Wegmans Deli Kiosk Web Services

Unless otherwise noted, all responses and sent packets should be in the form of JSON objects, with a key,value pair.

[Common JSON Data Packet Structures:](#h.u1sgyidih785)

[Order Object](#h.uf8g0zjrig4y)

[Ordered Item](#h.dh6gxmtpe04s)

[Completed Web Services](#h.5bm9iyao9n1p)

[TestSetup](#h.kv4woeoexc3g)

[GetEmployees](#h.1plg50l5nfz6)

[GetEmployee](#h.ym3fan8gfg8f)

[GetKiosks](#h.tef93nwgujo7)

[GetAllPrinters](#h.kyj48byti0zi)

[GetAllStores](#h.gptyfwgj2sm3)

[GetAllWorkFlows](#h.6d7hjz8jiu83)

[GetAllOrderItems](#h.8013mc5eb2qq)

[LoadDeliCatalogAsJSON](#h.ued34xf19vs3)

[GetOrdersFrom (FORMALLY GET ORDRDETAILS)](#h.dww4hge5nnb6)

[GetAllOrders](#h.5c1l4lgq3f9i)

[GetOrder](#h.cd3f7iynkim5)

[GetNewOrders](#h.urux8dmdwthr)

[ProcessOrder](#h.58699t3m0un8)

[ReprintOrder](#h.1lm5dhrf2msb)

[InitializePrinter](#h.8pno1zgob6v6)

[InitializeKiosk](#h.nqph86kuulne)

[GetItemImage](#h.jz6pmnhvpvzh)

[LoginManager](#h.im5r8qfctjvm)

[LogoutManager](#h.fvve51nm94uz)

[UpdateItem](#h.9qxsi9nbfaq5)

[GetSplashScreens](#h.ncdrmo4hv8x5)

[GetSplashScreenImage](#h.5te7anbzlaux)

[CheckSplashScreenUpdate](#h.ad1buifsrwgb)

[UpdatePrinterForKiosk](#h.estwtg3ztkeb)

[UploadSplashScreenImage](#h.xp0pzsd2up3r)

[RemoveSplashScreen](#h.j9if9bavhtxs)

[UpdateSplashScreenSequence](#h.7n1etpmrgz3y)

[UpdateSplashScreen](#h.tjrohvh3ytlo)

[AddSplashScreen](#h.hey1wn6pao2d)

[RemoveCoupon](#h.my628fnlzxxh)

[EditCoupon](#h.145butkfz37f)

[AddCoupon](#h.3unyqsm5z5e9)

[RemoveEmployee](#h.i9swoq3ya29a)

[EditEmployee](#h.xbuezr4pll0s)

[AddEmployee](#h.o0yxt4656dk2)

[RemoveKiosk](#h.6sjbihpa73)

[EditKiosk](#h.izqldsundpcs)

[RemovePrinter](#h.aa3emeb4zw78)

[EditPrinter](#h.d4qqt61u92an)

[AddPrinter](#h.gx7msynsqk65)

[CheckLastUpdate](#h.e2qlbzbwar92)

[GetAllItems](#h.dh1v22gb0e8e)

[GetKiosk](#h.jd9dgo1r1sex)

[UpdateItem](#h.gpcbilb6qw0r)

[SalesPerKioskReport](#h.30qw8tq8ibo)

[OrdersPerHourReport](#h.geah9toqki96)

[OrdersPerItemReport](#h.t8ps513ss50v)

[OrdersPerKioskReport](#h.6ixv61xf34wt)

[GetNewOrders](#h.2tpi77zh69la)

# 

# 

# Common JSON Data Packet Structures:

## Order Object

|  |  |  |
| --- | --- | --- |
| **Paramter Name** | **Type** | **Description** |
| DisplayID | int | Used to display order number for receipt. Takes original Order ID and mods by 999. |
| EmployeeID | int | ID of employee fulfilling order (not used) |
| KioskID | int | ID of kiosk that placed order |
| OrderID | int | ID of the Order |
| Timestamp | DateTime | Time of the order submission |
| TotalPrice | Decimal | Total estimated price of the order |
| Items | Array of OrderedItems | Array of Ordered Items that are contained in the order |

## Ordered Item

|  |  |  |
| --- | --- | --- |
| **Paramter Name** | **Type** | **Description** |
| itemID | int | ID of the base item used in the order |
| orderID | int | ID of the Order the item belongs to |
| orderItemID | int | ID of the item in the corresponding order |
| quantityOrdered | double | amount of quantity ordered |
| quantityType | string | denotes type of quanitty “pounds” or “slices” |
| quotedPrice | decimal | quoted price for the order based off tonnage and price |
| SliceType | string | String denoting the type of slice for the item |

# **Completed Web Services**

## TestSetup

**Purpose:**

To test the database connection for the web service.

**Type: GET**

**URL:**

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/TestSetup/

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| N/A | N/A | N/A |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| String | Single line string | “Found Jared Schutt” |

**Example(s):**

N/A

## GetEmployees

**Purpose:**

To retrieve a list of all of the employees in the database.

**Type: GET**

**URL:**

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetEmployees/

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| N/A | N/A | N/A |

**JSON Response:**

**An array containing Employee objects:**

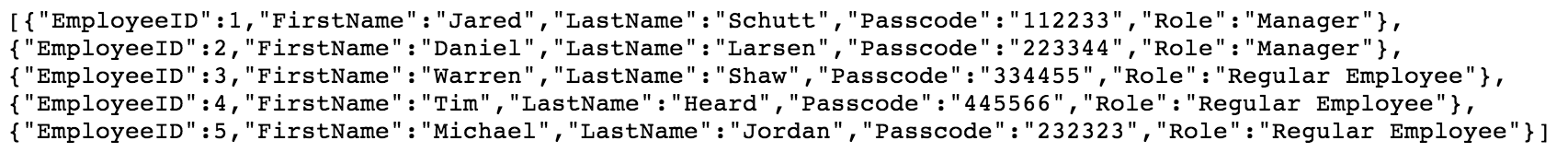
**Employee Structure:**

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Type** | **Description** |
| EmployeeID | int | ID of employee |
| FirstName | String | First Name of Employee |
| LastName | String | Last Name of Employee |
| Passcode | string | Pass code of Employee |
| Role | String | “manager” or “employee” |

**Example(s):**

request: <http://deli-cious/DeliServer/SystemController.svc/GetEmployees/>

response:



## GetEmployee

**Purpose:**

To retrieve a specific employee record from the database

**Type: GET**

**URL:**

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetEmployee?id={id}

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| id | int | the id of the employee you want to retrieve from the database |

**Response:**



**Single structure containing information of one employee**

|  |  |  |
| --- | --- | --- |
| **Paramter Name** | **Type** | **Description** |
| EmployeeID | int | ID of employee |
| FirstName | String | First Name of Employee |
| LastName | String | Last Name of Employee |
| Passcode | string | Pass code of Employee |
| Role | String | “manager” or “employee” |

**Example(s):**

request: http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetEmployee?id=1

response:



## GetKiosks

**Purpose:**

To retrieve all of the information for the kiosks in the database.

**Type : GET**

**URL:**

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetKiosks/

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| N/A | N/A | N/A |

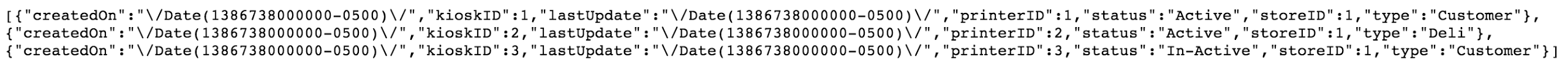
**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| JSON | {  “createdOn”,  “kioskID”,  “lastUpdate”,  “printerID”,  “status”,  “storeID”,  “type”  } | {  "createdOn":"\/Date(1386738000000-0500)\/",  "kioskID":1,  "lastUpdate":"\/Date(1386738000000-0500)\/",  "printerID":1,  "status":"Active",  "storeID":1,  "type":"Customer"  } |

**Example(s):**

request: <http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetKiosks/>

response:





## GetAllPrinters

**Purpose:**

To retrieve all of the information about the printers from the database.

**Type : GET**

**URL:**

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetAllPrinters/

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| N/A | N/A | N/A |

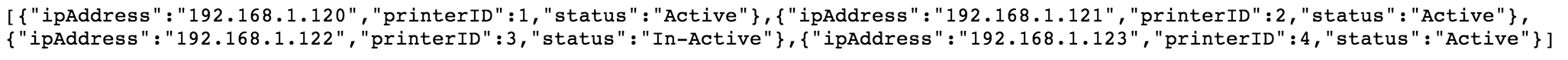
**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| JSON | {  ipAddress,  printerID,  status  } | {  "ipAddress":"192.168.1.120",  "printerID":1,  "status":"Active"  } |

**Example(s):**

request: <http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetAllPrinters/>

response:



## GetAllStores

**Purpose:**

To retrieve all of the information about the stores in the database.

**Type: GET**

**URL:**

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetAllStores/

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| N/A | N/A | N/A |

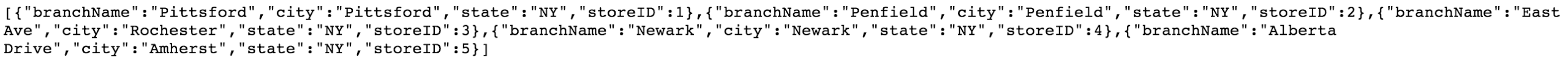
**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| JSON | {  branchName,  city,  state,  storeID  } | {  "branchName":"Pittsford",  "city":"Pittsford",  "state":"NY",  "storeID":1  } |

**Example(s):**

request: <http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetAllStores/>

response:



## GetAllWorkFlows

**Purpose:**

To retrieve all information from the database pertaining to the workflows stored there.

**Type: GET**

**URL:**

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetAllWorkFlows/

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| N/A | N/A | N/A |

**Response:**

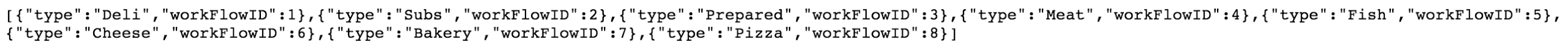
**Array of Workflow items:**

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Type** | **Description** |
| type | String | String name of the workflow |
| workFLowID | int | ID of Workflow. |

**Example(s):**

request: <http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetAllWorkFlows/>

response:



## GetAllOrderItems

**Purpose:**

To retrieve every order item from the database.

**Type: GET**

**URL:**

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetAllOrderItems/

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| N/A | N/A | N/A |

**Response:**

Array of Order Items

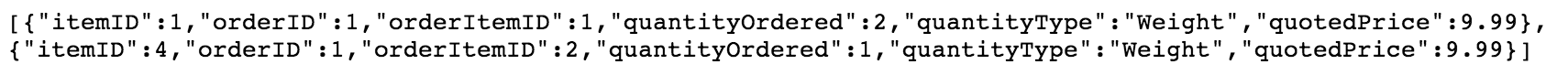
See “Order Item” structure.



**Example(s):**

request: <http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetAllOrderItems/>

response:



## LoadDeliCatalogAsJSON

**Purpose:**

To load the item catalog from the server as a json object

**Type: GET**

**URL:** http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/LoadDeliCatalogAsJSON?workflow={workflow}

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| workflow | String | the name of the workflow. For example, “DELI”. |

**Response:**

A N-Deep tree representing an item structure. Can be many levels deep, each level representing a category of items. In the case of Deli, this takes the form of Deli Category -> Item Category -> Specific Item

**Root Catalog JSON Object**

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Type** | **Description** |
| name | string | name of the catalog (“Deli”) |
| isLeaf | boolean | identifier if this level is a leaf or not. True if Leaf. False if has children |
| children | array of Categories | Array of Categories (children) **(See “Category” object below)** |

**Category JSON Object:**

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Type** | **Description** |
| name | string | category name |
| isLeaf | Boolean | identifier if this level is a leaf or not. True if Leaf. False if has children |
| type | string | Type identifying this level. In the case of Category, will = “Product Category” |
| children | array of Individual Items | Array of Individual Items(children)  **(See “Individual Item” below)** |

**Individual Item JSON Object**

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Type** | **Description** |
| name | String | Name of the Item |
| id | int | DeliKiosk ID (used for all calls) |
| wegmansId | int | Wegmans ID (used for search) |
| description | string | Description of the item |
| price | Decimal | Price of the item |
| onSpecial | Boolean | True if item is on special. False if regular |
| isAvailable | Boolean | True if available  False if not available |
| isLeaf | Boolean | True if it is a leaf (product)  False if there is another set of children |
| nextNavigation | **????** | **???** |
| children | Array of Children Items | Because these are products, children are null |
| customAttributes | **Custom Attribute Array** | Array of custom attributes for an item. |

**Example(s):**

request: [http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/LoadDeliCatalogAsJSON](http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/LoadDeliCatalogAsJSON/)?workflow=DELI

response:



## GetOrdersFrom (FORMALLY GET ORDRDETAILS)

**Purpose:**

Get an order from a specific DateTime to the present time.

**Type: GET**

**URL:** http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetOrdersFrom?date={DateTime}

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| id | string | the id of the order you want to get the details from. |

**Response:**

One Order Object corresponding to the ID provided.

See “[Order Object](#h.uf8g0zjrig4y)”







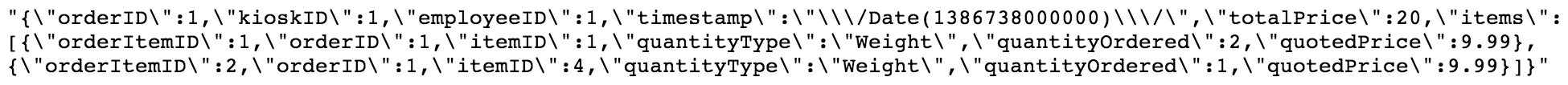




**Example(s):**

request: http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetOrdersFrom?date=DateTime

response:



## GetAllOrders

**Purpose:**

To retrieve all of the orders from the database.

**Type: GET**

**URL: http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetAllOrders/**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| N/A | N/A | N/A |

**Response:**

Array of Order Objects

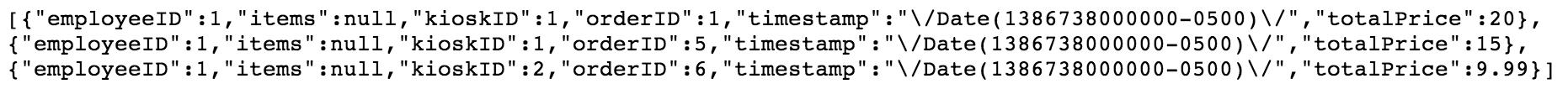
See “[Order Object](#h.uf8g0zjrig4y)”



**Example(s):**

request: http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetAllOrders/

response:



## GetOrder

**Purpose:**

To retrieve a specific order from the database.

**Type: GET**

**URL: http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetOrder?id={id}**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| id | string | the id of the order you want to get from the database. |

**Response:**

One Order Object corresponding to the ID sent

See “Order Object”

**Order Items will be NULL**



**Example(s):**

request: http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetOrder?id=1

response:



## GetNewOrders

**Purpose:**

To get any orders newer than a given order.

**Type: GET**

**URL:** http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetNewOrders?lastOrderID={lastOrderID}

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| lastOrderID | string | ID of the last order that the employee kiosk has. |

**Response:**

Array of Order Objects

See “[Order Object](#h.uf8g0zjrig4y)”

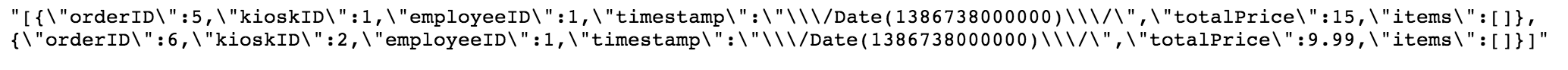
Order Objects contain order items.



**Example(s):**

**request:** http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetNewOrders?lastOrderID=1

**response:**



## ProcessOrder

**Purpose:**

To complete an order on the server.

**Type: POST**

**URL:** http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/ProcessOrder

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Type** | **Description** |
| KioskID | int | ID of the kiosk submitting the order |
| TotalPrice | Decimal | Total calculated price of the order |
| EmployeeID | NOT USED - Need to Deprecate | --------- |
| items | Array of Items objects | All items that are submitted with the order. See **Item** below. |

**Item:**

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Type** | **Description** |
| KioskID | int | ID of the kiosk submitting the order |
| QuantityType | string | “slices” or “pounds” depending on the type ordered. |
| QuantityOrdered | Double | Quantity Ordered. May be whole numbers for slices, or a double for pounds |
| QuotedPrice | Decimal | Quoted price for the item - based off quantity and base item price |
| SliceType | string | String used to denote slice type. May change - Currently “Thin” “Thick “Regular” |

**Response:**

**Response is a Dictionary of potential error messages**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Description** |
| EmployeePrinter | string | “Success” or “Fail” |
| CustomerPrinter | string | “Success” or “Fail” |
| GeneralError |  |  |

**Example(s):**

request:

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/ProcessOrder/

\*\* Note : must be a POST, and was tested using a Stream of data.

response:

1

## ReprintOrder

**Purpose:**

To re-print the deli ticket for a particular order.

**Type: GET**

**URL:** http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/ReprintOrder?orderID={id}

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Type** | **Description** |
| orderID | int | The ID of the order to reprint the order ticker for. |

**Response:**

**A Dictionary of potential error messages**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Description** |
| EmployeePrinter | string | “Success” or “Fail” |
| CustomerPrinter | string | “Success” or “Fail” |
| GeneralError |  |  |
| Success |  |  |

**Example(s):**

request:

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/ReprintOrder?orderID=346

response:

## InitializePrinter

**Purpose:**

To setup a printer for the use in the system.

**Type : POST**

**URL:** http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/InitializePrinter/

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| location | String | The Location of the networked printer  ex: 192.168.1.1. Eventually may be printserver names |
| workflow | String | the type of workflow : “DELI”, “PIZZA”, “CHEESE” |
| kioskType | String | the type of kiosk : “EMPLOYEE”, “CUSTOMER”. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| PrinterID | int | ID of the printer |
| location | string | Location of the printer |
| status | string | Status of the printer |

**Example(s):**

request:

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/InitializePrinter/

response:

"{\"printerID\":6,\"ipAddress\":\"192.168.1.149\",\"status\":\"Active\"}"

## InitializeKiosk

**Purpose:**

To setup a kiosk for the use in the system.

**Type: PUT**

**URL:** http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/InitializeKiosk/

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| kioskDetails | Kiosk  Must set values for :  type = DELI\_CUSTOMER, DELI\_EMPLOYEE  printerID  workID = 1  storeID = 1 | The kiosk to initialize |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| JSON | A kiosk object with the following format:  {  kioskID,  printerID,  storeID,  createdOn,  lastUpdate,  status,  workID  } | "{\"kioskID\":9,  \"printerID\":2,  \"storeID\":1,  \"createdOn\":\"\\\/Date(1393207743548)\\\/\",  \"lastUpdate\":\"\\\/Date(1393207743548)\\\/\",  \"status\":\"Active\",  \"type\":\"DELI\",  \"workID\":1}" |

**Example(s):**

request:

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/InitializeKiosk/

response:

"{\"kioskID\":9,

\"printerID\":2,

\"storeID\":1,

\"createdOn\":\"\\\/Date(1393207743548)\\\/\",

\"lastUpdate\":\"\\\/Date(1393207743548)\\\/\",

\"status\":\"Active\",

\"type\":\"DELI\",

\"workID\":1}"

## GetItemImage

**Purpose:**

To retrieve an image for a specific item.

**Type: GET**

**URL:** http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetItemImage?itemID={itemID}

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| itemID | int | The id of the item |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| Image (jpg) | N/A | N/A |

**Example(s):**

request:

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetItemImage?itemID=19

response:



## LoginManager

**Purpose:**

To validate the credentials entered for a manager user.

**Type: POST**

**URL:** http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/LoginManager/

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| passcode | string | the passcode of the manager that is trying to login. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| boolean | true or false, depending on if the login was successful or not. | true or false |
| string | userID | employee ID |

**Example(s):**

request: [http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/LoginManager](http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/LoginManager?passcode)/

response:

true

## LogoutManager

**Purpose:**

To logout a manager user and invalidate their session.

**Type: POST**

**URL:** http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/LogoutManager/

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| Passcode | Integer | The passcode of the manager that wants to logout. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| boolean | true or false, depending on if the logout was successful or not. | true or false |

**Example(s):**

request:http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/LogoutManager?passcode=112233

response: true

## UpdateItem

**Purpose:**

To modify an item in the system.

**URL:** http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/UpdateItem?item={item}

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| item | Item | the item that needs to be updated in the database. |
| managerID | int | ID of the manager currently logged in |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| boolean | true or false, depending on if the update was successful or not. | true |

**Example(s):**

request: http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/UpdateItem

response: true

## GetSplashScreens

**Purpose:**

To get the ids of the splash screens for the kiosk.

**URL:** http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetSplashScreens/

**Type: GET**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| N/A | N/A | N/A |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| List | [ screenID1, screenID2, … ] | [1,2,3] |

**Example(s):**

**request:** <http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetSplashScreens/>

response: [1,2]

## GetSplashScreenImage

**Purpose:**

To retrieve the splash screen image to display on the kiosk.

**URL:** deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetSplashScreenImage?screenID={screenID}

**Type: GET**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| screenID | int | the id of the splashScreen you want to get the image for. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| image/jpeg | N/A | N/A |

**Example(s):**

request: deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetSplashScreenImage?screenID=1

response:



## CheckSplashScreenUpdate

**Purpose:**

Check the timestamp of when the splash screens were last updated.

**URL:**

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/CheckSplashScreenUpdate?workflow={workflow

**Type: GET**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| workflow | string | The name of the workflow that you want to check the last splash screen update for. Example: “DELI” |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| Dictionary<String,String> | { lastUpdate : “Timestamp”} | { lastUpdate : “*4/22/2014 8:03:57 PM*”} |

**Example(s):**

request:

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/CheckSplashScreenUpdate?workflow=DELI

response:

*Object {lastUpdate: "4/22/2014 8:03:57 PM"}*

## UpdatePrinterForKiosk

**Purpose:**

To update the printer that is associated with a particular kiosk, so that the proper print commands can be sent to the proper printer.

**URL:**

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/UpdatePrinterForKiosk/

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| updateDetails | Dictionary<String,String>  {  kioskID : int,  printerID : int  } | This is a json object that contains the id of the kiosk we want to update the printer for and the id of the printer that we want to give to the kiosk. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| Dictionary<String,String> | {“result” : “boolean”} | { “result” : “true” } |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/UpdatePrinterForKiosk/>

response:

Object{ "result" : "true" }

## UploadSplashScreenImage

**Purpose:**

Used to upload a new splash screen image file to the system.

**URL:**

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/UploadSplashScreenImage/>

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| screenImage | image file (jpg, png) | The splash screen image file you want to add to the system. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| Dictionary<String,String> | { “result” : “boolean” } | { “result” : “true” } |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/UploadSplashScreenImage/>

response:

{ “result” : “true” }

## RemoveSplashScreen

**Purpose:**

Used to remove an existing splash screen from the system.

**URL:**

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/RemoveSplashScreen/>

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| screenID | int | id of the splash screen you want to remove |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| string | boolean | “true” |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/RemoveSplashScreen/>

response:

true

## UpdateSplashScreenSequence

**Purpose:**

Used to update the sequence of splash screens in the system and the order that they will be displayed on the client application.

**URL:** [http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/UpdateSplashScreenSequence/](http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/RemoveSplashScreen/)

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| screens | List<int> | A list of ids of the splash screens, in order. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| Dictionary<String,String> | { “result” : “boolean” } | { “result” : “true” } |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/UpdateSplashScreenSequence/>

response:

{ “result” : “true” }

## UpdateSplashScreen

**Purpose:**

Used to update an existing splash screen.

**URL:**

http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/UpdateSplashScreen/

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| splashScreen | SplashScreen | The splash screen you want to update, which has the following fields occupied:   * isActive * screenID |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| string | true or false | “true” |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/UpdateSplashScreen/>

response:

“true”

## AddSplashScreen

**Purpose:**

Used to add a new splash screen to the system.

**URL:**

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/AddSplashScreen/>

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| splashScreen | SplashScreen | A splash screen object with all of the values set except for screenID. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| string | true or false | “true” |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/AddSplashScreen/>

response:

“true”

## RemoveCoupon

**Purpose:**

Used to remove an existing coupon from the system.

**URL:**

<http://deli-cious.se.rit.edu/DeliSever/SystemController.svc/RemoveCoupon/>

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| couponID | int | id of the coupon you want to delete |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| string | true or false | “true” |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/RemoveCoupon/>

response:

“true”

## EditCoupon

**Purpose:**

Used to edit an existing coupon in the system.

**URL:**

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/EditCoupon/>

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| coupon | Coupon | A coupon object that you would like to update to the system. It must be populated with all of the values that you wish to update. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| String | true or false | “true” |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/EditCoupon/>

response:

“true”

## AddCoupon

**Purpose:**

Used to add a new coupon to the system.

**URL:**

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/AddCoupon/>

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| coupon | Coupon | A coupon object that you would like to add to the system. It must be populated with values for every variable except for couponID. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| string | true or false | “true” |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/AddCoupon/>

response:

“true”

## RemoveEmployee

**Purpose:**

Used to remove an existing employee from the system.

**URL:**

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/RemoveEmployee/>

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| employeeID | int | id of the employee to remove |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| string | true or false | “true” |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/RemoveEmployee/>

response:

“true”

## EditEmployee

**Purpose:**

Used to update an existing employees information.

**URL:**

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/EditEmployee/>

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| employee | Employee | The employee that you would like to update the data for. It should be populated with everything, including the employeeID. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| string | true or false | “true” |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/EditEmployee/>

response:

“true”

## AddEmployee

**Purpose:**

Used to add a new employee to the system.

**URL:**

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/AddEmployee/>

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| employee | Employee | The employee that you want to add to the system. The employee object should have values entered for everything except for employeeID. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| string | true or false | “true” |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/AddEmployee/>

response:

“true”

## RemoveKiosk

**Purpose:**

Used to remove an existing kiosk from the system.

**URL:**

[http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/RemoveKiosk/](http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/AddEmployee/)

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| kioskID | int | id of the kiosk to remove |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| string | true or false | “true” |

**Example(s):**

request:

[http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/RemoveKiosk/](http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/AddEmployee/)

response:

“true”

## EditKiosk

**Purpose:**

Used to edit an existing kiosk in the system.

**URL:**

[http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/EditKiosk/](http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/AddEmployee/)

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| kiosk | Kiosk | The kiosk that you want to update in the system. Should have values for every variable. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| string | true or false | “true” |

**Example(s):**

request:

[http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/EditKiosk/](http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/AddEmployee/)

response:

“true”

## RemovePrinter

**Purpose:**

Used to remove an existing printer from the system.

**URL:**

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/RemovePrinter/>

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| printerID | int | id of the printer to remove |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| string | true or false | “true” |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/RemovePrinter/>

response:

“true”

## EditPrinter

**Purpose:**

Used to update an existing printer in the system.

**URL:**

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/EditPrinter/>

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| printer | Printer | The printer that you want to update in the system. Should have values set for every variable in the printer. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| string | true or false | “true” |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/EditPrinter/>

response:

“true”

## AddPrinter

**Purpose:**

Used to add a new printer to the system.

**URL:**

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/AddPrinter/>

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| printer | Printer | The printer that you want to add to the system. It should have variables set for the entire object except for printerID. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| string | true or false | “true” |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/AddPrinter/>

response:

“true”

## CheckLastUpdate

**Purpose:**

Used to check for when the last time that the item catalog was updated.

**URL:**

[http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/CheckLastUpdate?workflow={workflow](http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/CheckLastUpdate?workflow=%7Bworkflow)}

**Type: GET**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| workflow | String | name of the workflow to update : “DELI”, “PIZZA”, etc. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| DateTime | “\/Date()\/” | "\/Date(1399584987000-0400)\/" |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/CheckLastUpdate?workflow=DELI>

response:

"\/Date(1399584987000-0400)\/"

## GetAllItems

**Purpose:**

Used to get a list of all of the items in the system.

**URL:**

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetAllItems/>

**Type: GET**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| N/A | N/A | N/A |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| List<Item> | List of item objects | List of item objects |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetAllItems/>

response:



## GetKiosk

**Purpose:**

Used to get a specific kiosk from the system.

**URL:**

[http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetKiosk?kioskID={kioskID](http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetKiosk?kioskID=%7BkioskID)}

**Type: GET**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| kioskID | int | id of the kiosk you want to get from the system. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| Kiosk | Kiosk object | {"BackColor":"#ff00ff",  "BorderColor":"#ff00ff",  "ThemeColor":"#ff08ff",  "kioskID":1,  "printerID":1,  "status":"Active",  "storeID":1,  "type":"DELI\_CUSTOMER",  "workID":1} |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetKiosk?kioskID=1>

response:

{"BackColor":"#ff00ff","BorderColor":"#ff00ff","ThemeColor":"#ff08ff","kioskID":1,"printerID":1,"status":"Active","storeID":1,"type":"DELI\_CUSTOMER","workID":1}

## UpdateItem

**Purpose:**

Used to update an existing item in the system.

**URL:**

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/UpdateItem/>

**Type: POST**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| details | Dictionary<String,String> | Dictionary object with all variables set to the variables of the Item object. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| boolean | true or false | true |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/UpdateItem/>

response:

true

## SalesPerKioskReport

**Purpose:**

Used to get generate a report of the number of sales per kiosk for a particular week.

**URL:**

[http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/SalesPerKioskReport?from={from](http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/SalesPerKioskReport?from=%7Bfrom)}

**Type: GET**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| from | DateTime | the date to start the report from. Will always be a sunday and be in the format:   * “MM/dd/yyyy” |

**Response:**

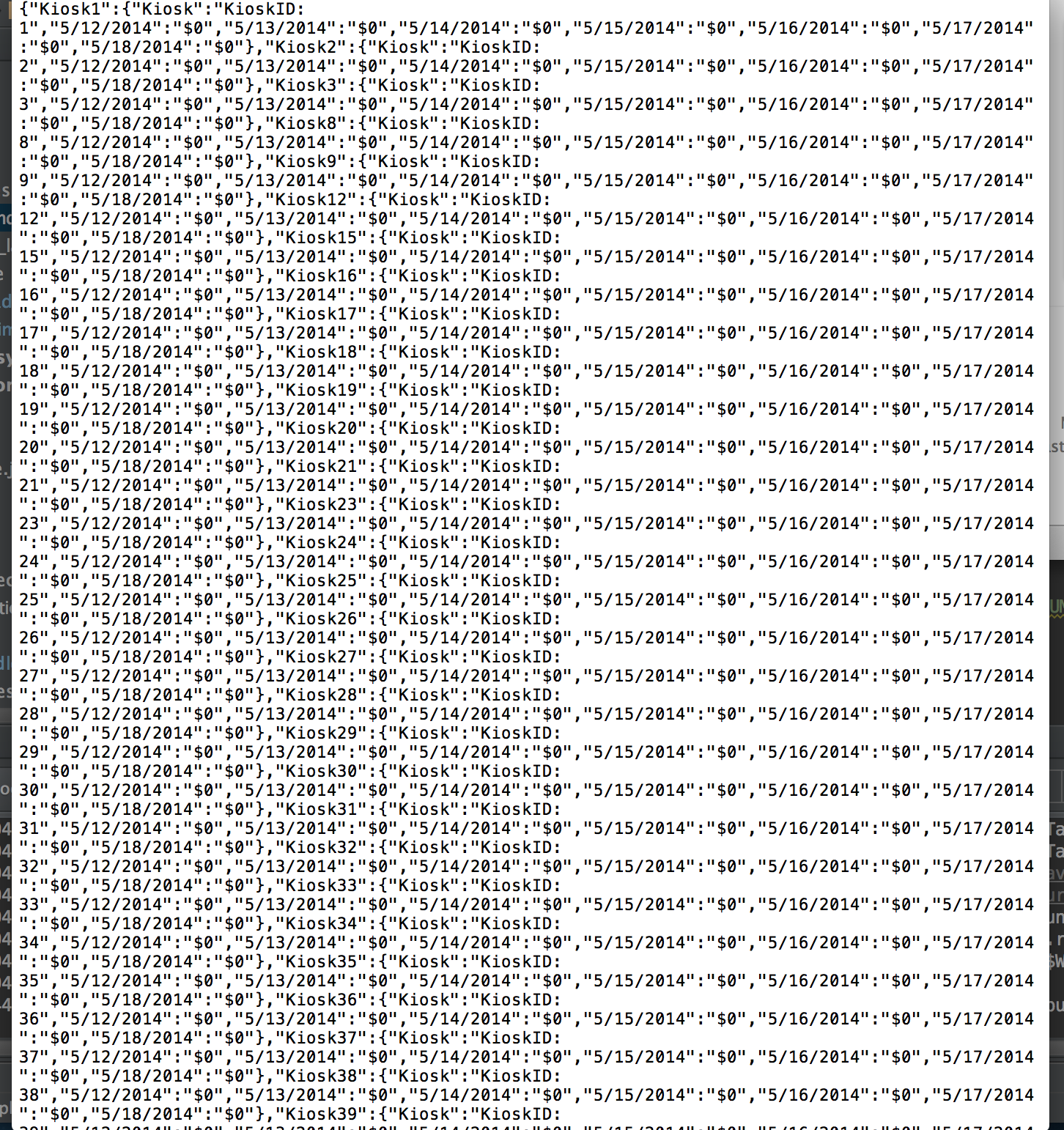
|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| json object | Contains a collection of kiosk sales data in the following format:  { “Kiosk” :  “KioskID” : “int”,  “Date1” : “sales string”,  “Date2” : “sales string”,  “Date3” : “sales string”,  “Date4” : “sales string”,  “Date5” : “sales string”,  “Date6” : “sales string”,  “Date7” : “sales string”} | {"Kiosk1":  {"Kiosk":  "KioskID:1",  "5/12/2014":"$0",  "5/13/2014":"$0",  "5/14/2014":"$0",  "5/15/2014":"$0",  "5/16/2014":"$0",  "5/17/2014":"$0",  "5/18/2014":"$0"} |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/SalesPerKioskReport?from=05/12/2014>

response:



## OrdersPerHourReport

**Purpose:**

Used to generate a report of the number of orders per hour.

**URL:**

[http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/OrdersPerHourReport?from{from](http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/OrdersPerHourReport?from%7Bfrom)}

**Type: GET**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| from | DateTime | the date to start the report from. Will always be a sunday and be in the format:   * “MM/dd/yyyy” |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| json object | Contains a collection of the number of orders per hour in the following format:  rowIndex:  {"Day":"Monday"  ,"09:00:00 - 10:00:00":"0",  "10:00:00 -11:00:00":"0",  "11:00:00 - 12:00:00":"8",  "12:00:00 - 13:00:00":"0",  "13:00:00 - 14:00:00":"0",  "14:00:00 - 15:00:00":"0",  "15:00:00 - 16:00:00":"0",  "16:00:00 - 17:00:00":"0",  "17:00:00 - 18:00:00":"0",  "18:00:00 - 19:00:00":"2",  "19:00:00 - 20:00:00":"1",  "20:00:00 - 21:00:00":"0"} | "0":{"Day":"Monday","09:00:00 - 10:00:00":"0","10:00:00 - 11:00:00":"0","11:00:00 - 12:00:00":"8","12:00:00 - 13:00:00":"0","13:00:00 - 14:00:00":"0","14:00:00 - 15:00:00":"0","15:00:00 - 16:00:00":"0","16:00:00 - 17:00:00":"0","17:00:00 - 18:00:00":"0","18:00:00 - 19:00:00":"2","19:00:00 - 20:00:00":"1","20:00:00 - 21:00:00":"0"} |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/OrdersPerHourReport?from=05/12/2014>

response:



## OrdersPerItemReport

**Purpose:**

Used to generate a report of number of orders per item.

**URL:**

[http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/OrdersPerItemReport?from={from](http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/OrdersPerItemReport?from=%7Bfrom)}

**Type: GET**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| from | DateTime | the date to start the report from. Will always be a sunday and be in the format:   * “MM/dd/yyyy” |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| json object | "itemNumber":  {"ItemNumber":"int",  "ProductName":"string",  "Date1":numOrders,  Date2:numOrders,  Date3:numOrders,  Date4:numOrders,  Date5:numOrders,  Date6:numOrders,  Date7:numOrders  "} | "item1"  :{"ItemNumber":"8846",  "ProductName":"Buffalo Style Chicken (Perdue)",  "5/12/2014":"0",  "5/13/2014":"0",  "5/14/2014":"0",  "5/15/2014":"0",  "5/16/2014":"0",  "5/17/2014":"0",  "5/18/2014":"0"} |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/OrdersPerItemReport?from=05/12/2014>

response:



## OrdersPerKioskReport

**Purpose:**

Used to generate a report of the number of order per kiosk for a specified week.

**URL:**

[http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/OrdersPerKioskReport?from={from](http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/OrdersPerKioskReport?from=%7Bfrom)}

**Type: GET**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| from | DateTime | the date to start the report from. Will always be a sunday and be in the format:   * “MM/dd/yyyy” |

**Response:**

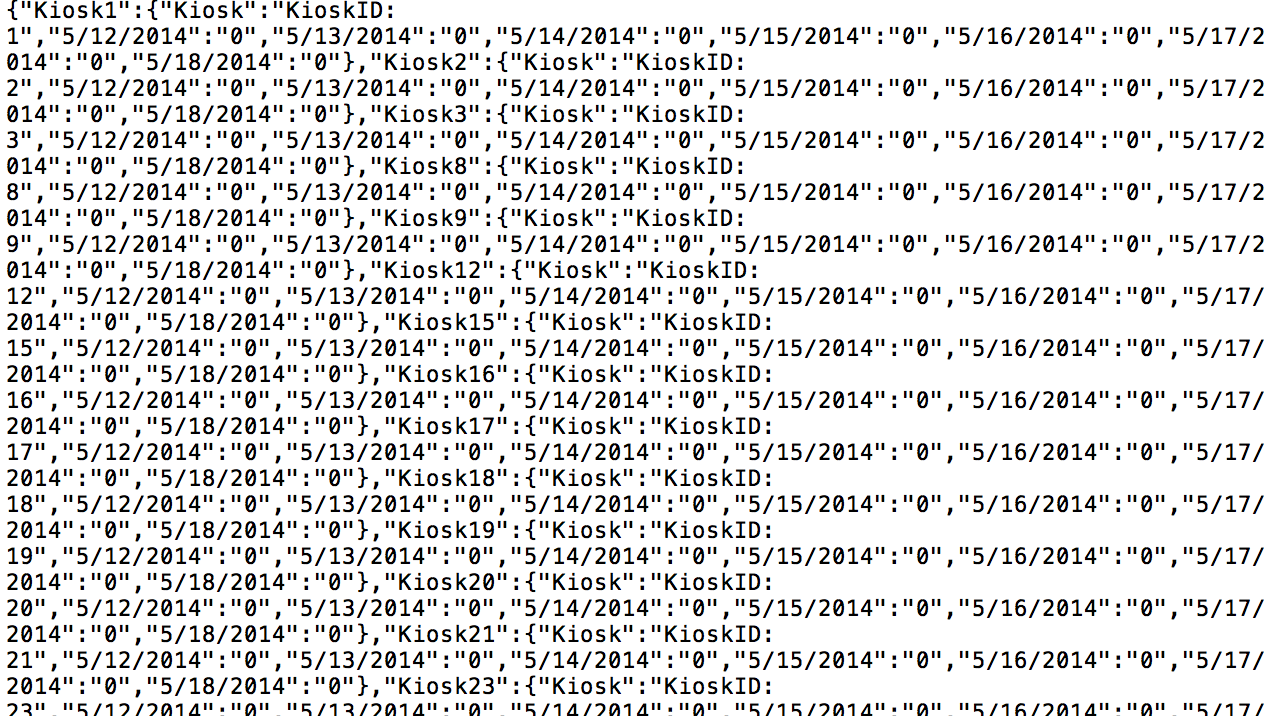
|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| json object | Kiosk orders report with the following format:  {"Kiosk1":  {"Kiosk":"KioskID: 1",  "5/12/2014":"0",  "5/13/2014":"0",  "5/14/2014":"0",  "5/15/2014":"0"  ,"5/16/2014":"0",  "5/17/2014":"0",  "5/18/2014":"0"} | {"Kiosk1":  {"Kiosk":  "KioskID:1",  "5/12/2014":"0",  "5/13/2014":"0",  "5/14/2014":"0",  "5/15/2014":"0",  "5/16/2014":"0",  "5/17/2014":"0",  "5/18/2014":"0"} |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/OrdersPerKioskReport?from=05/12/2014>

response:



## GetNewOrders

**Purpose:**

Used to get a list of all of the new orders in the system from a specific time and to the present.

**URL:**

[http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetNewOrders?lastOrderID={lastOrderID](http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetNewOrders?lastOrderID=%7BlastOrderID)}

**Type: GET**

**Parameters:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Explanation** |
| lastOrderID | int | id of the last order that client application already has. |

**Response:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Structure** | **Example** |
| List<Order> | List of order objects | [{"OrderID":11,  "KioskID":1,  "EmployeeID":1  ,"Timestamp":"\/Date(1393201637000)\/",  "TotalPrice":"22.5",  "Items":[  {"OrderItemID":4,  "OrderID":11,  "ItemID":1,  "QuantityType":"0"  ,"QuantityOrdered":4,  "QuotedPrice":"9.99",  "customAttributes":[]},  {"OrderItemID":5,  "OrderID":11,  "ItemID":2,"  QuantityType":"0"  ,"QuantityOrdered":3,  "QuotedPrice":"10",  "customAttributes":[]}],  "DisplayID":1} |

**Example(s):**

request:

<http://deli-cious.se.rit.edu/DeliServer/SystemController.svc/GetNewOrders?lastOrderID=1>

response:

