description), seqno = quote\_nullable(i\_seqno), filter\_1 = quote\_nullable(i\_filter\_1), filter\_2 = quote\_nullable(i\_filter\_2), active\_flag = quote\_nullable(i\_active\_flag) WHERE codes\_dtl\_sno = quote\_nullable(i\_codes\_dtl\_sno); |
| get\_codes\_dtl\_by\_id | SELECT t.codes\_dtl\_sno, t.codes\_hdr\_sno, t.description, t.seqno, t.filter\_1, t.filter\_2, t.active\_flag FROM masters.codes\_dtl t WHERE t.codes\_dtl\_sno = quote\_nullable(i\_codes\_dtl\_sno); |
| custom\_update\_codes\_dtl | UPDATE masters.codes\_dtl SET codes\_dtl\_sno = quote\_nullable(i\_codes\_dtl\_sno), codes\_hdr\_sno = quote\_nullable(i\_codes\_hdr\_sno), description = quote\_nullable(i\_description), seqno = quote\_nullable(i\_seqno), filter\_1 = quote\_nullable(i\_filter\_1), filter\_2 = quote\_nullable(i\_filter\_2), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| get\_all\_codes\_dtls | SELECT t.codes\_dtl\_sno, t.codes\_hdr\_sno, t.description, t.seqno, t.filter\_1, t.filter\_2, t.active\_flag FROM masters.codes\_dtl t ; |
| delete\_codes\_dtl\_by\_id | DELETE FROM masters.codes\_dtl WHERE codes\_dtl\_sno = quote\_nullable(i\_codes\_dtl\_sno); |
| custom\_search\_codes\_dtl\_by\_id | SELECT t.codes\_dtl\_sno, t.codes\_hdr\_sno, t.description, t.seqno, t.filter\_1, t.filter\_2, t.active\_flag FROM masters.codes\_dtl t WHERE <custom\_search\_conditions>; |
| custom\_delete\_codes\_dtl\_by\_id | DELETE FROM masters.codes\_dtl WHERE <custom\_conditions>; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Delivery Partner**

### **Database Table Name: delivery\_partner**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| delivery\_partner\_sno | smallint - (Numeric Precision: 16) | delivery\_partner\_pkey (PRIMARY KEY), |
| partner\_code | character varying | null |
| name | text | null |
| partner\_status\_cd | smallint - (Numeric Precision: 16) | delivery\_partner\_partner\_status\_cd\_fkey (FOREIGN KEY), |
| active\_flag | boolean | null |
| logo\_url | text | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Delivery Partner Sno | hidden | number | true | Manual Entry |
| Partner Code | textbox | string | true | Manual Entry |
| Name | textbox | string | true | Manual Entry |
| Partner Status Cd | listbox | number | true | API feed list box selection |
| Active Flag | checkbox | boolean | false | Manual Entry |
| Logo Url | textbox | string | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_delivery\_partners | get | /api/v1/get\_delivery\_partners |
| create\_delivery\_partner | post | /api/v1/create\_delivery\_partner |
| update\_delivery\_partner | put | /api/v1/update\_delivery\_partner |
| delete\_delivery\_partner | delete | /api/v1/delete\_delivery\_partner |
| get\_codes\_dtls | get | /api/v1/get\_codes\_dtls |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_delivery\_partners | {} | {"data":{"delivery\_partner\_sno":"smallint","logo\_url":"text","partner\_status\_cd":"smallint","name":"text","active\_flag":"boolean","partner\_code":"character varying"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_delivery\_partner | {"data":{"delivery\_partner\_sno":"smallint","logo\_url":"text","partner\_status\_cd":"smallint","name":"text","active\_flag":"boolean","partner\_code":"character varying"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_delivery\_partner | {"data":{"delivery\_partner\_sno":"smallint","logo\_url":"text","partner\_status\_cd":"smallint","name":"text","active\_flag":"boolean","partner\_code":"character varying"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_delivery\_partner | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_codes\_dtls | {"id":"integer"} | {"data":"array of codes\_dtl"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| create\_delivery\_partner | INSERT INTO masters.delivery\_partner (t.delivery\_partner\_sno, t.partner\_code, t.name, t.partner\_status\_cd, t.active\_flag, t.logo\_url) VALUES (i\_delivery\_partner\_sno, i\_partner\_code, i\_name, i\_partner\_status\_cd, i\_active\_flag, i\_logo\_url); |
| unique\_delivery\_partner\_exists | SELECT 1 FROM masters.delivery\_partner WHERE <unique\_conditions> LIMIT 1; |
| custom\_update\_delivery\_partner | UPDATE masters.delivery\_partner SET delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno), partner\_code = quote\_nullable(i\_partner\_code), name = quote\_nullable(i\_name), partner\_status\_cd = quote\_nullable(i\_partner\_status\_cd), active\_flag = quote\_nullable(i\_active\_flag), logo\_url = quote\_nullable(i\_logo\_url) WHERE <custom\_conditions>; |
| custom\_delete\_delivery\_partner\_by\_id | DELETE FROM masters.delivery\_partner WHERE <custom\_conditions>; |
| custom\_search\_delivery\_partner\_by\_id | SELECT t.delivery\_partner\_sno, t.partner\_code, t.name, t.partner\_status\_cd, t.active\_flag, t.logo\_url FROM masters.delivery\_partner t WHERE <custom\_search\_conditions>; |
| update\_delivery\_partner | UPDATE masters.delivery\_partner SET delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno), partner\_code = quote\_nullable(i\_partner\_code), name = quote\_nullable(i\_name), partner\_status\_cd = quote\_nullable(i\_partner\_status\_cd), active\_flag = quote\_nullable(i\_active\_flag), logo\_url = quote\_nullable(i\_logo\_url) WHERE delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno); |
| delete\_delivery\_partner\_by\_id | DELETE FROM masters.delivery\_partner WHERE delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno); |
| get\_all\_delivery\_partners | SELECT t.delivery\_partner\_sno, t.partner\_code, t.name, t.partner\_status\_cd, t.active\_flag, t.logo\_url FROM masters.delivery\_partner t ; |
| get\_delivery\_partner\_by\_id | SELECT t.delivery\_partner\_sno, t.partner\_code, t.name, t.partner\_status\_cd, t.active\_flag, t.logo\_url FROM masters.delivery\_partner t WHERE t.delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Infra**

### **Database Table Name: infra**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| infra\_sno | smallint - (Numeric Precision: 16) | infra\_pkey (PRIMARY KEY), |
| name | text | null |
| active\_flag | boolean | null |
| infra\_icon | character varying | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Infra Sno | hidden | number | true | Manual Entry |
| Name | textbox | string | true | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |
| Infra Icon | textbox | string | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_infras | get | /api/v1/get\_infras |
| create\_infra | post | /api/v1/create\_infra |
| update\_infra | put | /api/v1/update\_infra |
| delete\_infra | delete | /api/v1/delete\_infra |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_infras | {} | {"data":{"name":"text","active\_flag":"boolean","infra\_sno":"smallint","infra\_icon":"character varying"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_infra | {"data":{"name":"text","active\_flag":"boolean","infra\_sno":"smallint","infra\_icon":"character varying"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_infra | {"data":{"name":"text","active\_flag":"boolean","infra\_sno":"smallint","infra\_icon":"character varying"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_infra | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| custom\_delete\_infra\_by\_id | DELETE FROM masters.infra WHERE <custom\_conditions>; |
| custom\_search\_infra\_by\_id | SELECT t.infra\_sno, t.name, t.active\_flag, t.infra\_icon FROM masters.infra t WHERE <custom\_search\_conditions>; |
| custom\_update\_infra | UPDATE masters.infra SET infra\_sno = quote\_nullable(i\_infra\_sno), name = quote\_nullable(i\_name), active\_flag = quote\_nullable(i\_active\_flag), infra\_icon = quote\_nullable(i\_infra\_icon) WHERE <custom\_conditions>; |
| unique\_infra\_exists | SELECT 1 FROM masters.infra WHERE <unique\_conditions> LIMIT 1; |
| delete\_infra\_by\_id | DELETE FROM masters.infra WHERE infra\_sno = quote\_nullable(i\_infra\_sno); |
| update\_infra | UPDATE masters.infra SET infra\_sno = quote\_nullable(i\_infra\_sno), name = quote\_nullable(i\_name), active\_flag = quote\_nullable(i\_active\_flag), infra\_icon = quote\_nullable(i\_infra\_icon) WHERE infra\_sno = quote\_nullable(i\_infra\_sno); |
| create\_infra | INSERT INTO masters.infra (t.infra\_sno, t.name, t.active\_flag, t.infra\_icon) VALUES (i\_infra\_sno, i\_name, i\_active\_flag, i\_infra\_icon); |
| get\_infra\_by\_id | SELECT t.infra\_sno, t.name, t.active\_flag, t.infra\_icon FROM masters.infra t WHERE t.infra\_sno = quote\_nullable(i\_infra\_sno); |
| get\_all\_infras | SELECT t.infra\_sno, t.name, t.active\_flag, t.infra\_icon FROM masters.infra t ; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Entity Infra Property**

### **Database Table Name: entity\_infra\_property**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| entity\_infra\_property\_sno | bigint - (Numeric Precision: 64) | entity\_infra\_property\_pkey (PRIMARY KEY), |
| entity\_infra\_sno | bigint - (Numeric Precision: 64) | entity\_infra\_property\_entity\_infra\_sno\_fkey (FOREIGN KEY), |
| infra\_property\_sno | smallint - (Numeric Precision: 16) | entity\_infra\_property\_infra\_property\_sno\_fkey (FOREIGN KEY), |
| value | text | null |
| description | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Entity Infra Property Sno | hidden | number | true | Manual Entry |
| Entity Infra Sno | listbox | number | true | API feed list box selection |
| Infra Property Sno | listbox | number | true | API feed list box selection |
| Value | textbox | string | true | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_entity\_infra\_propertys | get | /api/v1/get\_entity\_infra\_propertys |
| create\_entity\_infra\_property | post | /api/v1/create\_entity\_infra\_property |
| update\_entity\_infra\_property | put | /api/v1/update\_entity\_infra\_property |
| delete\_entity\_infra\_property | delete | /api/v1/delete\_entity\_infra\_property |
| get\_entity\_infras | get | /api/v1/get\_entity\_infras |
| get\_infra\_propertys | get | /api/v1/get\_infra\_propertys |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_entity\_infra\_propertys | {} | {"data":{"infra\_property\_sno":"smallint","entity\_infra\_property\_sno":"bigint","description":"text","active\_flag":"boolean","value":"text","entity\_infra\_sno":"bigint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_entity\_infra\_property | {"data":{"infra\_property\_sno":"smallint","entity\_infra\_property\_sno":"bigint","description":"text","active\_flag":"boolean","value":"text","entity\_infra\_sno":"bigint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_entity\_infra\_property | {"data":{"infra\_property\_sno":"smallint","entity\_infra\_property\_sno":"bigint","description":"text","active\_flag":"boolean","value":"text","entity\_infra\_sno":"bigint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_entity\_infra\_property | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_entity\_infras | {"id":"integer"} | {"data":"array of entity\_infra"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_infra\_propertys | {"id":"integer"} | {"data":"array of infra\_property"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| create\_entity\_infra\_property | INSERT INTO masters.entity\_infra\_property (t.entity\_infra\_property\_sno, t.entity\_infra\_sno, t.infra\_property\_sno, t.value, t.description, t.active\_flag) VALUES (i\_entity\_infra\_property\_sno, i\_entity\_infra\_sno, i\_infra\_property\_sno, i\_value, i\_description, i\_active\_flag); |
| get\_all\_entity\_infra\_propertys | SELECT t.entity\_infra\_property\_sno, t.entity\_infra\_sno, t.infra\_property\_sno, t.value, t.description, t.active\_flag, f\_infra\_property.name FROM masters.entity\_infra\_property t LEFT JOIN masters.infra\_property f\_infra\_property ON t.infra\_property\_sno = f\_infra\_property.infra\_property\_sno ; |
| custom\_update\_entity\_infra\_property | UPDATE masters.entity\_infra\_property SET entity\_infra\_property\_sno = quote\_nullable(i\_entity\_infra\_property\_sno), entity\_infra\_sno = quote\_nullable(i\_entity\_infra\_sno), infra\_property\_sno = quote\_nullable(i\_infra\_property\_sno), value = quote\_nullable(i\_value), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| get\_entity\_infra\_property\_by\_id | SELECT t.entity\_infra\_property\_sno, t.entity\_infra\_sno, t.infra\_property\_sno, t.value, t.description, t.active\_flag, f\_infra\_property.name FROM masters.entity\_infra\_property t LEFT JOIN masters.infra\_property f\_infra\_property ON t.infra\_property\_sno = f\_infra\_property.infra\_property\_sno WHERE t.entity\_infra\_property\_sno = quote\_nullable(i\_entity\_infra\_property\_sno); |
| custom\_search\_entity\_infra\_property\_by\_id | SELECT t.entity\_infra\_property\_sno, t.entity\_infra\_sno, t.infra\_property\_sno, t.value, t.description, t.active\_flag, f\_infra\_property.name FROM masters.entity\_infra\_property t LEFT JOIN masters.infra\_property f\_infra\_property ON t.infra\_property\_sno = f\_infra\_property.infra\_property\_sno WHERE <custom\_search\_conditions>; |
| custom\_delete\_entity\_infra\_property\_by\_id | DELETE FROM masters.entity\_infra\_property WHERE <custom\_conditions>; |
| unique\_entity\_infra\_property\_exists | SELECT 1 FROM masters.entity\_infra\_property WHERE <unique\_conditions> LIMIT 1; |
| update\_entity\_infra\_property | UPDATE masters.entity\_infra\_property SET entity\_infra\_property\_sno = quote\_nullable(i\_entity\_infra\_property\_sno), entity\_infra\_sno = quote\_nullable(i\_entity\_infra\_sno), infra\_property\_sno = quote\_nullable(i\_infra\_property\_sno), value = quote\_nullable(i\_value), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE entity\_infra\_property\_sno = quote\_nullable(i\_entity\_infra\_property\_sno); |
| delete\_entity\_infra\_property\_by\_id | DELETE FROM masters.entity\_infra\_property WHERE entity\_infra\_property\_sno = quote\_nullable(i\_entity\_infra\_property\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Infra Property**

### **Database Table Name: infra\_property**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| infra\_property\_sno | smallint - (Numeric Precision: 16) | infra\_property\_pkey (PRIMARY KEY), |
| infra\_sno | smallint - (Numeric Precision: 16) | infra\_property\_infra\_sno\_fkey (FOREIGN KEY), |
| name | text | null |
| data\_type\_cd | smallint - (Numeric Precision: 16) | infra\_property\_data\_type\_cd\_fkey (FOREIGN KEY), |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Infra Property Sno | hidden | number | true | Manual Entry |
| Infra Sno | listbox | number | true | API feed list box selection |
| Name | textbox | string | true | Manual Entry |
| Data Type Cd | listbox | number | true | API feed list box selection |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_infra\_propertys | get | /api/v1/get\_infra\_propertys |
| create\_infra\_property | post | /api/v1/create\_infra\_property |
| update\_infra\_property | put | /api/v1/update\_infra\_property |
| delete\_infra\_property | delete | /api/v1/delete\_infra\_property |
| get\_codes\_dtls | get | /api/v1/get\_codes\_dtls |
| get\_infras | get | /api/v1/get\_infras |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_infra\_propertys | {} | {"data":{"infra\_property\_sno":"smallint","name":"text","data\_type\_cd":"smallint","active\_flag":"boolean","infra\_sno":"smallint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_infra\_property | {"data":{"infra\_property\_sno":"smallint","name":"text","data\_type\_cd":"smallint","active\_flag":"boolean","infra\_sno":"smallint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_infra\_property | {"data":{"infra\_property\_sno":"smallint","name":"text","data\_type\_cd":"smallint","active\_flag":"boolean","infra\_sno":"smallint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_infra\_property | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_codes\_dtls | {"id":"integer"} | {"data":"array of codes\_dtl"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_infras | {"id":"integer"} | {"data":"array of infra"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| unique\_infra\_property\_exists | SELECT 1 FROM masters.infra\_property WHERE <unique\_conditions> LIMIT 1; |
| create\_infra\_property | INSERT INTO masters.infra\_property (t.infra\_property\_sno, t.infra\_sno, t.name, t.data\_type\_cd, t.active\_flag) VALUES (i\_infra\_property\_sno, i\_infra\_sno, i\_name, i\_data\_type\_cd, i\_active\_flag); |
| delete\_infra\_property\_by\_id | DELETE FROM masters.infra\_property WHERE infra\_property\_sno = quote\_nullable(i\_infra\_property\_sno); |
| custom\_delete\_infra\_property\_by\_id | DELETE FROM masters.infra\_property WHERE <custom\_conditions>; |
| get\_all\_infra\_propertys | SELECT t.infra\_property\_sno, t.infra\_sno, t.name, t.data\_type\_cd, t.active\_flag, f\_infra.name FROM masters.infra\_property t LEFT JOIN masters.infra f\_infra ON t.infra\_sno = f\_infra.infra\_sno ; |
| custom\_update\_infra\_property | UPDATE masters.infra\_property SET infra\_property\_sno = quote\_nullable(i\_infra\_property\_sno), infra\_sno = quote\_nullable(i\_infra\_sno), name = quote\_nullable(i\_name), data\_type\_cd = quote\_nullable(i\_data\_type\_cd), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| custom\_search\_infra\_property\_by\_id | SELECT t.infra\_property\_sno, t.infra\_sno, t.name, t.data\_type\_cd, t.active\_flag, f\_infra.name FROM masters.infra\_property t LEFT JOIN masters.infra f\_infra ON t.infra\_sno = f\_infra.infra\_sno WHERE <custom\_search\_conditions>; |
| get\_infra\_property\_by\_id | SELECT t.infra\_property\_sno, t.infra\_sno, t.name, t.data\_type\_cd, t.active\_flag, f\_infra.name FROM masters.infra\_property t LEFT JOIN masters.infra f\_infra ON t.infra\_sno = f\_infra.infra\_sno WHERE t.infra\_property\_sno = quote\_nullable(i\_infra\_property\_sno); |
| update\_infra\_property | UPDATE masters.infra\_property SET infra\_property\_sno = quote\_nullable(i\_infra\_property\_sno), infra\_sno = quote\_nullable(i\_infra\_sno), name = quote\_nullable(i\_name), data\_type\_cd = quote\_nullable(i\_data\_type\_cd), active\_flag = quote\_nullable(i\_active\_flag) WHERE infra\_property\_sno = quote\_nullable(i\_infra\_property\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Partner Food Sku**

### **Database Table Name: partner\_food\_sku**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| partner\_food\_sku\_sno | smallint - (Numeric Precision: 16) | partner\_food\_sku\_pkey (PRIMARY KEY), |
| delivery\_partner\_sno | smallint - (Numeric Precision: 16) | partner\_food\_sku\_delivery\_partner\_sno\_fkey (FOREIGN KEY), |
| restaurant\_food\_sku\_sno | smallint - (Numeric Precision: 16) | partner\_food\_sku\_restaurant\_food\_sku\_sno\_fkey (FOREIGN KEY), |
| partner\_food\_code | text | null |
| description | text | null |
| active\_flag | boolean | null |
| purchase\_sku\_price | double precision | null |
| sales\_sku\_price | double precision | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Partner Food Sku Sno | hidden | number | true | Manual Entry |
| Delivery Partner Sno | listbox | number | true | API feed list box selection |
| Restaurant Food Sku Sno | listbox | number | true | API feed list box selection |
| Partner Food Code | textbox | string | true | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |
| Purchase Sku Price | textbox | number | false | Manual Entry |
| Sales Sku Price | textbox | number | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_partner\_food\_skus | get | /api/v1/get\_partner\_food\_skus |
| create\_partner\_food\_sku | post | /api/v1/create\_partner\_food\_sku |
| update\_partner\_food\_sku | put | /api/v1/update\_partner\_food\_sku |
| delete\_partner\_food\_sku | delete | /api/v1/delete\_partner\_food\_sku |
| get\_delivery\_partners | get | /api/v1/get\_delivery\_partners |
| get\_restaurant\_food\_skus | get | /api/v1/get\_restaurant\_food\_skus |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_partner\_food\_skus | {} | {"data":{"sales\_sku\_price":"double precision","delivery\_partner\_sno":"smallint","purchase\_sku\_price":"double precision","restaurant\_food\_sku\_sno":"smallint","description":"text","partner\_food\_code":"text","active\_flag":"boolean","partner\_food\_sku\_sno":"smallint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_partner\_food\_sku | {"data":{"sales\_sku\_price":"double precision","delivery\_partner\_sno":"smallint","purchase\_sku\_price":"double precision","restaurant\_food\_sku\_sno":"smallint","description":"text","partner\_food\_code":"text","active\_flag":"boolean","partner\_food\_sku\_sno":"smallint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_partner\_food\_sku | {"data":{"sales\_sku\_price":"double precision","delivery\_partner\_sno":"smallint","purchase\_sku\_price":"double precision","restaurant\_food\_sku\_sno":"smallint","description":"text","partner\_food\_code":"text","active\_flag":"boolean","partner\_food\_sku\_sno":"smallint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_partner\_food\_sku | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_delivery\_partners | {"id":"integer"} | {"data":"array of delivery\_partner"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_restaurant\_food\_skus | {"id":"integer"} | {"data":"array of restaurant\_food\_sku"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| create\_partner\_food\_sku | INSERT INTO masters.partner\_food\_sku (t.partner\_food\_sku\_sno, t.delivery\_partner\_sno, t.restaurant\_food\_sku\_sno, t.partner\_food\_code, t.description, t.active\_flag, t.purchase\_sku\_price, t.sales\_sku\_price) VALUES (i\_partner\_food\_sku\_sno, i\_delivery\_partner\_sno, i\_restaurant\_food\_sku\_sno, i\_partner\_food\_code, i\_description, i\_active\_flag, i\_purchase\_sku\_price, i\_sales\_sku\_price); |
| custom\_update\_partner\_food\_sku | UPDATE masters.partner\_food\_sku SET partner\_food\_sku\_sno = quote\_nullable(i\_partner\_food\_sku\_sno), delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno), restaurant\_food\_sku\_sno = quote\_nullable(i\_restaurant\_food\_sku\_sno), partner\_food\_code = quote\_nullable(i\_partner\_food\_code), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), purchase\_sku\_price = quote\_nullable(i\_purchase\_sku\_price), sales\_sku\_price = quote\_nullable(i\_sales\_sku\_price) WHERE <custom\_conditions>; |
| get\_all\_partner\_food\_skus | SELECT t.partner\_food\_sku\_sno, t.delivery\_partner\_sno, t.restaurant\_food\_sku\_sno, t.partner\_food\_code, t.description, t.active\_flag, t.purchase\_sku\_price, t.sales\_sku\_price, f\_delivery\_partner.name, f\_restaurant\_food\_sku.restaurant\_sku\_name FROM masters.partner\_food\_sku t LEFT JOIN masters.delivery\_partner f\_delivery\_partner ON t.delivery\_partner\_sno = f\_delivery\_partner.delivery\_partner\_sno LEFT JOIN masters.restaurant\_food\_sku f\_restaurant\_food\_sku ON t.restaurant\_food\_sku\_sno = f\_restaurant\_food\_sku.restaurant\_food\_sku\_sno ; |
| custom\_search\_partner\_food\_sku\_by\_id | SELECT t.partner\_food\_sku\_sno, t.delivery\_partner\_sno, t.restaurant\_food\_sku\_sno, t.partner\_food\_code, t.description, t.active\_flag, t.purchase\_sku\_price, t.sales\_sku\_price, f\_delivery\_partner.name, f\_restaurant\_food\_sku.restaurant\_sku\_name FROM masters.partner\_food\_sku t LEFT JOIN masters.delivery\_partner f\_delivery\_partner ON t.delivery\_partner\_sno = f\_delivery\_partner.delivery\_partner\_sno LEFT JOIN masters.restaurant\_food\_sku f\_restaurant\_food\_sku ON t.restaurant\_food\_sku\_sno = f\_restaurant\_food\_sku.restaurant\_food\_sku\_sno WHERE <custom\_search\_conditions>; |
| update\_partner\_food\_sku | UPDATE masters.partner\_food\_sku SET partner\_food\_sku\_sno = quote\_nullable(i\_partner\_food\_sku\_sno), delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno), restaurant\_food\_sku\_sno = quote\_nullable(i\_restaurant\_food\_sku\_sno), partner\_food\_code = quote\_nullable(i\_partner\_food\_code), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), purchase\_sku\_price = quote\_nullable(i\_purchase\_sku\_price), sales\_sku\_price = quote\_nullable(i\_sales\_sku\_price) WHERE partner\_food\_sku\_sno = quote\_nullable(i\_partner\_food\_sku\_sno); |
| get\_partner\_food\_sku\_by\_id | SELECT t.partner\_food\_sku\_sno, t.delivery\_partner\_sno, t.restaurant\_food\_sku\_sno, t.partner\_food\_code, t.description, t.active\_flag, t.purchase\_sku\_price, t.sales\_sku\_price, f\_delivery\_partner.name, f\_restaurant\_food\_sku.restaurant\_sku\_name FROM masters.partner\_food\_sku t LEFT JOIN masters.delivery\_partner f\_delivery\_partner ON t.delivery\_partner\_sno = f\_delivery\_partner.delivery\_partner\_sno LEFT JOIN masters.restaurant\_food\_sku f\_restaurant\_food\_sku ON t.restaurant\_food\_sku\_sno = f\_restaurant\_food\_sku.restaurant\_food\_sku\_sno WHERE t.partner\_food\_sku\_sno = quote\_nullable(i\_partner\_food\_sku\_sno); |
| delete\_partner\_food\_sku\_by\_id | DELETE FROM masters.partner\_food\_sku WHERE partner\_food\_sku\_sno = quote\_nullable(i\_partner\_food\_sku\_sno); |
| custom\_delete\_partner\_food\_sku\_by\_id | DELETE FROM masters.partner\_food\_sku WHERE <custom\_conditions>; |
| unique\_partner\_food\_sku\_exists | SELECT 1 FROM masters.partner\_food\_sku WHERE <unique\_conditions> LIMIT 1; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Purchase Order Dtl**

### **Database Table Name: purchase\_order\_dtl**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| purchase\_order\_dtl\_sno | bigint - (Numeric Precision: 64) | purchase\_order\_dtl\_pkey (PRIMARY KEY), |
| purchase\_order\_sno | bigint - (Numeric Precision: 64) | purchase\_order\_dtl\_purchase\_order\_sno\_fkey (FOREIGN KEY), |
| restaurant\_food\_sku\_sno | bigint - (Numeric Precision: 64) | purchase\_order\_dtl\_restaurant\_food\_sku\_sno\_fkey (FOREIGN KEY), |
| order\_quantity | smallint - (Numeric Precision: 16) | null |
| sku\_price | numeric | null |
| partner\_food\_code | text | null |
| accepted\_quantity | smallint - (Numeric Precision: 16) | null |
| description | text | null |
| active\_flag | boolean | null |
| qbox\_entity\_sno | integer | fk\_qbox\_entity (FOREIGN KEY), |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Purchase Order Dtl Sno | hidden | number | true | Manual Entry |
| Purchase Order Sno | listbox | number | true | API feed list box selection |
| Restaurant Food Sku Sno | listbox | number | false | API feed list box selection |
| Order Quantity | textbox\_number | number | true | Manual Entry |
| Sku Price | textbox | number | true | Manual Entry |
| Partner Food Code | textbox | string | false | Manual Entry |
| Accepted Quantity | textbox\_number | number | false | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |
| Qbox Entity Sno | listbox | number | false | API feed list box selection |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_purchase\_order\_dtls | get | /api/v1/get\_purchase\_order\_dtls |
| create\_purchase\_order\_dtl | post | /api/v1/create\_purchase\_order\_dtl |
| update\_purchase\_order\_dtl | put | /api/v1/update\_purchase\_order\_dtl |
| delete\_purchase\_order\_dtl | delete | /api/v1/delete\_purchase\_order\_dtl |
| get\_qbox\_entitys | get | /api/v1/get\_qbox\_entitys |
| get\_qbox\_entitys | get | /api/v1/get\_qbox\_entitys |
| get\_qbox\_entitys | get | /api/v1/get\_qbox\_entitys |
| get\_qbox\_entitys | get | /api/v1/get\_qbox\_entitys |
| get\_purchase\_orders | get | /api/v1/get\_purchase\_orders |
| get\_restaurant\_food\_skus | get | /api/v1/get\_restaurant\_food\_skus |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_purchase\_order\_dtls | {} | {"data":{"restaurant\_food\_sku\_sno":"bigint","sku\_price":"numeric","order\_quantity":"smallint","description":"text","qbox\_entity\_sno":"integer","accepted\_quantity":"smallint","partner\_food\_code":"text","active\_flag":"boolean","purchase\_order\_dtl\_sno":"bigint","purchase\_order\_sno":"bigint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_purchase\_order\_dtl | {"data":{"restaurant\_food\_sku\_sno":"bigint","sku\_price":"numeric","order\_quantity":"smallint","description":"text","qbox\_entity\_sno":"integer","accepted\_quantity":"smallint","partner\_food\_code":"text","active\_flag":"boolean","purchase\_order\_dtl\_sno":"bigint","purchase\_order\_sno":"bigint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_purchase\_order\_dtl | {"data":{"restaurant\_food\_sku\_sno":"bigint","sku\_price":"numeric","order\_quantity":"smallint","description":"text","qbox\_entity\_sno":"integer","accepted\_quantity":"smallint","partner\_food\_code":"text","active\_flag":"boolean","purchase\_order\_dtl\_sno":"bigint","purchase\_order\_sno":"bigint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_purchase\_order\_dtl | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_qbox\_entitys | {"id":"integer"} | {"data":"array of qbox\_entity"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_qbox\_entitys | {"id":"integer"} | {"data":"array of qbox\_entity"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_qbox\_entitys | {"id":"integer"} | {"data":"array of qbox\_entity"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_qbox\_entitys | {"id":"integer"} | {"data":"array of qbox\_entity"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_purchase\_orders | {"id":"integer"} | {"data":"array of purchase\_order"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_restaurant\_food\_skus | {"id":"integer"} | {"data":"array of restaurant\_food\_sku"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| get\_all\_purchase\_order\_dtls | SELECT t.purchase\_order\_dtl\_sno, t.purchase\_order\_sno, t.restaurant\_food\_sku\_sno, t.order\_quantity, t.sku\_price, t.partner\_food\_code, t.accepted\_quantity, t.description, t.active\_flag, t.qbox\_entity\_sno, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_purchase\_order.order\_status\_desc, f\_restaurant\_food\_sku.restaurant\_sku\_name FROM masters.purchase\_order\_dtl t LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.purchase\_order f\_purchase\_order ON t.purchase\_order\_sno = f\_purchase\_order.purchase\_order\_sno LEFT JOIN masters.restaurant\_food\_sku f\_restaurant\_food\_sku ON t.restaurant\_food\_sku\_sno = f\_restaurant\_food\_sku.restaurant\_food\_sku\_sno ; |
| get\_purchase\_order\_dtl\_by\_id | SELECT t.purchase\_order\_dtl\_sno, t.purchase\_order\_sno, t.restaurant\_food\_sku\_sno, t.order\_quantity, t.sku\_price, t.partner\_food\_code, t.accepted\_quantity, t.description, t.active\_flag, t.qbox\_entity\_sno, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_purchase\_order.order\_status\_desc, f\_restaurant\_food\_sku.restaurant\_sku\_name FROM masters.purchase\_order\_dtl t LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.purchase\_order f\_purchase\_order ON t.purchase\_order\_sno = f\_purchase\_order.purchase\_order\_sno LEFT JOIN masters.restaurant\_food\_sku f\_restaurant\_food\_sku ON t.restaurant\_food\_sku\_sno = f\_restaurant\_food\_sku.restaurant\_food\_sku\_sno WHERE t.purchase\_order\_dtl\_sno = quote\_nullable(i\_purchase\_order\_dtl\_sno); |
| custom\_update\_purchase\_order\_dtl | UPDATE masters.purchase\_order\_dtl SET purchase\_order\_dtl\_sno = quote\_nullable(i\_purchase\_order\_dtl\_sno), purchase\_order\_sno = quote\_nullable(i\_purchase\_order\_sno), restaurant\_food\_sku\_sno = quote\_nullable(i\_restaurant\_food\_sku\_sno), order\_quantity = quote\_nullable(i\_order\_quantity), sku\_price = quote\_nullable(i\_sku\_price), partner\_food\_code = quote\_nullable(i\_partner\_food\_code), accepted\_quantity = quote\_nullable(i\_accepted\_quantity), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno) WHERE <custom\_conditions>; |
| delete\_purchase\_order\_dtl\_by\_id | DELETE FROM masters.purchase\_order\_dtl WHERE purchase\_order\_dtl\_sno = quote\_nullable(i\_purchase\_order\_dtl\_sno); |
| update\_purchase\_order\_dtl | UPDATE masters.purchase\_order\_dtl SET purchase\_order\_dtl\_sno = quote\_nullable(i\_purchase\_order\_dtl\_sno), purchase\_order\_sno = quote\_nullable(i\_purchase\_order\_sno), restaurant\_food\_sku\_sno = quote\_nullable(i\_restaurant\_food\_sku\_sno), order\_quantity = quote\_nullable(i\_order\_quantity), sku\_price = quote\_nullable(i\_sku\_price), partner\_food\_code = quote\_nullable(i\_partner\_food\_code), accepted\_quantity = quote\_nullable(i\_accepted\_quantity), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno) WHERE purchase\_order\_dtl\_sno = quote\_nullable(i\_purchase\_order\_dtl\_sno); |
| create\_purchase\_order\_dtl | INSERT INTO masters.purchase\_order\_dtl (t.purchase\_order\_dtl\_sno, t.purchase\_order\_sno, t.restaurant\_food\_sku\_sno, t.order\_quantity, t.sku\_price, t.partner\_food\_code, t.accepted\_quantity, t.description, t.active\_flag, t.qbox\_entity\_sno) VALUES (i\_purchase\_order\_dtl\_sno, i\_purchase\_order\_sno, i\_restaurant\_food\_sku\_sno, i\_order\_quantity, i\_sku\_price, i\_partner\_food\_code, i\_accepted\_quantity, i\_description, i\_active\_flag, i\_qbox\_entity\_sno); |
| unique\_purchase\_order\_dtl\_exists | SELECT 1 FROM masters.purchase\_order\_dtl WHERE <unique\_conditions> LIMIT 1; |
| custom\_delete\_purchase\_order\_dtl\_by\_id | DELETE FROM masters.purchase\_order\_dtl WHERE <custom\_conditions>; |
| custom\_search\_purchase\_order\_dtl\_by\_id | SELECT t.purchase\_order\_dtl\_sno, t.purchase\_order\_sno, t.restaurant\_food\_sku\_sno, t.order\_quantity, t.sku\_price, t.partner\_food\_code, t.accepted\_quantity, t.description, t.active\_flag, t.qbox\_entity\_sno, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_qbox\_entity.name, f\_purchase\_order.order\_status\_desc, f\_restaurant\_food\_sku.restaurant\_sku\_name FROM masters.purchase\_order\_dtl t LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.purchase\_order f\_purchase\_order ON t.purchase\_order\_sno = f\_purchase\_order.purchase\_order\_sno LEFT JOIN masters.restaurant\_food\_sku f\_restaurant\_food\_sku ON t.restaurant\_food\_sku\_sno = f\_restaurant\_food\_sku.restaurant\_food\_sku\_sno WHERE <custom\_search\_conditions>; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Restaurant Food Sku**

### **Database Table Name: restaurant\_food\_sku**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| restaurant\_food\_sku\_sno | bigint - (Numeric Precision: 64) | restaurant\_food\_sku\_pkey (PRIMARY KEY), |
| restaurant\_sno | bigint - (Numeric Precision: 64) | restaurant\_food\_sku\_restaurant\_sno\_fkey (FOREIGN KEY), |
| description | text | null |
| sku\_code | text | null |
| active\_flag | boolean | null |
| restaurant\_sku\_name | text | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Restaurant Food Sku Sno | hidden | number | true | Manual Entry |
| Restaurant Sno | listbox | number | true | API feed list box selection |
| Description | textbox | string | false | Manual Entry |
| Sku Code | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |
| Restaurant Sku Name | textbox | string | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_restaurant\_food\_skus | get | /api/v1/get\_restaurant\_food\_skus |
| create\_restaurant\_food\_sku | post | /api/v1/create\_restaurant\_food\_sku |
| update\_restaurant\_food\_sku | put | /api/v1/update\_restaurant\_food\_sku |
| delete\_restaurant\_food\_sku | delete | /api/v1/delete\_restaurant\_food\_sku |
| get\_restaurants | get | /api/v1/get\_restaurants |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_restaurant\_food\_skus | {} | {"data":{"restaurant\_food\_sku\_sno":"bigint","restaurant\_sno":"bigint","description":"text","restaurant\_sku\_name":"text","active\_flag":"boolean","sku\_code":"text"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_restaurant\_food\_sku | {"data":{"restaurant\_food\_sku\_sno":"bigint","restaurant\_sno":"bigint","description":"text","restaurant\_sku\_name":"text","active\_flag":"boolean","sku\_code":"text"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_restaurant\_food\_sku | {"data":{"restaurant\_food\_sku\_sno":"bigint","restaurant\_sno":"bigint","description":"text","restaurant\_sku\_name":"text","active\_flag":"boolean","sku\_code":"text"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_restaurant\_food\_sku | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_restaurants | {"id":"integer"} | {"data":"array of restaurant"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| custom\_search\_restaurant\_food\_sku\_by\_id | SELECT t.restaurant\_food\_sku\_sno, t.restaurant\_sno, t.description, t.sku\_code, t.active\_flag, t.restaurant\_sku\_name, f\_restaurant.name FROM masters.restaurant\_food\_sku t LEFT JOIN masters.restaurant f\_restaurant ON t.restaurant\_sno = f\_restaurant.restaurant\_sno WHERE <custom\_search\_conditions>; |
| create\_restaurant\_food\_sku | INSERT INTO masters.restaurant\_food\_sku (t.restaurant\_food\_sku\_sno, t.restaurant\_sno, t.description, t.sku\_code, t.active\_flag, t.restaurant\_sku\_name) VALUES (i\_restaurant\_food\_sku\_sno, i\_restaurant\_sno, i\_description, i\_sku\_code, i\_active\_flag, i\_restaurant\_sku\_name); |
| get\_restaurant\_food\_sku\_by\_id | SELECT t.restaurant\_food\_sku\_sno, t.restaurant\_sno, t.description, t.sku\_code, t.active\_flag, t.restaurant\_sku\_name, f\_restaurant.name FROM masters.restaurant\_food\_sku t LEFT JOIN masters.restaurant f\_restaurant ON t.restaurant\_sno = f\_restaurant.restaurant\_sno WHERE t.restaurant\_food\_sku\_sno = quote\_nullable(i\_restaurant\_food\_sku\_sno); |
| custom\_update\_restaurant\_food\_sku | UPDATE masters.restaurant\_food\_sku SET restaurant\_food\_sku\_sno = quote\_nullable(i\_restaurant\_food\_sku\_sno), restaurant\_sno = quote\_nullable(i\_restaurant\_sno), description = quote\_nullable(i\_description), sku\_code = quote\_nullable(i\_sku\_code), active\_flag = quote\_nullable(i\_active\_flag), restaurant\_sku\_name = quote\_nullable(i\_restaurant\_sku\_name) WHERE <custom\_conditions>; |
| update\_restaurant\_food\_sku | UPDATE masters.restaurant\_food\_sku SET restaurant\_food\_sku\_sno = quote\_nullable(i\_restaurant\_food\_sku\_sno), restaurant\_sno = quote\_nullable(i\_restaurant\_sno), description = quote\_nullable(i\_description), sku\_code = quote\_nullable(i\_sku\_code), active\_flag = quote\_nullable(i\_active\_flag), restaurant\_sku\_name = quote\_nullable(i\_restaurant\_sku\_name) WHERE restaurant\_food\_sku\_sno = quote\_nullable(i\_restaurant\_food\_sku\_sno); |
| get\_all\_restaurant\_food\_skus | SELECT t.restaurant\_food\_sku\_sno, t.restaurant\_sno, t.description, t.sku\_code, t.active\_flag, t.restaurant\_sku\_name, f\_restaurant.name FROM masters.restaurant\_food\_sku t LEFT JOIN masters.restaurant f\_restaurant ON t.restaurant\_sno = f\_restaurant.restaurant\_sno ; |
| delete\_restaurant\_food\_sku\_by\_id | DELETE FROM masters.restaurant\_food\_sku WHERE restaurant\_food\_sku\_sno = quote\_nullable(i\_restaurant\_food\_sku\_sno); |
| unique\_restaurant\_food\_sku\_exists | SELECT 1 FROM masters.restaurant\_food\_sku WHERE <unique\_conditions> LIMIT 1; |
| custom\_delete\_restaurant\_food\_sku\_by\_id | DELETE FROM masters.restaurant\_food\_sku WHERE <custom\_conditions>; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Low Stock Trigger**

### **Database Table Name: low\_stock\_trigger**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| low\_stock\_trigger\_sno | bigint - (Numeric Precision: 64) | low\_stock\_trigger\_pkey (PRIMARY KEY), |
| restaurant\_sno | bigint - (Numeric Precision: 64) | low\_stock\_trigger\_restaurant\_sno\_fkey (FOREIGN KEY), |
| restaurant\_food\_sku\_sno | bigint - (Numeric Precision: 64) | low\_stock\_trigger\_restaurant\_food\_sku\_sno\_fkey (FOREIGN KEY), |
| qbox\_entity\_sno | bigint - (Numeric Precision: 64) | null |
| threshold\_quantity | numeric - (Numeric Precision: 10) | null |
| order\_triggered | boolean | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Low Stock Trigger Sno | hidden | number | true | Manual Entry |
| Restaurant Sno | listbox | number | true | API feed list box selection |
| Restaurant Food Sku Sno | listbox | number | true | API feed list box selection |
| Qbox Entity Sno | textbox\_number | number | false | Manual Entry |
| Threshold Quantity | textbox | number | true | Manual Entry |
| Order Triggered | checkbox | boolean | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_low\_stock\_triggers | get | /api/v1/get\_low\_stock\_triggers |
| create\_low\_stock\_trigger | post | /api/v1/create\_low\_stock\_trigger |
| update\_low\_stock\_trigger | put | /api/v1/update\_low\_stock\_trigger |
| delete\_low\_stock\_trigger | delete | /api/v1/delete\_low\_stock\_trigger |
| get\_restaurant\_food\_skus | get | /api/v1/get\_restaurant\_food\_skus |
| get\_restaurants | get | /api/v1/get\_restaurants |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_low\_stock\_triggers | {} | {"data":{"threshold\_quantity":"numeric","restaurant\_food\_sku\_sno":"bigint","restaurant\_sno":"bigint","qbox\_entity\_sno":"bigint","active\_flag":"boolean","low\_stock\_trigger\_sno":"bigint","order\_triggered":"boolean"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_low\_stock\_trigger | {"data":{"threshold\_quantity":"numeric","restaurant\_food\_sku\_sno":"bigint","restaurant\_sno":"bigint","qbox\_entity\_sno":"bigint","active\_flag":"boolean","low\_stock\_trigger\_sno":"bigint","order\_triggered":"boolean"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_low\_stock\_trigger | {"data":{"threshold\_quantity":"numeric","restaurant\_food\_sku\_sno":"bigint","restaurant\_sno":"bigint","qbox\_entity\_sno":"bigint","active\_flag":"boolean","low\_stock\_trigger\_sno":"bigint","order\_triggered":"boolean"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_low\_stock\_trigger | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_restaurant\_food\_skus | {"id":"integer"} | {"data":"array of restaurant\_food\_sku"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_restaurants | {"id":"integer"} | {"data":"array of restaurant"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| get\_low\_stock\_trigger\_by\_id | SELECT t.low\_stock\_trigger\_sno, t.restaurant\_sno, t.restaurant\_food\_sku\_sno, t.qbox\_entity\_sno, t.threshold\_quantity, t.order\_triggered, t.active\_flag, f\_restaurant\_food\_sku.restaurant\_sku\_name, f\_restaurant.name FROM masters.low\_stock\_trigger t LEFT JOIN masters.restaurant\_food\_sku f\_restaurant\_food\_sku ON t.restaurant\_food\_sku\_sno = f\_restaurant\_food\_sku.restaurant\_food\_sku\_sno LEFT JOIN masters.restaurant f\_restaurant ON t.restaurant\_sno = f\_restaurant.restaurant\_sno WHERE t.low\_stock\_trigger\_sno = quote\_nullable(i\_low\_stock\_trigger\_sno); |
| custom\_delete\_low\_stock\_trigger\_by\_id | DELETE FROM masters.low\_stock\_trigger WHERE <custom\_conditions>; |
| custom\_update\_low\_stock\_trigger | UPDATE masters.low\_stock\_trigger SET low\_stock\_trigger\_sno = quote\_nullable(i\_low\_stock\_trigger\_sno), restaurant\_sno = quote\_nullable(i\_restaurant\_sno), restaurant\_food\_sku\_sno = quote\_nullable(i\_restaurant\_food\_sku\_sno), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), threshold\_quantity = quote\_nullable(i\_threshold\_quantity), order\_triggered = quote\_nullable(i\_order\_triggered), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| get\_all\_low\_stock\_triggers | SELECT t.low\_stock\_trigger\_sno, t.restaurant\_sno, t.restaurant\_food\_sku\_sno, t.qbox\_entity\_sno, t.threshold\_quantity, t.order\_triggered, t.active\_flag, f\_restaurant\_food\_sku.restaurant\_sku\_name, f\_restaurant.name FROM masters.low\_stock\_trigger t LEFT JOIN masters.restaurant\_food\_sku f\_restaurant\_food\_sku ON t.restaurant\_food\_sku\_sno = f\_restaurant\_food\_sku.restaurant\_food\_sku\_sno LEFT JOIN masters.restaurant f\_restaurant ON t.restaurant\_sno = f\_restaurant.restaurant\_sno ; |
| create\_low\_stock\_trigger | INSERT INTO masters.low\_stock\_trigger (t.low\_stock\_trigger\_sno, t.restaurant\_sno, t.restaurant\_food\_sku\_sno, t.qbox\_entity\_sno, t.threshold\_quantity, t.order\_triggered, t.active\_flag) VALUES (i\_low\_stock\_trigger\_sno, i\_restaurant\_sno, i\_restaurant\_food\_sku\_sno, i\_qbox\_entity\_sno, i\_threshold\_quantity, i\_order\_triggered, i\_active\_flag); |
| update\_low\_stock\_trigger | UPDATE masters.low\_stock\_trigger SET low\_stock\_trigger\_sno = quote\_nullable(i\_low\_stock\_trigger\_sno), restaurant\_sno = quote\_nullable(i\_restaurant\_sno), restaurant\_food\_sku\_sno = quote\_nullable(i\_restaurant\_food\_sku\_sno), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), threshold\_quantity = quote\_nullable(i\_threshold\_quantity), order\_triggered = quote\_nullable(i\_order\_triggered), active\_flag = quote\_nullable(i\_active\_flag) WHERE low\_stock\_trigger\_sno = quote\_nullable(i\_low\_stock\_trigger\_sno); |
| custom\_search\_low\_stock\_trigger\_by\_id | SELECT t.low\_stock\_trigger\_sno, t.restaurant\_sno, t.restaurant\_food\_sku\_sno, t.qbox\_entity\_sno, t.threshold\_quantity, t.order\_triggered, t.active\_flag, f\_restaurant\_food\_sku.restaurant\_sku\_name, f\_restaurant.name FROM masters.low\_stock\_trigger t LEFT JOIN masters.restaurant\_food\_sku f\_restaurant\_food\_sku ON t.restaurant\_food\_sku\_sno = f\_restaurant\_food\_sku.restaurant\_food\_sku\_sno LEFT JOIN masters.restaurant f\_restaurant ON t.restaurant\_sno = f\_restaurant.restaurant\_sno WHERE <custom\_search\_conditions>; |
| delete\_low\_stock\_trigger\_by\_id | DELETE FROM masters.low\_stock\_trigger WHERE low\_stock\_trigger\_sno = quote\_nullable(i\_low\_stock\_trigger\_sno); |
| unique\_low\_stock\_trigger\_exists | SELECT 1 FROM masters.low\_stock\_trigger WHERE <unique\_conditions> LIMIT 1; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Restaurant**

### **Database Table Name: restaurant**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| restaurant\_sno | bigint - (Numeric Precision: 64) | restaurant\_pkey (PRIMARY KEY), |
| name | text | null |
| address\_sno | bigint - (Numeric Precision: 64) | restaurant\_address\_sno\_fkey (FOREIGN KEY), |
| area\_sno | bigint - (Numeric Precision: 64) | restaurant\_area\_sno\_fkey (FOREIGN KEY), |
| restaurant\_status\_cd | smallint - (Numeric Precision: 16) | null |
| city\_code | character varying | null |
| restaurant\_brand\_cd | character varying | null |
| active\_flag | boolean | null |
| restaurant\_code | character varying | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Restaurant Sno | hidden | number | true | Manual Entry |
| Name | textbox | string | true | Manual Entry |
| Address Sno | listbox | number | false | API feed list box selection |
| Area Sno | listbox | number | true | API feed list box selection |
| Restaurant Status Cd | textbox\_number | number | true | Manual Entry |
| City Code | textbox | string | false | Manual Entry |
| Restaurant Brand Cd | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |
| Restaurant Code | textbox | string | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_restaurants | get | /api/v1/get\_restaurants |
| create\_restaurant | post | /api/v1/create\_restaurant |
| update\_restaurant | put | /api/v1/update\_restaurant |
| delete\_restaurant | delete | /api/v1/delete\_restaurant |
| get\_addresss | get | /api/v1/get\_addresss |
| get\_areas | get | /api/v1/get\_areas |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_restaurants | {} | {"data":{"restaurant\_brand\_cd":"character varying","restaurant\_code":"character varying","address\_sno":"bigint","area\_sno":"bigint","restaurant\_sno":"bigint","name":"text","city\_code":"character varying","active\_flag":"boolean","restaurant\_status\_cd":"smallint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_restaurant | {"data":{"restaurant\_brand\_cd":"character varying","restaurant\_code":"character varying","address\_sno":"bigint","area\_sno":"bigint","restaurant\_sno":"bigint","name":"text","city\_code":"character varying","active\_flag":"boolean","restaurant\_status\_cd":"smallint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_restaurant | {"data":{"restaurant\_brand\_cd":"character varying","restaurant\_code":"character varying","address\_sno":"bigint","area\_sno":"bigint","restaurant\_sno":"bigint","name":"text","city\_code":"character varying","active\_flag":"boolean","restaurant\_status\_cd":"smallint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_restaurant | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_addresss | {"id":"integer"} | {"data":"array of address"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_areas | {"id":"integer"} | {"data":"array of area"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| custom\_delete\_restaurant\_by\_id | DELETE FROM masters.restaurant WHERE <custom\_conditions>; |
| update\_restaurant | UPDATE masters.restaurant SET restaurant\_sno = quote\_nullable(i\_restaurant\_sno), name = quote\_nullable(i\_name), address\_sno = quote\_nullable(i\_address\_sno), area\_sno = quote\_nullable(i\_area\_sno), restaurant\_status\_cd = quote\_nullable(i\_restaurant\_status\_cd), city\_code = quote\_nullable(i\_city\_code), restaurant\_brand\_cd = quote\_nullable(i\_restaurant\_brand\_cd), active\_flag = quote\_nullable(i\_active\_flag), restaurant\_code = quote\_nullable(i\_restaurant\_code) WHERE restaurant\_sno = quote\_nullable(i\_restaurant\_sno); |
| get\_all\_restaurants | SELECT t.restaurant\_sno, t.name, t.address\_sno, t.area\_sno, t.restaurant\_status\_cd, t.city\_code, t.restaurant\_brand\_cd, t.active\_flag, t.restaurant\_code, f\_area.name FROM masters.restaurant t LEFT JOIN masters.area f\_area ON t.area\_sno = f\_area.area\_sno ; |
| custom\_search\_restaurant\_by\_id | SELECT t.restaurant\_sno, t.name, t.address\_sno, t.area\_sno, t.restaurant\_status\_cd, t.city\_code, t.restaurant\_brand\_cd, t.active\_flag, t.restaurant\_code, f\_area.name FROM masters.restaurant t LEFT JOIN masters.area f\_area ON t.area\_sno = f\_area.area\_sno WHERE <custom\_search\_conditions>; |
| custom\_update\_restaurant | UPDATE masters.restaurant SET restaurant\_sno = quote\_nullable(i\_restaurant\_sno), name = quote\_nullable(i\_name), address\_sno = quote\_nullable(i\_address\_sno), area\_sno = quote\_nullable(i\_area\_sno), restaurant\_status\_cd = quote\_nullable(i\_restaurant\_status\_cd), city\_code = quote\_nullable(i\_city\_code), restaurant\_brand\_cd = quote\_nullable(i\_restaurant\_brand\_cd), active\_flag = quote\_nullable(i\_active\_flag), restaurant\_code = quote\_nullable(i\_restaurant\_code) WHERE <custom\_conditions>; |
| create\_restaurant | INSERT INTO masters.restaurant (t.restaurant\_sno, t.name, t.address\_sno, t.area\_sno, t.restaurant\_status\_cd, t.city\_code, t.restaurant\_brand\_cd, t.active\_flag, t.restaurant\_code) VALUES (i\_restaurant\_sno, i\_name, i\_address\_sno, i\_area\_sno, i\_restaurant\_status\_cd, i\_city\_code, i\_restaurant\_brand\_cd, i\_active\_flag, i\_restaurant\_code); |
| delete\_restaurant\_by\_id | DELETE FROM masters.restaurant WHERE restaurant\_sno = quote\_nullable(i\_restaurant\_sno); |
| get\_restaurant\_by\_id | SELECT t.restaurant\_sno, t.name, t.address\_sno, t.area\_sno, t.restaurant\_status\_cd, t.city\_code, t.restaurant\_brand\_cd, t.active\_flag, t.restaurant\_code, f\_area.name FROM masters.restaurant t LEFT JOIN masters.area f\_area ON t.area\_sno = f\_area.area\_sno WHERE t.restaurant\_sno = quote\_nullable(i\_restaurant\_sno); |
| unique\_restaurant\_exists | SELECT 1 FROM masters.restaurant WHERE <unique\_conditions> LIMIT 1; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: App Permission**

### **Database Table Name: app\_permission**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| app\_permission\_sno | bigint - (Numeric Precision: 64) | app\_function\_pkey (PRIMARY KEY), |
| name | text | null |
| created\_at | timestamp without time zone | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| App Permission Sno | hidden | number | true | Manual Entry |
| Name | textbox | string | true | Manual Entry |
| Created At | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_app\_permissions | get | /api/v1/get\_app\_permissions |
| create\_app\_permission | post | /api/v1/create\_app\_permission |
| update\_app\_permission | put | /api/v1/update\_app\_permission |
| delete\_app\_permission | delete | /api/v1/delete\_app\_permission |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_app\_permissions | {} | {"data":{"app\_permission\_sno":"bigint","name":"text","created\_at":"timestamp without time zone","active\_flag":"boolean"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_app\_permission | {"data":{"app\_permission\_sno":"bigint","name":"text","created\_at":"timestamp without time zone","active\_flag":"boolean"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_app\_permission | {"data":{"app\_permission\_sno":"bigint","name":"text","created\_at":"timestamp without time zone","active\_flag":"boolean"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_app\_permission | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| create\_app\_permission | INSERT INTO masters.app\_permission (t.app\_permission\_sno, t.name, t.created\_at, t.active\_flag) VALUES (i\_app\_permission\_sno, i\_name, i\_created\_at, i\_active\_flag); |
| get\_app\_permission\_by\_id | SELECT t.app\_permission\_sno, t.name, t.created\_at, t.active\_flag FROM masters.app\_permission t WHERE t.app\_permission\_sno = quote\_nullable(i\_app\_permission\_sno); |
| custom\_update\_app\_permission | UPDATE masters.app\_permission SET app\_permission\_sno = quote\_nullable(i\_app\_permission\_sno), name = quote\_nullable(i\_name), created\_at = quote\_nullable(i\_created\_at), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| delete\_app\_permission\_by\_id | DELETE FROM masters.app\_permission WHERE app\_permission\_sno = quote\_nullable(i\_app\_permission\_sno); |
| update\_app\_permission | UPDATE masters.app\_permission SET app\_permission\_sno = quote\_nullable(i\_app\_permission\_sno), name = quote\_nullable(i\_name), created\_at = quote\_nullable(i\_created\_at), active\_flag = quote\_nullable(i\_active\_flag) WHERE app\_permission\_sno = quote\_nullable(i\_app\_permission\_sno); |
| unique\_app\_permission\_exists | SELECT 1 FROM masters.app\_permission WHERE <unique\_conditions> LIMIT 1; |
| get\_all\_app\_permissions | SELECT t.app\_permission\_sno, t.name, t.created\_at, t.active\_flag FROM masters.app\_permission t ; |
| custom\_delete\_app\_permission\_by\_id | DELETE FROM masters.app\_permission WHERE <custom\_conditions>; |
| custom\_search\_app\_permission\_by\_id | SELECT t.app\_permission\_sno, t.name, t.created\_at, t.active\_flag FROM masters.app\_permission t WHERE <custom\_search\_conditions>; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Menu Permission**

### **Database Table Name: menu\_permission**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| menu\_permission\_sno | smallint - (Numeric Precision: 16) | menu\_function\_pkey (PRIMARY KEY), |
| app\_menu\_sno | smallint - (Numeric Precision: 16) | menu\_function\_app\_menu\_sno\_fkey (FOREIGN KEY), |
| app\_permission\_sno | smallint - (Numeric Precision: 16) | menu\_function\_app\_function\_sno\_fkey (FOREIGN KEY), |
| status | boolean | null |
| description | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Menu Permission Sno | hidden | number | true | Manual Entry |
| App Menu Sno | listbox | number | true | API feed list box selection |
| App Permission Sno | listbox | number | true | API feed list box selection |
| Status | checkbox | boolean | false | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_menu\_permissions | get | /api/v1/get\_menu\_permissions |
| create\_menu\_permission | post | /api/v1/create\_menu\_permission |
| update\_menu\_permission | put | /api/v1/update\_menu\_permission |
| delete\_menu\_permission | delete | /api/v1/delete\_menu\_permission |
| get\_app\_permissions | get | /api/v1/get\_app\_permissions |
| get\_app\_menus | get | /api/v1/get\_app\_menus |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_menu\_permissions | {} | {"data":{"app\_permission\_sno":"smallint","description":"text","active\_flag":"boolean","app\_menu\_sno":"smallint","status":"boolean","menu\_permission\_sno":"smallint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_menu\_permission | {"data":{"app\_permission\_sno":"smallint","description":"text","active\_flag":"boolean","app\_menu\_sno":"smallint","status":"boolean","menu\_permission\_sno":"smallint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_menu\_permission | {"data":{"app\_permission\_sno":"smallint","description":"text","active\_flag":"boolean","app\_menu\_sno":"smallint","status":"boolean","menu\_permission\_sno":"smallint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_menu\_permission | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_app\_permissions | {"id":"integer"} | {"data":"array of app\_permission"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_app\_menus | {"id":"integer"} | {"data":"array of app\_menu"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| unique\_menu\_permission\_exists | SELECT 1 FROM masters.menu\_permission WHERE <unique\_conditions> LIMIT 1; |
| custom\_search\_menu\_permission\_by\_id | SELECT t.menu\_permission\_sno, t.app\_menu\_sno, t.app\_permission\_sno, t.status, t.description, t.active\_flag, f\_app\_permission.name, f\_app\_menu.name FROM masters.menu\_permission t LEFT JOIN masters.app\_permission f\_app\_permission ON t.app\_permission\_sno = f\_app\_permission.app\_permission\_sno LEFT JOIN masters.app\_menu f\_app\_menu ON t.app\_menu\_sno = f\_app\_menu.app\_menu\_sno WHERE <custom\_search\_conditions>; |
| delete\_menu\_permission\_by\_id | DELETE FROM masters.menu\_permission WHERE menu\_permission\_sno = quote\_nullable(i\_menu\_permission\_sno); |
| custom\_update\_menu\_permission | UPDATE masters.menu\_permission SET menu\_permission\_sno = quote\_nullable(i\_menu\_permission\_sno), app\_menu\_sno = quote\_nullable(i\_app\_menu\_sno), app\_permission\_sno = quote\_nullable(i\_app\_permission\_sno), status = quote\_nullable(i\_status), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| update\_menu\_permission | UPDATE masters.menu\_permission SET menu\_permission\_sno = quote\_nullable(i\_menu\_permission\_sno), app\_menu\_sno = quote\_nullable(i\_app\_menu\_sno), app\_permission\_sno = quote\_nullable(i\_app\_permission\_sno), status = quote\_nullable(i\_status), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE menu\_permission\_sno = quote\_nullable(i\_menu\_permission\_sno); |
| create\_menu\_permission | INSERT INTO masters.menu\_permission (t.menu\_permission\_sno, t.app\_menu\_sno, t.app\_permission\_sno, t.status, t.description, t.active\_flag) VALUES (i\_menu\_permission\_sno, i\_app\_menu\_sno, i\_app\_permission\_sno, i\_status, i\_description, i\_active\_flag); |
| get\_menu\_permission\_by\_id | SELECT t.menu\_permission\_sno, t.app\_menu\_sno, t.app\_permission\_sno, t.status, t.description, t.active\_flag, f\_app\_permission.name, f\_app\_menu.name FROM masters.menu\_permission t LEFT JOIN masters.app\_permission f\_app\_permission ON t.app\_permission\_sno = f\_app\_permission.app\_permission\_sno LEFT JOIN masters.app\_menu f\_app\_menu ON t.app\_menu\_sno = f\_app\_menu.app\_menu\_sno WHERE t.menu\_permission\_sno = quote\_nullable(i\_menu\_permission\_sno); |
| custom\_delete\_menu\_permission\_by\_id | DELETE FROM masters.menu\_permission WHERE <custom\_conditions>; |
| get\_all\_menu\_permissions | SELECT t.menu\_permission\_sno, t.app\_menu\_sno, t.app\_permission\_sno, t.status, t.description, t.active\_flag, f\_app\_permission.name, f\_app\_menu.name FROM masters.menu\_permission t LEFT JOIN masters.app\_permission f\_app\_permission ON t.app\_permission\_sno = f\_app\_permission.app\_permission\_sno LEFT JOIN masters.app\_menu f\_app\_menu ON t.app\_menu\_sno = f\_app\_menu.app\_menu\_sno ; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: App Menu**

### **Database Table Name: app\_menu**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| app\_menu\_sno | smallint - (Numeric Precision: 16) | app\_menu\_pkey (PRIMARY KEY), |
| name | text | null |
| parent\_menu\_sno | smallint - (Numeric Precision: 16) | null |
| href | text | null |
| title | text | null |
| status | boolean | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| App Menu Sno | hidden | number | true | Manual Entry |
| Name | textbox | string | true | Manual Entry |
| Parent Menu Sno | textbox\_number | number | true | Manual Entry |
| Href | textbox | string | true | Manual Entry |
| Title | textbox | string | true | Manual Entry |
| Status | checkbox | boolean | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_app\_menus | get | /api/v1/get\_app\_menus |
| create\_app\_menu | post | /api/v1/create\_app\_menu |
| update\_app\_menu | put | /api/v1/update\_app\_menu |
| delete\_app\_menu | delete | /api/v1/delete\_app\_menu |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_app\_menus | {} | {"data":{"name":"text","parent\_menu\_sno":"smallint","active\_flag":"boolean","href":"text","title":"text","app\_menu\_sno":"smallint","status":"boolean"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_app\_menu | {"data":{"name":"text","parent\_menu\_sno":"smallint","active\_flag":"boolean","href":"text","title":"text","app\_menu\_sno":"smallint","status":"boolean"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_app\_menu | {"data":{"name":"text","parent\_menu\_sno":"smallint","active\_flag":"boolean","href":"text","title":"text","app\_menu\_sno":"smallint","status":"boolean"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_app\_menu | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| custom\_delete\_app\_menu\_by\_id | DELETE FROM masters.app\_menu WHERE <custom\_conditions>; |
| delete\_app\_menu\_by\_id | DELETE FROM masters.app\_menu WHERE app\_menu\_sno = quote\_nullable(i\_app\_menu\_sno); |
| update\_app\_menu | UPDATE masters.app\_menu SET app\_menu\_sno = quote\_nullable(i\_app\_menu\_sno), name = quote\_nullable(i\_name), parent\_menu\_sno = quote\_nullable(i\_parent\_menu\_sno), href = quote\_nullable(i\_href), title = quote\_nullable(i\_title), status = quote\_nullable(i\_status), active\_flag = quote\_nullable(i\_active\_flag) WHERE app\_menu\_sno = quote\_nullable(i\_app\_menu\_sno); |
| get\_all\_app\_menus | SELECT t.app\_menu\_sno, t.name, t.parent\_menu\_sno, t.href, t.title, t.status, t.active\_flag FROM masters.app\_menu t ; |
| custom\_update\_app\_menu | UPDATE masters.app\_menu SET app\_menu\_sno = quote\_nullable(i\_app\_menu\_sno), name = quote\_nullable(i\_name), parent\_menu\_sno = quote\_nullable(i\_parent\_menu\_sno), href = quote\_nullable(i\_href), title = quote\_nullable(i\_title), status = quote\_nullable(i\_status), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| create\_app\_menu | INSERT INTO masters.app\_menu (t.app\_menu\_sno, t.name, t.parent\_menu\_sno, t.href, t.title, t.status, t.active\_flag) VALUES (i\_app\_menu\_sno, i\_name, i\_parent\_menu\_sno, i\_href, i\_title, i\_status, i\_active\_flag); |
| custom\_search\_app\_menu\_by\_id | SELECT t.app\_menu\_sno, t.name, t.parent\_menu\_sno, t.href, t.title, t.status, t.active\_flag FROM masters.app\_menu t WHERE <custom\_search\_conditions>; |
| unique\_app\_menu\_exists | SELECT 1 FROM masters.app\_menu WHERE <unique\_conditions> LIMIT 1; |
| get\_app\_menu\_by\_id | SELECT t.app\_menu\_sno, t.name, t.parent\_menu\_sno, t.href, t.title, t.status, t.active\_flag FROM masters.app\_menu t WHERE t.app\_menu\_sno = quote\_nullable(i\_app\_menu\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Otp**

### **Database Table Name: otp**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| otp\_sno | bigint - (Numeric Precision: 64) | otp\_pkey (PRIMARY KEY), |
| app\_user\_sno | bigint - (Numeric Precision: 64) | otp\_app\_user\_sno\_fkey (FOREIGN KEY), |
| sms\_otp | character varying | null |
| service\_otp | text | null |
| push\_otp | text | null |
| device\_id | text | null |
| otp\_expire\_time\_cd | smallint - (Numeric Precision: 16) | otp\_otp\_expire\_time\_cd\_fkey (FOREIGN KEY), |
| otp\_expire\_time | timestamp without time zone | null |
| otp\_status | boolean | null |
| description | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Otp Sno | hidden | number | true | Manual Entry |
| App User Sno | listbox | number | false | API feed list box selection |
| Sms Otp | textbox | string | false | Manual Entry |
| Service Otp | textbox | string | false | Manual Entry |
| Push Otp | textbox | string | false | Manual Entry |
| Device Id | textbox | string | false | Manual Entry |
| Otp Expire Time Cd | listbox | number | false | API feed list box selection |
| Otp Expire Time | textbox | string | false | Manual Entry |
| Otp Status | checkbox | boolean | false | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_otps | get | /api/v1/get\_otps |
| create\_otp | post | /api/v1/create\_otp |
| update\_otp | put | /api/v1/update\_otp |
| delete\_otp | delete | /api/v1/delete\_otp |
| get\_app\_users | get | /api/v1/get\_app\_users |
| get\_codes\_dtls | get | /api/v1/get\_codes\_dtls |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_otps | {} | {"data":{"device\_id":"text","service\_otp":"text","push\_otp":"text","description":"text","app\_user\_sno":"bigint","active\_flag":"boolean","sms\_otp":"character varying","otp\_expire\_time\_cd":"smallint","otp\_sno":"bigint","otp\_status":"boolean","otp\_expire\_time":"timestamp without time zone"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_otp | {"data":{"device\_id":"text","service\_otp":"text","push\_otp":"text","description":"text","app\_user\_sno":"bigint","active\_flag":"boolean","sms\_otp":"character varying","otp\_expire\_time\_cd":"smallint","otp\_sno":"bigint","otp\_status":"boolean","otp\_expire\_time":"timestamp without time zone"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_otp | {"data":{"device\_id":"text","service\_otp":"text","push\_otp":"text","description":"text","app\_user\_sno":"bigint","active\_flag":"boolean","sms\_otp":"character varying","otp\_expire\_time\_cd":"smallint","otp\_sno":"bigint","otp\_status":"boolean","otp\_expire\_time":"timestamp without time zone"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_otp | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_app\_users | {"id":"integer"} | {"data":"array of app\_user"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_codes\_dtls | {"id":"integer"} | {"data":"array of codes\_dtl"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| unique\_otp\_exists | SELECT 1 FROM masters.otp WHERE <unique\_conditions> LIMIT 1; |
| get\_all\_otps | SELECT t.otp\_sno, t.app\_user\_sno, t.sms\_otp, t.service\_otp, t.push\_otp, t.device\_id, t.otp\_expire\_time\_cd, t.otp\_expire\_time, t.otp\_status, t.description, t.active\_flag, f\_app\_user.name FROM masters.otp t LEFT JOIN masters.app\_user f\_app\_user ON t.app\_user\_sno = f\_app\_user.app\_user\_sno ; |
| update\_otp | UPDATE masters.otp SET otp\_sno = quote\_nullable(i\_otp\_sno), app\_user\_sno = quote\_nullable(i\_app\_user\_sno), sms\_otp = quote\_nullable(i\_sms\_otp), service\_otp = quote\_nullable(i\_service\_otp), push\_otp = quote\_nullable(i\_push\_otp), device\_id = quote\_nullable(i\_device\_id), otp\_expire\_time\_cd = quote\_nullable(i\_otp\_expire\_time\_cd), otp\_expire\_time = quote\_nullable(i\_otp\_expire\_time), otp\_status = quote\_nullable(i\_otp\_status), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE otp\_sno = quote\_nullable(i\_otp\_sno); |
| custom\_delete\_otp\_by\_id | DELETE FROM masters.otp WHERE <custom\_conditions>; |
| create\_otp | INSERT INTO masters.otp (t.otp\_sno, t.app\_user\_sno, t.sms\_otp, t.service\_otp, t.push\_otp, t.device\_id, t.otp\_expire\_time\_cd, t.otp\_expire\_time, t.otp\_status, t.description, t.active\_flag) VALUES (i\_otp\_sno, i\_app\_user\_sno, i\_sms\_otp, i\_service\_otp, i\_push\_otp, i\_device\_id, i\_otp\_expire\_time\_cd, i\_otp\_expire\_time, i\_otp\_status, i\_description, i\_active\_flag); |
| delete\_otp\_by\_id | DELETE FROM masters.otp WHERE otp\_sno = quote\_nullable(i\_otp\_sno); |
| custom\_search\_otp\_by\_id | SELECT t.otp\_sno, t.app\_user\_sno, t.sms\_otp, t.service\_otp, t.push\_otp, t.device\_id, t.otp\_expire\_time\_cd, t.otp\_expire\_time, t.otp\_status, t.description, t.active\_flag, f\_app\_user.name FROM masters.otp t LEFT JOIN masters.app\_user f\_app\_user ON t.app\_user\_sno = f\_app\_user.app\_user\_sno WHERE <custom\_search\_conditions>; |
| custom\_update\_otp | UPDATE masters.otp SET otp\_sno = quote\_nullable(i\_otp\_sno), app\_user\_sno = quote\_nullable(i\_app\_user\_sno), sms\_otp = quote\_nullable(i\_sms\_otp), service\_otp = quote\_nullable(i\_service\_otp), push\_otp = quote\_nullable(i\_push\_otp), device\_id = quote\_nullable(i\_device\_id), otp\_expire\_time\_cd = quote\_nullable(i\_otp\_expire\_time\_cd), otp\_expire\_time = quote\_nullable(i\_otp\_expire\_time), otp\_status = quote\_nullable(i\_otp\_status), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| get\_otp\_by\_id | SELECT t.otp\_sno, t.app\_user\_sno, t.sms\_otp, t.service\_otp, t.push\_otp, t.device\_id, t.otp\_expire\_time\_cd, t.otp\_expire\_time, t.otp\_status, t.description, t.active\_flag, f\_app\_user.name FROM masters.otp t LEFT JOIN masters.app\_user f\_app\_user ON t.app\_user\_sno = f\_app\_user.app\_user\_sno WHERE t.otp\_sno = quote\_nullable(i\_otp\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Profile**

### **Database Table Name: profile**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| profile\_id | bigint - (Numeric Precision: 64) | profile\_pkey (PRIMARY KEY), |
| profile\_name | character varying | null |
| contact | character varying | null |
| aadhar\_number | character | null |
| profile\_image | bigint - (Numeric Precision: 64) | null |
| auth\_user\_id | integer | profile\_auth\_user\_id\_fkey (FOREIGN KEY), |
| supervisor\_sno | bigint - (Numeric Precision: 64) | null |
| shift\_time | text | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Profile Id | hidden | number | true | Manual Entry |
| Profile Name | textbox | string | true | Manual Entry |
| Contact | textbox | string | true | Manual Entry |
| Aadhar Number | textbox | unknown | true | Manual Entry |
| Profile Image | textbox\_number | number | false | Manual Entry |
| Auth User Id | listbox | number | false | API feed list box selection |
| Supervisor Sno | textbox\_number | number | false | Manual Entry |
| Shift Time | textbox | string | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_profiles | get | /api/v1/get\_profiles |
| create\_profile | post | /api/v1/create\_profile |
| update\_profile | put | /api/v1/update\_profile |
| delete\_profile | delete | /api/v1/delete\_profile |
| get\_auth\_users | get | /api/v1/get\_auth\_users |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_profiles | {} | {"data":{"profile\_name":"character varying","profile\_image":"bigint","profile\_id":"bigint","supervisor\_sno":"bigint","contact":"character varying","auth\_user\_id":"integer","aadhar\_number":"character","shift\_time":"text"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_profile | {"data":{"profile\_name":"character varying","profile\_image":"bigint","profile\_id":"bigint","supervisor\_sno":"bigint","contact":"character varying","auth\_user\_id":"integer","aadhar\_number":"character","shift\_time":"text"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_profile | {"data":{"profile\_name":"character varying","profile\_image":"bigint","profile\_id":"bigint","supervisor\_sno":"bigint","contact":"character varying","auth\_user\_id":"integer","aadhar\_number":"character","shift\_time":"text"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_profile | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_auth\_users | {"id":"integer"} | {"data":"array of auth\_user"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| custom\_search\_profile\_by\_id | SELECT t.profile\_id, t.profile\_name, t.contact, t.aadhar\_number, t.profile\_image, t.auth\_user\_id, t.supervisor\_sno, t.shift\_time FROM masters.profile t WHERE <custom\_search\_conditions>; |
| update\_profile | UPDATE masters.profile SET profile\_id = quote\_nullable(i\_profile\_id), profile\_name = quote\_nullable(i\_profile\_name), contact = quote\_nullable(i\_contact), aadhar\_number = quote\_nullable(i\_aadhar\_number), profile\_image = quote\_nullable(i\_profile\_image), auth\_user\_id = quote\_nullable(i\_auth\_user\_id), supervisor\_sno = quote\_nullable(i\_supervisor\_sno), shift\_time = quote\_nullable(i\_shift\_time) WHERE profile\_id = quote\_nullable(i\_profile\_id); |
| custom\_update\_profile | UPDATE masters.profile SET profile\_id = quote\_nullable(i\_profile\_id), profile\_name = quote\_nullable(i\_profile\_name), contact = quote\_nullable(i\_contact), aadhar\_number = quote\_nullable(i\_aadhar\_number), profile\_image = quote\_nullable(i\_profile\_image), auth\_user\_id = quote\_nullable(i\_auth\_user\_id), supervisor\_sno = quote\_nullable(i\_supervisor\_sno), shift\_time = quote\_nullable(i\_shift\_time) WHERE <custom\_conditions>; |
| unique\_profile\_exists | SELECT 1 FROM masters.profile WHERE <unique\_conditions> LIMIT 1; |
| create\_profile | INSERT INTO masters.profile (t.profile\_id, t.profile\_name, t.contact, t.aadhar\_number, t.profile\_image, t.auth\_user\_id, t.supervisor\_sno, t.shift\_time) VALUES (i\_profile\_id, i\_profile\_name, i\_contact, i\_aadhar\_number, i\_profile\_image, i\_auth\_user\_id, i\_supervisor\_sno, i\_shift\_time); |
| delete\_profile\_by\_id | DELETE FROM masters.profile WHERE profile\_id = quote\_nullable(i\_profile\_id); |
| custom\_delete\_profile\_by\_id | DELETE FROM masters.profile WHERE <custom\_conditions>; |
| get\_profile\_by\_id | SELECT t.profile\_id, t.profile\_name, t.contact, t.aadhar\_number, t.profile\_image, t.auth\_user\_id, t.supervisor\_sno, t.shift\_time FROM masters.profile t WHERE t.profile\_id = quote\_nullable(i\_profile\_id); |
| get\_all\_profiles | SELECT t.profile\_id, t.profile\_name, t.contact, t.aadhar\_number, t.profile\_image, t.auth\_user\_id, t.supervisor\_sno, t.shift\_time FROM masters.profile t ; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Purchase Order**

### **Database Table Name: purchase\_order**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| purchase\_order\_sno | bigint - (Numeric Precision: 64) | purchase\_order\_pkey (PRIMARY KEY), |
| qbox\_entity\_sno | bigint - (Numeric Precision: 64) | purchase\_order\_qbox\_entity\_sno\_fkey (FOREIGN KEY), |
| restaurant\_sno | bigint - (Numeric Precision: 64) | purchase\_order\_restaurant\_sno\_fkey (FOREIGN KEY), |
| delivery\_partner\_sno | smallint - (Numeric Precision: 16) | purchase\_order\_delivery\_partner\_sno\_fkey (FOREIGN KEY), |
| order\_status\_cd | smallint - (Numeric Precision: 16) | null |
| ordered\_time | timestamp without time zone | null |
| ordered\_by | bigint - (Numeric Precision: 64) | null |
| partner\_purchase\_order\_id | text | partner\_purchase\_order\_id\_uk (UNIQUE), |
| meal\_time\_cd | smallint - (Numeric Precision: 16) | null |
| description | text | null |
| active\_flag | boolean | null |
| etl\_job\_sno | integer | null |
| transaction\_date | date | null |
| order\_status\_desc | character varying | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Purchase Order Sno | hidden | number | true | Manual Entry |
| Qbox Entity Sno | listbox | number | true | API feed list box selection |
| Restaurant Sno | listbox | number | true | API feed list box selection |
| Delivery Partner Sno | listbox | number | false | API feed list box selection |
| Order Status Cd | textbox\_number | number | true | Manual Entry |
| Ordered Time | textbox | string | false | Manual Entry |
| Ordered By | textbox\_number | number | true | Manual Entry |
| Partner Purchase Order Id | textbox | string | true | Manual Entry |
| Meal Time Cd | textbox\_number | number | false | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |
| Etl Job Sno | textbox | number | false | Manual Entry |
| Transaction Date | timestamp | string | false | Manual Entry |
| Order Status Desc | textbox | string | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_purchase\_orders | get | /api/v1/get\_purchase\_orders |
| create\_purchase\_order | post | /api/v1/create\_purchase\_order |
| update\_purchase\_order | put | /api/v1/update\_purchase\_order |
| delete\_purchase\_order | delete | /api/v1/delete\_purchase\_order |
| get\_delivery\_partners | get | /api/v1/get\_delivery\_partners |
| get\_qbox\_entitys | get | /api/v1/get\_qbox\_entitys |
| get\_restaurants | get | /api/v1/get\_restaurants |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_purchase\_orders | {} | {"data":{"ordered\_time":"timestamp without time zone","transaction\_date":"date","order\_status\_cd":"smallint","delivery\_partner\_sno":"smallint","description":"text","qbox\_entity\_sno":"bigint","ordered\_by":"bigint","active\_flag":"boolean","etl\_job\_sno":"integer","restaurant\_sno":"bigint","meal\_time\_cd":"smallint","partner\_purchase\_order\_id":"text","order\_status\_desc":"character varying","purchase\_order\_sno":"bigint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_purchase\_order | {"data":{"ordered\_time":"timestamp without time zone","transaction\_date":"date","order\_status\_cd":"smallint","delivery\_partner\_sno":"smallint","description":"text","qbox\_entity\_sno":"bigint","ordered\_by":"bigint","active\_flag":"boolean","etl\_job\_sno":"integer","restaurant\_sno":"bigint","meal\_time\_cd":"smallint","partner\_purchase\_order\_id":"text","order\_status\_desc":"character varying","purchase\_order\_sno":"bigint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_purchase\_order | {"data":{"ordered\_time":"timestamp without time zone","transaction\_date":"date","order\_status\_cd":"smallint","delivery\_partner\_sno":"smallint","description":"text","qbox\_entity\_sno":"bigint","ordered\_by":"bigint","active\_flag":"boolean","etl\_job\_sno":"integer","restaurant\_sno":"bigint","meal\_time\_cd":"smallint","partner\_purchase\_order\_id":"text","order\_status\_desc":"character varying","purchase\_order\_sno":"bigint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_purchase\_order | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_delivery\_partners | {"id":"integer"} | {"data":"array of delivery\_partner"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_qbox\_entitys | {"id":"integer"} | {"data":"array of qbox\_entity"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_restaurants | {"id":"integer"} | {"data":"array of restaurant"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| unique\_purchase\_order\_exists | SELECT 1 FROM masters.purchase\_order WHERE <unique\_conditions> LIMIT 1; |
| delete\_purchase\_order\_by\_id | DELETE FROM masters.purchase\_order WHERE purchase\_order\_sno = quote\_nullable(i\_purchase\_order\_sno); |
| custom\_delete\_purchase\_order\_by\_id | DELETE FROM masters.purchase\_order WHERE <custom\_conditions>; |
| update\_purchase\_order | UPDATE masters.purchase\_order SET purchase\_order\_sno = quote\_nullable(i\_purchase\_order\_sno), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), restaurant\_sno = quote\_nullable(i\_restaurant\_sno), delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno), order\_status\_cd = quote\_nullable(i\_order\_status\_cd), ordered\_time = quote\_nullable(i\_ordered\_time), ordered\_by = quote\_nullable(i\_ordered\_by), partner\_purchase\_order\_id = quote\_nullable(i\_partner\_purchase\_order\_id), meal\_time\_cd = quote\_nullable(i\_meal\_time\_cd), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), etl\_job\_sno = quote\_nullable(i\_etl\_job\_sno), transaction\_date = quote\_nullable(i\_transaction\_date), order\_status\_desc = quote\_nullable(i\_order\_status\_desc) WHERE purchase\_order\_sno = quote\_nullable(i\_purchase\_order\_sno); |
| get\_purchase\_order\_by\_id | SELECT t.purchase\_order\_sno, t.qbox\_entity\_sno, t.restaurant\_sno, t.delivery\_partner\_sno, t.order\_status\_cd, t.ordered\_time, t.ordered\_by, t.partner\_purchase\_order\_id, t.meal\_time\_cd, t.description, t.active\_flag, t.etl\_job\_sno, t.transaction\_date, t.order\_status\_desc, f\_delivery\_partner.name, f\_qbox\_entity.name, f\_restaurant.name FROM masters.purchase\_order t LEFT JOIN masters.delivery\_partner f\_delivery\_partner ON t.delivery\_partner\_sno = f\_delivery\_partner.delivery\_partner\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.restaurant f\_restaurant ON t.restaurant\_sno = f\_restaurant.restaurant\_sno WHERE t.purchase\_order\_sno = quote\_nullable(i\_purchase\_order\_sno); |
| custom\_update\_purchase\_order | UPDATE masters.purchase\_order SET purchase\_order\_sno = quote\_nullable(i\_purchase\_order\_sno), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), restaurant\_sno = quote\_nullable(i\_restaurant\_sno), delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno), order\_status\_cd = quote\_nullable(i\_order\_status\_cd), ordered\_time = quote\_nullable(i\_ordered\_time), ordered\_by = quote\_nullable(i\_ordered\_by), partner\_purchase\_order\_id = quote\_nullable(i\_partner\_purchase\_order\_id), meal\_time\_cd = quote\_nullable(i\_meal\_time\_cd), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), etl\_job\_sno = quote\_nullable(i\_etl\_job\_sno), transaction\_date = quote\_nullable(i\_transaction\_date), order\_status\_desc = quote\_nullable(i\_order\_status\_desc) WHERE <custom\_conditions>; |
| create\_purchase\_order | INSERT INTO masters.purchase\_order (t.purchase\_order\_sno, t.qbox\_entity\_sno, t.restaurant\_sno, t.delivery\_partner\_sno, t.order\_status\_cd, t.ordered\_time, t.ordered\_by, t.partner\_purchase\_order\_id, t.meal\_time\_cd, t.description, t.active\_flag, t.etl\_job\_sno, t.transaction\_date, t.order\_status\_desc) VALUES (i\_purchase\_order\_sno, i\_qbox\_entity\_sno, i\_restaurant\_sno, i\_delivery\_partner\_sno, i\_order\_status\_cd, i\_ordered\_time, i\_ordered\_by, i\_partner\_purchase\_order\_id, i\_meal\_time\_cd, i\_description, i\_active\_flag, i\_etl\_job\_sno, i\_transaction\_date, i\_order\_status\_desc); |
| get\_all\_purchase\_orders | SELECT t.purchase\_order\_sno, t.qbox\_entity\_sno, t.restaurant\_sno, t.delivery\_partner\_sno, t.order\_status\_cd, t.ordered\_time, t.ordered\_by, t.partner\_purchase\_order\_id, t.meal\_time\_cd, t.description, t.active\_flag, t.etl\_job\_sno, t.transaction\_date, t.order\_status\_desc, f\_delivery\_partner.name, f\_qbox\_entity.name, f\_restaurant.name FROM masters.purchase\_order t LEFT JOIN masters.delivery\_partner f\_delivery\_partner ON t.delivery\_partner\_sno = f\_delivery\_partner.delivery\_partner\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.restaurant f\_restaurant ON t.restaurant\_sno = f\_restaurant.restaurant\_sno ; |
| custom\_search\_purchase\_order\_by\_id | SELECT t.purchase\_order\_sno, t.qbox\_entity\_sno, t.restaurant\_sno, t.delivery\_partner\_sno, t.order\_status\_cd, t.ordered\_time, t.ordered\_by, t.partner\_purchase\_order\_id, t.meal\_time\_cd, t.description, t.active\_flag, t.etl\_job\_sno, t.transaction\_date, t.order\_status\_desc, f\_delivery\_partner.name, f\_qbox\_entity.name, f\_restaurant.name FROM masters.purchase\_order t LEFT JOIN masters.delivery\_partner f\_delivery\_partner ON t.delivery\_partner\_sno = f\_delivery\_partner.delivery\_partner\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno LEFT JOIN masters.restaurant f\_restaurant ON t.restaurant\_sno = f\_restaurant.restaurant\_sno WHERE <custom\_search\_conditions>; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Address**

### **Database Table Name: address**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| address\_sno | bigint - (Numeric Precision: 64) | address\_pkey (PRIMARY KEY), |
| line1 | text | null |
| line2 | text | null |
| area\_sno | integer | null |
| city\_sno | integer | null |
| geo\_loc\_code | text | null |
| description | text | null |
| active\_flag | boolean | null |
| state\_sno | integer | null |
| country\_sno | integer | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Address Sno | hidden | number | true | Manual Entry |
| Line1 | textbox | string | true | Manual Entry |
| Line2 | textbox | string | false | Manual Entry |
| Area Sno | textbox | number | false | Manual Entry |
| City Sno | textbox | number | true | Manual Entry |
| Geo Loc Code | textbox | string | true | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |
| State Sno | textbox | number | false | Manual Entry |
| Country Sno | textbox | number | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_addresss | get | /api/v1/get\_addresss |
| create\_address | post | /api/v1/create\_address |
| update\_address | put | /api/v1/update\_address |
| delete\_address | delete | /api/v1/delete\_address |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_addresss | {} | {"data":{"address\_sno":"bigint","area\_sno":"integer","state\_sno":"integer","geo\_loc\_code":"text","country\_sno":"integer","description":"text","city\_sno":"integer","active\_flag":"boolean","line2":"text","line1":"text"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_address | {"data":{"address\_sno":"bigint","area\_sno":"integer","state\_sno":"integer","geo\_loc\_code":"text","country\_sno":"integer","description":"text","city\_sno":"integer","active\_flag":"boolean","line2":"text","line1":"text"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_address | {"data":{"address\_sno":"bigint","area\_sno":"integer","state\_sno":"integer","geo\_loc\_code":"text","country\_sno":"integer","description":"text","city\_sno":"integer","active\_flag":"boolean","line2":"text","line1":"text"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_address | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| custom\_update\_address | UPDATE masters.address SET address\_sno = quote\_nullable(i\_address\_sno), line1 = quote\_nullable(i\_line1), line2 = quote\_nullable(i\_line2), area\_sno = quote\_nullable(i\_area\_sno), city\_sno = quote\_nullable(i\_city\_sno), geo\_loc\_code = quote\_nullable(i\_geo\_loc\_code), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), state\_sno = quote\_nullable(i\_state\_sno), country\_sno = quote\_nullable(i\_country\_sno) WHERE <custom\_conditions>; |
| custom\_delete\_address\_by\_id | DELETE FROM masters.address WHERE <custom\_conditions>; |
| get\_all\_addresss | SELECT t.address\_sno, t.line1, t.line2, t.area\_sno, t.city\_sno, t.geo\_loc\_code, t.description, t.active\_flag, t.state\_sno, t.country\_sno FROM masters.address t ; |
| update\_address | UPDATE masters.address SET address\_sno = quote\_nullable(i\_address\_sno), line1 = quote\_nullable(i\_line1), line2 = quote\_nullable(i\_line2), area\_sno = quote\_nullable(i\_area\_sno), city\_sno = quote\_nullable(i\_city\_sno), geo\_loc\_code = quote\_nullable(i\_geo\_loc\_code), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), state\_sno = quote\_nullable(i\_state\_sno), country\_sno = quote\_nullable(i\_country\_sno) WHERE address\_sno = quote\_nullable(i\_address\_sno); |
| custom\_search\_address\_by\_id | SELECT t.address\_sno, t.line1, t.line2, t.area\_sno, t.city\_sno, t.geo\_loc\_code, t.description, t.active\_flag, t.state\_sno, t.country\_sno FROM masters.address t WHERE <custom\_search\_conditions>; |
| unique\_address\_exists | SELECT 1 FROM masters.address WHERE <unique\_conditions> LIMIT 1; |
| create\_address | INSERT INTO masters.address (t.address\_sno, t.line1, t.line2, t.area\_sno, t.city\_sno, t.geo\_loc\_code, t.description, t.active\_flag, t.state\_sno, t.country\_sno) VALUES (i\_address\_sno, i\_line1, i\_line2, i\_area\_sno, i\_city\_sno, i\_geo\_loc\_code, i\_description, i\_active\_flag, i\_state\_sno, i\_country\_sno); |
| delete\_address\_by\_id | DELETE FROM masters.address WHERE address\_sno = quote\_nullable(i\_address\_sno); |
| get\_address\_by\_id | SELECT t.address\_sno, t.line1, t.line2, t.area\_sno, t.city\_sno, t.geo\_loc\_code, t.description, t.active\_flag, t.state\_sno, t.country\_sno FROM masters.address t WHERE t.address\_sno = quote\_nullable(i\_address\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Qbox Entity Delivery Partner**

### **Database Table Name: qbox\_entity\_delivery\_partner**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| qbox\_entity\_delivery\_partner\_sno | bigint - (Numeric Precision: 64) | qbox\_entity\_delivery\_partner\_pkey (PRIMARY KEY), |
| delivery\_type\_cd | smallint - (Numeric Precision: 16) | qbox\_entity\_delivery\_partner\_delivery\_type\_cd\_fkey (FOREIGN KEY), |
| qbox\_entity\_sno | bigint - (Numeric Precision: 64) | qbox\_entity\_delivery\_partner\_qbox\_entity\_sno\_fkey (FOREIGN KEY), |
| delivery\_partner\_sno | smallint - (Numeric Precision: 16) | qbox\_entity\_delivery\_partner\_delivery\_partner\_sno\_fkey (FOREIGN KEY), |
| description | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Qbox Entity Delivery Partner Sno | hidden | number | true | Manual Entry |
| Delivery Type Cd | listbox | number | true | API feed list box selection |
| Qbox Entity Sno | listbox | number | true | API feed list box selection |
| Delivery Partner Sno | listbox | number | true | API feed list box selection |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_qbox\_entity\_delivery\_partners | get | /api/v1/get\_qbox\_entity\_delivery\_partners |
| create\_qbox\_entity\_delivery\_partner | post | /api/v1/create\_qbox\_entity\_delivery\_partner |
| update\_qbox\_entity\_delivery\_partner | put | /api/v1/update\_qbox\_entity\_delivery\_partner |
| delete\_qbox\_entity\_delivery\_partner | delete | /api/v1/delete\_qbox\_entity\_delivery\_partner |
| get\_delivery\_partners | get | /api/v1/get\_delivery\_partners |
| get\_codes\_dtls | get | /api/v1/get\_codes\_dtls |
| get\_qbox\_entitys | get | /api/v1/get\_qbox\_entitys |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_qbox\_entity\_delivery\_partners | {} | {"data":{"delivery\_type\_cd":"smallint","delivery\_partner\_sno":"smallint","description":"text","qbox\_entity\_sno":"bigint","active\_flag":"boolean","qbox\_entity\_delivery\_partner\_sno":"bigint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_qbox\_entity\_delivery\_partner | {"data":{"delivery\_type\_cd":"smallint","delivery\_partner\_sno":"smallint","description":"text","qbox\_entity\_sno":"bigint","active\_flag":"boolean","qbox\_entity\_delivery\_partner\_sno":"bigint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_qbox\_entity\_delivery\_partner | {"data":{"delivery\_type\_cd":"smallint","delivery\_partner\_sno":"smallint","description":"text","qbox\_entity\_sno":"bigint","active\_flag":"boolean","qbox\_entity\_delivery\_partner\_sno":"bigint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_qbox\_entity\_delivery\_partner | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_delivery\_partners | {"id":"integer"} | {"data":"array of delivery\_partner"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_codes\_dtls | {"id":"integer"} | {"data":"array of codes\_dtl"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_qbox\_entitys | {"id":"integer"} | {"data":"array of qbox\_entity"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| create\_qbox\_entity\_delivery\_partner | INSERT INTO masters.qbox\_entity\_delivery\_partner (t.qbox\_entity\_delivery\_partner\_sno, t.delivery\_type\_cd, t.qbox\_entity\_sno, t.delivery\_partner\_sno, t.description, t.active\_flag) VALUES (i\_qbox\_entity\_delivery\_partner\_sno, i\_delivery\_type\_cd, i\_qbox\_entity\_sno, i\_delivery\_partner\_sno, i\_description, i\_active\_flag); |
| update\_qbox\_entity\_delivery\_partner | UPDATE masters.qbox\_entity\_delivery\_partner SET qbox\_entity\_delivery\_partner\_sno = quote\_nullable(i\_qbox\_entity\_delivery\_partner\_sno), delivery\_type\_cd = quote\_nullable(i\_delivery\_type\_cd), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE qbox\_entity\_delivery\_partner\_sno = quote\_nullable(i\_qbox\_entity\_delivery\_partner\_sno); |
| custom\_search\_qbox\_entity\_delivery\_partner\_by\_id | SELECT t.qbox\_entity\_delivery\_partner\_sno, t.delivery\_type\_cd, t.qbox\_entity\_sno, t.delivery\_partner\_sno, t.description, t.active\_flag, f\_delivery\_partner.name, f\_qbox\_entity.name FROM masters.qbox\_entity\_delivery\_partner t LEFT JOIN masters.delivery\_partner f\_delivery\_partner ON t.delivery\_partner\_sno = f\_delivery\_partner.delivery\_partner\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno WHERE <custom\_search\_conditions>; |
| get\_qbox\_entity\_delivery\_partner\_by\_id | SELECT t.qbox\_entity\_delivery\_partner\_sno, t.delivery\_type\_cd, t.qbox\_entity\_sno, t.delivery\_partner\_sno, t.description, t.active\_flag, f\_delivery\_partner.name, f\_qbox\_entity.name FROM masters.qbox\_entity\_delivery\_partner t LEFT JOIN masters.delivery\_partner f\_delivery\_partner ON t.delivery\_partner\_sno = f\_delivery\_partner.delivery\_partner\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno WHERE t.qbox\_entity\_delivery\_partner\_sno = quote\_nullable(i\_qbox\_entity\_delivery\_partner\_sno); |
| delete\_qbox\_entity\_delivery\_partner\_by\_id | DELETE FROM masters.qbox\_entity\_delivery\_partner WHERE qbox\_entity\_delivery\_partner\_sno = quote\_nullable(i\_qbox\_entity\_delivery\_partner\_sno); |
| unique\_qbox\_entity\_delivery\_partner\_exists | SELECT 1 FROM masters.qbox\_entity\_delivery\_partner WHERE <unique\_conditions> LIMIT 1; |
| custom\_update\_qbox\_entity\_delivery\_partner | UPDATE masters.qbox\_entity\_delivery\_partner SET qbox\_entity\_delivery\_partner\_sno = quote\_nullable(i\_qbox\_entity\_delivery\_partner\_sno), delivery\_type\_cd = quote\_nullable(i\_delivery\_type\_cd), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| custom\_delete\_qbox\_entity\_delivery\_partner\_by\_id | DELETE FROM masters.qbox\_entity\_delivery\_partner WHERE <custom\_conditions>; |
| get\_all\_qbox\_entity\_delivery\_partners | SELECT t.qbox\_entity\_delivery\_partner\_sno, t.delivery\_type\_cd, t.qbox\_entity\_sno, t.delivery\_partner\_sno, t.description, t.active\_flag, f\_delivery\_partner.name, f\_qbox\_entity.name FROM masters.qbox\_entity\_delivery\_partner t LEFT JOIN masters.delivery\_partner f\_delivery\_partner ON t.delivery\_partner\_sno = f\_delivery\_partner.delivery\_partner\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno ; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Role Permission**

### **Database Table Name: role\_permission**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| role\_permission\_sno | bigint - (Numeric Precision: 64) | role\_function\_pkey (PRIMARY KEY), |
| app\_role\_sno | smallint - (Numeric Precision: 16) | role\_function\_app\_role\_sno\_fkey (FOREIGN KEY), |
| app\_permission\_sno | integer | role\_function\_app\_function\_sno\_fkey (FOREIGN KEY), |
| description | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Role Permission Sno | hidden | number | true | Manual Entry |
| App Role Sno | listbox | number | true | API feed list box selection |
| App Permission Sno | listbox | number | true | API feed list box selection |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_role\_permissions | get | /api/v1/get\_role\_permissions |
| create\_role\_permission | post | /api/v1/create\_role\_permission |
| update\_role\_permission | put | /api/v1/update\_role\_permission |
| delete\_role\_permission | delete | /api/v1/delete\_role\_permission |
| get\_app\_permissions | get | /api/v1/get\_app\_permissions |
| get\_app\_roles | get | /api/v1/get\_app\_roles |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_role\_permissions | {} | {"data":{"app\_permission\_sno":"integer","description":"text","active\_flag":"boolean","app\_role\_sno":"smallint","role\_permission\_sno":"bigint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_role\_permission | {"data":{"app\_permission\_sno":"integer","description":"text","active\_flag":"boolean","app\_role\_sno":"smallint","role\_permission\_sno":"bigint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_role\_permission | {"data":{"app\_permission\_sno":"integer","description":"text","active\_flag":"boolean","app\_role\_sno":"smallint","role\_permission\_sno":"bigint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_role\_permission | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_app\_permissions | {"id":"integer"} | {"data":"array of app\_permission"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_app\_roles | {"id":"integer"} | {"data":"array of app\_role"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| delete\_role\_permission\_by\_id | DELETE FROM masters.role\_permission WHERE role\_permission\_sno = quote\_nullable(i\_role\_permission\_sno); |
| update\_role\_permission | UPDATE masters.role\_permission SET role\_permission\_sno = quote\_nullable(i\_role\_permission\_sno), app\_role\_sno = quote\_nullable(i\_app\_role\_sno), app\_permission\_sno = quote\_nullable(i\_app\_permission\_sno), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE role\_permission\_sno = quote\_nullable(i\_role\_permission\_sno); |
| custom\_update\_role\_permission | UPDATE masters.role\_permission SET role\_permission\_sno = quote\_nullable(i\_role\_permission\_sno), app\_role\_sno = quote\_nullable(i\_app\_role\_sno), app\_permission\_sno = quote\_nullable(i\_app\_permission\_sno), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| unique\_role\_permission\_exists | SELECT 1 FROM masters.role\_permission WHERE <unique\_conditions> LIMIT 1; |
| custom\_delete\_role\_permission\_by\_id | DELETE FROM masters.role\_permission WHERE <custom\_conditions>; |
| get\_role\_permission\_by\_id | SELECT t.role\_permission\_sno, t.app\_role\_sno, t.app\_permission\_sno, t.description, t.active\_flag, f\_app\_permission.name, f\_app\_role.name FROM masters.role\_permission t LEFT JOIN masters.app\_permission f\_app\_permission ON t.app\_permission\_sno = f\_app\_permission.app\_permission\_sno LEFT JOIN masters.app\_role f\_app\_role ON t.app\_role\_sno = f\_app\_role.app\_role\_sno WHERE t.role\_permission\_sno = quote\_nullable(i\_role\_permission\_sno); |
| create\_role\_permission | INSERT INTO masters.role\_permission (t.role\_permission\_sno, t.app\_role\_sno, t.app\_permission\_sno, t.description, t.active\_flag) VALUES (i\_role\_permission\_sno, i\_app\_role\_sno, i\_app\_permission\_sno, i\_description, i\_active\_flag); |
| custom\_search\_role\_permission\_by\_id | SELECT t.role\_permission\_sno, t.app\_role\_sno, t.app\_permission\_sno, t.description, t.active\_flag, f\_app\_permission.name, f\_app\_role.name FROM masters.role\_permission t LEFT JOIN masters.app\_permission f\_app\_permission ON t.app\_permission\_sno = f\_app\_permission.app\_permission\_sno LEFT JOIN masters.app\_role f\_app\_role ON t.app\_role\_sno = f\_app\_role.app\_role\_sno WHERE <custom\_search\_conditions>; |
| get\_all\_role\_permissions | SELECT t.role\_permission\_sno, t.app\_role\_sno, t.app\_permission\_sno, t.description, t.active\_flag, f\_app\_permission.name, f\_app\_role.name FROM masters.role\_permission t LEFT JOIN masters.app\_permission f\_app\_permission ON t.app\_permission\_sno = f\_app\_permission.app\_permission\_sno LEFT JOIN masters.app\_role f\_app\_role ON t.app\_role\_sno = f\_app\_role.app\_role\_sno ; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Role Menu**

### **Database Table Name: role\_menu**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| role\_menu\_sno | smallint - (Numeric Precision: 16) | role\_menu\_pkey (PRIMARY KEY), |
| app\_menu\_sno | smallint - (Numeric Precision: 16) | null |
| app\_role\_sno | smallint - (Numeric Precision: 16) | role\_menu\_app\_role\_sno\_fkey (FOREIGN KEY), |
| status | boolean | null |
| description | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Role Menu Sno | hidden | number | true | Manual Entry |
| App Menu Sno | textbox\_number | number | true | Manual Entry |
| App Role Sno | listbox | number | true | API feed list box selection |
| Status | checkbox | boolean | false | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_role\_menus | get | /api/v1/get\_role\_menus |
| create\_role\_menu | post | /api/v1/create\_role\_menu |
| update\_role\_menu | put | /api/v1/update\_role\_menu |
| delete\_role\_menu | delete | /api/v1/delete\_role\_menu |
| get\_app\_roles | get | /api/v1/get\_app\_roles |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_role\_menus | {} | {"data":{"description":"text","active\_flag":"boolean","app\_role\_sno":"smallint","app\_menu\_sno":"smallint","status":"boolean","role\_menu\_sno":"smallint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_role\_menu | {"data":{"description":"text","active\_flag":"boolean","app\_role\_sno":"smallint","app\_menu\_sno":"smallint","status":"boolean","role\_menu\_sno":"smallint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_role\_menu | {"data":{"description":"text","active\_flag":"boolean","app\_role\_sno":"smallint","app\_menu\_sno":"smallint","status":"boolean","role\_menu\_sno":"smallint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_role\_menu | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_app\_roles | {"id":"integer"} | {"data":"array of app\_role"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| delete\_role\_menu\_by\_id | DELETE FROM masters.role\_menu WHERE role\_menu\_sno = quote\_nullable(i\_role\_menu\_sno); |
| create\_role\_menu | INSERT INTO masters.role\_menu (t.role\_menu\_sno, t.app\_menu\_sno, t.app\_role\_sno, t.status, t.description, t.active\_flag) VALUES (i\_role\_menu\_sno, i\_app\_menu\_sno, i\_app\_role\_sno, i\_status, i\_description, i\_active\_flag); |
| custom\_search\_role\_menu\_by\_id | SELECT t.role\_menu\_sno, t.app\_menu\_sno, t.app\_role\_sno, t.status, t.description, t.active\_flag, f\_app\_role.name FROM masters.role\_menu t LEFT JOIN masters.app\_role f\_app\_role ON t.app\_role\_sno = f\_app\_role.app\_role\_sno WHERE <custom\_search\_conditions>; |
| custom\_delete\_role\_menu\_by\_id | DELETE FROM masters.role\_menu WHERE <custom\_conditions>; |
| get\_role\_menu\_by\_id | SELECT t.role\_menu\_sno, t.app\_menu\_sno, t.app\_role\_sno, t.status, t.description, t.active\_flag, f\_app\_role.name FROM masters.role\_menu t LEFT JOIN masters.app\_role f\_app\_role ON t.app\_role\_sno = f\_app\_role.app\_role\_sno WHERE t.role\_menu\_sno = quote\_nullable(i\_role\_menu\_sno); |
| unique\_role\_menu\_exists | SELECT 1 FROM masters.role\_menu WHERE <unique\_conditions> LIMIT 1; |
| update\_role\_menu | UPDATE masters.role\_menu SET role\_menu\_sno = quote\_nullable(i\_role\_menu\_sno), app\_menu\_sno = quote\_nullable(i\_app\_menu\_sno), app\_role\_sno = quote\_nullable(i\_app\_role\_sno), status = quote\_nullable(i\_status), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE role\_menu\_sno = quote\_nullable(i\_role\_menu\_sno); |
| get\_all\_role\_menus | SELECT t.role\_menu\_sno, t.app\_menu\_sno, t.app\_role\_sno, t.status, t.description, t.active\_flag, f\_app\_role.name FROM masters.role\_menu t LEFT JOIN masters.app\_role f\_app\_role ON t.app\_role\_sno = f\_app\_role.app\_role\_sno ; |
| custom\_update\_role\_menu | UPDATE masters.role\_menu SET role\_menu\_sno = quote\_nullable(i\_role\_menu\_sno), app\_menu\_sno = quote\_nullable(i\_app\_menu\_sno), app\_role\_sno = quote\_nullable(i\_app\_role\_sno), status = quote\_nullable(i\_status), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Sales Order**

### **Database Table Name: sales\_order**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| sales\_order\_sno | bigint - (Numeric Precision: 64) | sales\_order\_pkey (PRIMARY KEY), |
| partner\_sales\_order\_id | text | null |
| qbox\_entity\_sno | bigint - (Numeric Precision: 64) | sales\_order\_qbox\_entity\_sno\_fkey (FOREIGN KEY), |
| delivery\_partner\_sno | bigint - (Numeric Precision: 64) | sales\_order\_delivery\_partner\_sno\_fkey (FOREIGN KEY), |
| sales\_order\_status\_cd | smallint - (Numeric Precision: 16) | null |
| ordered\_time | timestamp without time zone | null |
| ordered\_by | bigint - (Numeric Precision: 64) | null |
| partner\_customer\_ref | text | null |
| description | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Sales Order Sno | hidden | number | true | Manual Entry |
| Partner Sales Order Id | textbox | string | true | Manual Entry |
| Qbox Entity Sno | listbox | number | true | API feed list box selection |
| Delivery Partner Sno | listbox | number | true | API feed list box selection |
| Sales Order Status Cd | textbox\_number | number | true | Manual Entry |
| Ordered Time | textbox | string | false | Manual Entry |
| Ordered By | textbox\_number | number | true | Manual Entry |
| Partner Customer Ref | textbox | string | false | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_sales\_orders | get | /api/v1/get\_sales\_orders |
| create\_sales\_order | post | /api/v1/create\_sales\_order |
| update\_sales\_order | put | /api/v1/update\_sales\_order |
| delete\_sales\_order | delete | /api/v1/delete\_sales\_order |
| get\_delivery\_partners | get | /api/v1/get\_delivery\_partners |
| get\_qbox\_entitys | get | /api/v1/get\_qbox\_entitys |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_sales\_orders | {} | {"data":{"ordered\_time":"timestamp without time zone","sales\_order\_sno":"bigint","delivery\_partner\_sno":"bigint","partner\_customer\_ref":"text","description":"text","qbox\_entity\_sno":"bigint","ordered\_by":"bigint","active\_flag":"boolean","sales\_order\_status\_cd":"smallint","partner\_sales\_order\_id":"text"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_sales\_order | {"data":{"ordered\_time":"timestamp without time zone","sales\_order\_sno":"bigint","delivery\_partner\_sno":"bigint","partner\_customer\_ref":"text","description":"text","qbox\_entity\_sno":"bigint","ordered\_by":"bigint","active\_flag":"boolean","sales\_order\_status\_cd":"smallint","partner\_sales\_order\_id":"text"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_sales\_order | {"data":{"ordered\_time":"timestamp without time zone","sales\_order\_sno":"bigint","delivery\_partner\_sno":"bigint","partner\_customer\_ref":"text","description":"text","qbox\_entity\_sno":"bigint","ordered\_by":"bigint","active\_flag":"boolean","sales\_order\_status\_cd":"smallint","partner\_sales\_order\_id":"text"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_sales\_order | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_delivery\_partners | {"id":"integer"} | {"data":"array of delivery\_partner"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_qbox\_entitys | {"id":"integer"} | {"data":"array of qbox\_entity"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| custom\_update\_sales\_order | UPDATE masters.sales\_order SET sales\_order\_sno = quote\_nullable(i\_sales\_order\_sno), partner\_sales\_order\_id = quote\_nullable(i\_partner\_sales\_order\_id), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno), sales\_order\_status\_cd = quote\_nullable(i\_sales\_order\_status\_cd), ordered\_time = quote\_nullable(i\_ordered\_time), ordered\_by = quote\_nullable(i\_ordered\_by), partner\_customer\_ref = quote\_nullable(i\_partner\_customer\_ref), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| create\_sales\_order | INSERT INTO masters.sales\_order (t.sales\_order\_sno, t.partner\_sales\_order\_id, t.qbox\_entity\_sno, t.delivery\_partner\_sno, t.sales\_order\_status\_cd, t.ordered\_time, t.ordered\_by, t.partner\_customer\_ref, t.description, t.active\_flag) VALUES (i\_sales\_order\_sno, i\_partner\_sales\_order\_id, i\_qbox\_entity\_sno, i\_delivery\_partner\_sno, i\_sales\_order\_status\_cd, i\_ordered\_time, i\_ordered\_by, i\_partner\_customer\_ref, i\_description, i\_active\_flag); |
| delete\_sales\_order\_by\_id | DELETE FROM masters.sales\_order WHERE sales\_order\_sno = quote\_nullable(i\_sales\_order\_sno); |
| custom\_search\_sales\_order\_by\_id | SELECT t.sales\_order\_sno, t.partner\_sales\_order\_id, t.qbox\_entity\_sno, t.delivery\_partner\_sno, t.sales\_order\_status\_cd, t.ordered\_time, t.ordered\_by, t.partner\_customer\_ref, t.description, t.active\_flag, f\_delivery\_partner.name, f\_qbox\_entity.name FROM masters.sales\_order t LEFT JOIN masters.delivery\_partner f\_delivery\_partner ON t.delivery\_partner\_sno = f\_delivery\_partner.delivery\_partner\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno WHERE <custom\_search\_conditions>; |
| get\_all\_sales\_orders | SELECT t.sales\_order\_sno, t.partner\_sales\_order\_id, t.qbox\_entity\_sno, t.delivery\_partner\_sno, t.sales\_order\_status\_cd, t.ordered\_time, t.ordered\_by, t.partner\_customer\_ref, t.description, t.active\_flag, f\_delivery\_partner.name, f\_qbox\_entity.name FROM masters.sales\_order t LEFT JOIN masters.delivery\_partner f\_delivery\_partner ON t.delivery\_partner\_sno = f\_delivery\_partner.delivery\_partner\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno ; |
| unique\_sales\_order\_exists | SELECT 1 FROM masters.sales\_order WHERE <unique\_conditions> LIMIT 1; |
| custom\_delete\_sales\_order\_by\_id | DELETE FROM masters.sales\_order WHERE <custom\_conditions>; |
| update\_sales\_order | UPDATE masters.sales\_order SET sales\_order\_sno = quote\_nullable(i\_sales\_order\_sno), partner\_sales\_order\_id = quote\_nullable(i\_partner\_sales\_order\_id), qbox\_entity\_sno = quote\_nullable(i\_qbox\_entity\_sno), delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno), sales\_order\_status\_cd = quote\_nullable(i\_sales\_order\_status\_cd), ordered\_time = quote\_nullable(i\_ordered\_time), ordered\_by = quote\_nullable(i\_ordered\_by), partner\_customer\_ref = quote\_nullable(i\_partner\_customer\_ref), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE sales\_order\_sno = quote\_nullable(i\_sales\_order\_sno); |
| get\_sales\_order\_by\_id | SELECT t.sales\_order\_sno, t.partner\_sales\_order\_id, t.qbox\_entity\_sno, t.delivery\_partner\_sno, t.sales\_order\_status\_cd, t.ordered\_time, t.ordered\_by, t.partner\_customer\_ref, t.description, t.active\_flag, f\_delivery\_partner.name, f\_qbox\_entity.name FROM masters.sales\_order t LEFT JOIN masters.delivery\_partner f\_delivery\_partner ON t.delivery\_partner\_sno = f\_delivery\_partner.delivery\_partner\_sno LEFT JOIN masters.qbox\_entity f\_qbox\_entity ON t.qbox\_entity\_sno = f\_qbox\_entity.qbox\_entity\_sno WHERE t.sales\_order\_sno = quote\_nullable(i\_sales\_order\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Sales Order Dtl**

### **Database Table Name: sales\_order\_dtl**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| sales\_order\_dtl\_sno | bigint - (Numeric Precision: 64) | sales\_order\_dtl\_pkey (PRIMARY KEY), |
| sales\_order\_sno | bigint - (Numeric Precision: 64) | sales\_order\_dtl\_sales\_order\_sno\_fkey (FOREIGN KEY), |
| partner\_food\_code | text | null |
| restaurant\_food\_sku\_sno | bigint - (Numeric Precision: 64) | sales\_order\_dtl\_restaurant\_food\_sku\_sno\_fkey (FOREIGN KEY), |
| order\_quantity | integer | null |
| sku\_price | numeric | null |
| delivered\_quantity | smallint - (Numeric Precision: 16) | null |
| description | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Sales Order Dtl Sno | hidden | number | true | Manual Entry |
| Sales Order Sno | listbox | number | true | API feed list box selection |
| Partner Food Code | textbox | string | false | Manual Entry |
| Restaurant Food Sku Sno | listbox | number | true | API feed list box selection |
| Order Quantity | textbox | number | true | Manual Entry |
| Sku Price | textbox | number | false | Manual Entry |
| Delivered Quantity | textbox\_number | number | false | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_sales\_order\_dtls | get | /api/v1/get\_sales\_order\_dtls |
| create\_sales\_order\_dtl | post | /api/v1/create\_sales\_order\_dtl |
| update\_sales\_order\_dtl | put | /api/v1/update\_sales\_order\_dtl |
| delete\_sales\_order\_dtl | delete | /api/v1/delete\_sales\_order\_dtl |
| get\_restaurant\_food\_skus | get | /api/v1/get\_restaurant\_food\_skus |
| get\_sales\_orders | get | /api/v1/get\_sales\_orders |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_sales\_order\_dtls | {} | {"data":{"sales\_order\_sno":"bigint","sales\_order\_dtl\_sno":"bigint","restaurant\_food\_sku\_sno":"bigint","sku\_price":"numeric","order\_quantity":"integer","description":"text","partner\_food\_code":"text","active\_flag":"boolean","delivered\_quantity":"smallint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_sales\_order\_dtl | {"data":{"sales\_order\_sno":"bigint","sales\_order\_dtl\_sno":"bigint","restaurant\_food\_sku\_sno":"bigint","sku\_price":"numeric","order\_quantity":"integer","description":"text","partner\_food\_code":"text","active\_flag":"boolean","delivered\_quantity":"smallint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_sales\_order\_dtl | {"data":{"sales\_order\_sno":"bigint","sales\_order\_dtl\_sno":"bigint","restaurant\_food\_sku\_sno":"bigint","sku\_price":"numeric","order\_quantity":"integer","description":"text","partner\_food\_code":"text","active\_flag":"boolean","delivered\_quantity":"smallint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_sales\_order\_dtl | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_restaurant\_food\_skus | {"id":"integer"} | {"data":"array of restaurant\_food\_sku"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_sales\_orders | {"id":"integer"} | {"data":"array of sales\_order"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| update\_sales\_order\_dtl | UPDATE masters.sales\_order\_dtl SET sales\_order\_dtl\_sno = quote\_nullable(i\_sales\_order\_dtl\_sno), sales\_order\_sno = quote\_nullable(i\_sales\_order\_sno), partner\_food\_code = quote\_nullable(i\_partner\_food\_code), restaurant\_food\_sku\_sno = quote\_nullable(i\_restaurant\_food\_sku\_sno), order\_quantity = quote\_nullable(i\_order\_quantity), sku\_price = quote\_nullable(i\_sku\_price), delivered\_quantity = quote\_nullable(i\_delivered\_quantity), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE sales\_order\_dtl\_sno = quote\_nullable(i\_sales\_order\_dtl\_sno); |
| get\_sales\_order\_dtl\_by\_id | SELECT t.sales\_order\_dtl\_sno, t.sales\_order\_sno, t.partner\_food\_code, t.restaurant\_food\_sku\_sno, t.order\_quantity, t.sku\_price, t.delivered\_quantity, t.description, t.active\_flag, f\_restaurant\_food\_sku.restaurant\_sku\_name FROM masters.sales\_order\_dtl t LEFT JOIN masters.restaurant\_food\_sku f\_restaurant\_food\_sku ON t.restaurant\_food\_sku\_sno = f\_restaurant\_food\_sku.restaurant\_food\_sku\_sno WHERE t.sales\_order\_dtl\_sno = quote\_nullable(i\_sales\_order\_dtl\_sno); |
| custom\_update\_sales\_order\_dtl | UPDATE masters.sales\_order\_dtl SET sales\_order\_dtl\_sno = quote\_nullable(i\_sales\_order\_dtl\_sno), sales\_order\_sno = quote\_nullable(i\_sales\_order\_sno), partner\_food\_code = quote\_nullable(i\_partner\_food\_code), restaurant\_food\_sku\_sno = quote\_nullable(i\_restaurant\_food\_sku\_sno), order\_quantity = quote\_nullable(i\_order\_quantity), sku\_price = quote\_nullable(i\_sku\_price), delivered\_quantity = quote\_nullable(i\_delivered\_quantity), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| delete\_sales\_order\_dtl\_by\_id | DELETE FROM masters.sales\_order\_dtl WHERE sales\_order\_dtl\_sno = quote\_nullable(i\_sales\_order\_dtl\_sno); |
| create\_sales\_order\_dtl | INSERT INTO masters.sales\_order\_dtl (t.sales\_order\_dtl\_sno, t.sales\_order\_sno, t.partner\_food\_code, t.restaurant\_food\_sku\_sno, t.order\_quantity, t.sku\_price, t.delivered\_quantity, t.description, t.active\_flag) VALUES (i\_sales\_order\_dtl\_sno, i\_sales\_order\_sno, i\_partner\_food\_code, i\_restaurant\_food\_sku\_sno, i\_order\_quantity, i\_sku\_price, i\_delivered\_quantity, i\_description, i\_active\_flag); |
| unique\_sales\_order\_dtl\_exists | SELECT 1 FROM masters.sales\_order\_dtl WHERE <unique\_conditions> LIMIT 1; |
| custom\_search\_sales\_order\_dtl\_by\_id | SELECT t.sales\_order\_dtl\_sno, t.sales\_order\_sno, t.partner\_food\_code, t.restaurant\_food\_sku\_sno, t.order\_quantity, t.sku\_price, t.delivered\_quantity, t.description, t.active\_flag, f\_restaurant\_food\_sku.restaurant\_sku\_name FROM masters.sales\_order\_dtl t LEFT JOIN masters.restaurant\_food\_sku f\_restaurant\_food\_sku ON t.restaurant\_food\_sku\_sno = f\_restaurant\_food\_sku.restaurant\_food\_sku\_sno WHERE <custom\_search\_conditions>; |
| get\_all\_sales\_order\_dtls | SELECT t.sales\_order\_dtl\_sno, t.sales\_order\_sno, t.partner\_food\_code, t.restaurant\_food\_sku\_sno, t.order\_quantity, t.sku\_price, t.delivered\_quantity, t.description, t.active\_flag, f\_restaurant\_food\_sku.restaurant\_sku\_name FROM masters.sales\_order\_dtl t LEFT JOIN masters.restaurant\_food\_sku f\_restaurant\_food\_sku ON t.restaurant\_food\_sku\_sno = f\_restaurant\_food\_sku.restaurant\_food\_sku\_sno ; |
| custom\_delete\_sales\_order\_dtl\_by\_id | DELETE FROM masters.sales\_order\_dtl WHERE <custom\_conditions>; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Signin Info**

### **Database Table Name: signin\_info**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| signin\_info\_sno | bigint - (Numeric Precision: 64) | signin\_info\_pkey (PRIMARY KEY), |
| app\_user\_sno | bigint - (Numeric Precision: 64) | signin\_info\_app\_user\_sno\_fkey (FOREIGN KEY), |
| push\_token | text | null |
| device\_type\_cd | smallint - (Numeric Precision: 16) | signin\_info\_device\_type\_cd\_fkey (FOREIGN KEY), |
| device\_id | text | null |
| login\_on | timestamp without time zone | null |
| logout\_on | timestamp without time zone | null |
| description | text | null |
| active\_flag | boolean | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Signin Info Sno | hidden | number | true | Manual Entry |
| App User Sno | listbox | number | false | API feed list box selection |
| Push Token | textbox | string | false | Manual Entry |
| Device Type Cd | listbox | number | false | API feed list box selection |
| Device Id | textbox | string | false | Manual Entry |
| Login On | textbox | string | false | Manual Entry |
| Logout On | textbox | string | false | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_signin\_infos | get | /api/v1/get\_signin\_infos |
| create\_signin\_info | post | /api/v1/create\_signin\_info |
| update\_signin\_info | put | /api/v1/update\_signin\_info |
| delete\_signin\_info | delete | /api/v1/delete\_signin\_info |
| get\_app\_users | get | /api/v1/get\_app\_users |
| get\_codes\_dtls | get | /api/v1/get\_codes\_dtls |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_signin\_infos | {} | {"data":{"device\_id":"text","login\_on":"timestamp without time zone","signin\_info\_sno":"bigint","logout\_on":"timestamp without time zone","description":"text","app\_user\_sno":"bigint","active\_flag":"boolean","device\_type\_cd":"smallint","push\_token":"text"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_signin\_info | {"data":{"device\_id":"text","login\_on":"timestamp without time zone","signin\_info\_sno":"bigint","logout\_on":"timestamp without time zone","description":"text","app\_user\_sno":"bigint","active\_flag":"boolean","device\_type\_cd":"smallint","push\_token":"text"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_signin\_info | {"data":{"device\_id":"text","login\_on":"timestamp without time zone","signin\_info\_sno":"bigint","logout\_on":"timestamp without time zone","description":"text","app\_user\_sno":"bigint","active\_flag":"boolean","device\_type\_cd":"smallint","push\_token":"text"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_signin\_info | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_app\_users | {"id":"integer"} | {"data":"array of app\_user"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_codes\_dtls | {"id":"integer"} | {"data":"array of codes\_dtl"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| get\_all\_signin\_infos | SELECT t.signin\_info\_sno, t.app\_user\_sno, t.push\_token, t.device\_type\_cd, t.device\_id, t.login\_on, t.logout\_on, t.description, t.active\_flag, f\_app\_user.name FROM masters.signin\_info t LEFT JOIN masters.app\_user f\_app\_user ON t.app\_user\_sno = f\_app\_user.app\_user\_sno ; |
| unique\_signin\_info\_exists | SELECT 1 FROM masters.signin\_info WHERE <unique\_conditions> LIMIT 1; |
| custom\_update\_signin\_info | UPDATE masters.signin\_info SET signin\_info\_sno = quote\_nullable(i\_signin\_info\_sno), app\_user\_sno = quote\_nullable(i\_app\_user\_sno), push\_token = quote\_nullable(i\_push\_token), device\_type\_cd = quote\_nullable(i\_device\_type\_cd), device\_id = quote\_nullable(i\_device\_id), login\_on = quote\_nullable(i\_login\_on), logout\_on = quote\_nullable(i\_logout\_on), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE <custom\_conditions>; |
| custom\_delete\_signin\_info\_by\_id | DELETE FROM masters.signin\_info WHERE <custom\_conditions>; |
| get\_signin\_info\_by\_id | SELECT t.signin\_info\_sno, t.app\_user\_sno, t.push\_token, t.device\_type\_cd, t.device\_id, t.login\_on, t.logout\_on, t.description, t.active\_flag, f\_app\_user.name FROM masters.signin\_info t LEFT JOIN masters.app\_user f\_app\_user ON t.app\_user\_sno = f\_app\_user.app\_user\_sno WHERE t.signin\_info\_sno = quote\_nullable(i\_signin\_info\_sno); |
| update\_signin\_info | UPDATE masters.signin\_info SET signin\_info\_sno = quote\_nullable(i\_signin\_info\_sno), app\_user\_sno = quote\_nullable(i\_app\_user\_sno), push\_token = quote\_nullable(i\_push\_token), device\_type\_cd = quote\_nullable(i\_device\_type\_cd), device\_id = quote\_nullable(i\_device\_id), login\_on = quote\_nullable(i\_login\_on), logout\_on = quote\_nullable(i\_logout\_on), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag) WHERE signin\_info\_sno = quote\_nullable(i\_signin\_info\_sno); |
| create\_signin\_info | INSERT INTO masters.signin\_info (t.signin\_info\_sno, t.app\_user\_sno, t.push\_token, t.device\_type\_cd, t.device\_id, t.login\_on, t.logout\_on, t.description, t.active\_flag) VALUES (i\_signin\_info\_sno, i\_app\_user\_sno, i\_push\_token, i\_device\_type\_cd, i\_device\_id, i\_login\_on, i\_logout\_on, i\_description, i\_active\_flag); |
| delete\_signin\_info\_by\_id | DELETE FROM masters.signin\_info WHERE signin\_info\_sno = quote\_nullable(i\_signin\_info\_sno); |
| custom\_search\_signin\_info\_by\_id | SELECT t.signin\_info\_sno, t.app\_user\_sno, t.push\_token, t.device\_type\_cd, t.device\_id, t.login\_on, t.logout\_on, t.description, t.active\_flag, f\_app\_user.name FROM masters.signin\_info t LEFT JOIN masters.app\_user f\_app\_user ON t.app\_user\_sno = f\_app\_user.app\_user\_sno WHERE <custom\_search\_conditions>; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Sku Trace Wf**

### **Database Table Name: sku\_trace\_wf**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| sku\_trace\_wf\_sno | bigint - (Numeric Precision: 64) | sku\_trace\_wf\_pkey (PRIMARY KEY), |
| sku\_inventory\_sno | bigint - (Numeric Precision: 64) | sku\_trace\_wf\_sku\_inventory\_sno\_fkey (FOREIGN KEY), |
| wf\_stage\_cd | smallint - (Numeric Precision: 16) | sku\_trace\_wf\_wf\_stage\_cd\_fkey (FOREIGN KEY), |
| action\_time | timestamp without time zone | null |
| action\_by | bigint - (Numeric Precision: 64) | null |
| reference | text | null |
| description | text | null |
| active\_flag | boolean | null |
| media\_sno | bigint - (Numeric Precision: 64) | sku\_trace\_wf\_media\_sno\_fkey (FOREIGN KEY), |
| reject\_reason\_sno | bigint - (Numeric Precision: 64) | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Sku Trace Wf Sno | hidden | number | true | Manual Entry |
| Sku Inventory Sno | listbox | number | true | API feed list box selection |
| Wf Stage Cd | listbox | number | false | API feed list box selection |
| Action Time | textbox | string | false | Manual Entry |
| Action By | textbox\_number | number | false | Manual Entry |
| Reference | textbox | string | false | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Active Flag | checkbox | boolean | false | Manual Entry |
| Media Sno | listbox | number | false | API feed list box selection |
| Reject Reason Sno | textbox\_number | number | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_sku\_trace\_wfs | get | /api/v1/get\_sku\_trace\_wfs |
| create\_sku\_trace\_wf | post | /api/v1/create\_sku\_trace\_wf |
| update\_sku\_trace\_wf | put | /api/v1/update\_sku\_trace\_wf |
| delete\_sku\_trace\_wf | delete | /api/v1/delete\_sku\_trace\_wf |
| get\_medias | get | /api/v1/get\_medias |
| get\_sku\_inventorys | get | /api/v1/get\_sku\_inventorys |
| get\_codes\_dtls | get | /api/v1/get\_codes\_dtls |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_sku\_trace\_wfs | {} | {"data":{"media\_sno":"bigint","reference":"text","wf\_stage\_cd":"smallint","action\_by":"bigint","action\_time":"timestamp without time zone","reject\_reason\_sno":"bigint","description":"text","active\_flag":"boolean","sku\_trace\_wf\_sno":"bigint","sku\_inventory\_sno":"bigint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_sku\_trace\_wf | {"data":{"media\_sno":"bigint","reference":"text","wf\_stage\_cd":"smallint","action\_by":"bigint","action\_time":"timestamp without time zone","reject\_reason\_sno":"bigint","description":"text","active\_flag":"boolean","sku\_trace\_wf\_sno":"bigint","sku\_inventory\_sno":"bigint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_sku\_trace\_wf | {"data":{"media\_sno":"bigint","reference":"text","wf\_stage\_cd":"smallint","action\_by":"bigint","action\_time":"timestamp without time zone","reject\_reason\_sno":"bigint","description":"text","active\_flag":"boolean","sku\_trace\_wf\_sno":"bigint","sku\_inventory\_sno":"bigint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_sku\_trace\_wf | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_medias | {"id":"integer"} | {"data":"array of media"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_sku\_inventorys | {"id":"integer"} | {"data":"array of sku\_inventory"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_codes\_dtls | {"id":"integer"} | {"data":"array of codes\_dtl"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| get\_all\_sku\_trace\_wfs | SELECT t.sku\_trace\_wf\_sno, t.sku\_inventory\_sno, t.wf\_stage\_cd, t.action\_time, t.action\_by, t.reference, t.description, t.active\_flag, t.media\_sno, t.reject\_reason\_sno FROM masters.sku\_trace\_wf t ; |
| delete\_sku\_trace\_wf\_by\_id | DELETE FROM masters.sku\_trace\_wf WHERE sku\_trace\_wf\_sno = quote\_nullable(i\_sku\_trace\_wf\_sno); |
| custom\_search\_sku\_trace\_wf\_by\_id | SELECT t.sku\_trace\_wf\_sno, t.sku\_inventory\_sno, t.wf\_stage\_cd, t.action\_time, t.action\_by, t.reference, t.description, t.active\_flag, t.media\_sno, t.reject\_reason\_sno FROM masters.sku\_trace\_wf t WHERE <custom\_search\_conditions>; |
| get\_sku\_trace\_wf\_by\_id | SELECT t.sku\_trace\_wf\_sno, t.sku\_inventory\_sno, t.wf\_stage\_cd, t.action\_time, t.action\_by, t.reference, t.description, t.active\_flag, t.media\_sno, t.reject\_reason\_sno FROM masters.sku\_trace\_wf t WHERE t.sku\_trace\_wf\_sno = quote\_nullable(i\_sku\_trace\_wf\_sno); |
| custom\_delete\_sku\_trace\_wf\_by\_id | DELETE FROM masters.sku\_trace\_wf WHERE <custom\_conditions>; |
| custom\_update\_sku\_trace\_wf | UPDATE masters.sku\_trace\_wf SET sku\_trace\_wf\_sno = quote\_nullable(i\_sku\_trace\_wf\_sno), sku\_inventory\_sno = quote\_nullable(i\_sku\_inventory\_sno), wf\_stage\_cd = quote\_nullable(i\_wf\_stage\_cd), action\_time = quote\_nullable(i\_action\_time), action\_by = quote\_nullable(i\_action\_by), reference = quote\_nullable(i\_reference), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), media\_sno = quote\_nullable(i\_media\_sno), reject\_reason\_sno = quote\_nullable(i\_reject\_reason\_sno) WHERE <custom\_conditions>; |
| unique\_sku\_trace\_wf\_exists | SELECT 1 FROM masters.sku\_trace\_wf WHERE <unique\_conditions> LIMIT 1; |
| create\_sku\_trace\_wf | INSERT INTO masters.sku\_trace\_wf (t.sku\_trace\_wf\_sno, t.sku\_inventory\_sno, t.wf\_stage\_cd, t.action\_time, t.action\_by, t.reference, t.description, t.active\_flag, t.media\_sno, t.reject\_reason\_sno) VALUES (i\_sku\_trace\_wf\_sno, i\_sku\_inventory\_sno, i\_wf\_stage\_cd, i\_action\_time, i\_action\_by, i\_reference, i\_description, i\_active\_flag, i\_media\_sno, i\_reject\_reason\_sno); |
| update\_sku\_trace\_wf | UPDATE masters.sku\_trace\_wf SET sku\_trace\_wf\_sno = quote\_nullable(i\_sku\_trace\_wf\_sno), sku\_inventory\_sno = quote\_nullable(i\_sku\_inventory\_sno), wf\_stage\_cd = quote\_nullable(i\_wf\_stage\_cd), action\_time = quote\_nullable(i\_action\_time), action\_by = quote\_nullable(i\_action\_by), reference = quote\_nullable(i\_reference), description = quote\_nullable(i\_description), active\_flag = quote\_nullable(i\_active\_flag), media\_sno = quote\_nullable(i\_media\_sno), reject\_reason\_sno = quote\_nullable(i\_reject\_reason\_sno) WHERE sku\_trace\_wf\_sno = quote\_nullable(i\_sku\_trace\_wf\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |