

# Table Documentation

Database Team

Spring Semester 2020

## 1 ItemHasOptions

- itemOptionID (int): Identifier for the menu-item/option combination
- itemID (int): Identifier for the menu-item
- optionID (int): Identifier for the option associated with the itemID

## 2 ItemHasSides

- id (int): Identifier for the menu-item/side combination
- menuItemID (int): Identifier for the menu-item
- sideItemID (int): Identifier for the side associated with the menuItemID
- additionalPrice (decimal): Price for the side to be added onto the total cost of the menu-item and side combined.

## 3 Menu

- menuID (int): Identifier for the menu
- menuStatus (boolean): 0 = menu is inactive; 1 = menu is active
- menuName (string): The name of the menu (ex: Dinner, Happy Hour...)
- restaurantID (int): The identifier for the restaurant using this menu

## 4 MenuContains

- containsID (int): Identifier for the unique pairing of menu, item, and price
- menuID (int): Identifier for the menu containing the item
- menuItemID (int): Identifier for a menu item present on the corresponding menu
- price (decimal): Holds the price of the menu item

## 5 MenuItem

- itemID (int): Identifier for the menu item
- itemName (string): The name of the menu item
- description (string): A brief description of the menu item
- image (string): The path or url to the image of the menu item
- itemStatus (boolean): 0 = inactive; 1 = item is active
- calories (int): The number of calories the menu item has
- category (string): The category of the menu item
- gluten (boolean): 0 = menu item is gluten-free; 1 = menu item contains gluten
- meat (boolean): 0 = menu item doesn't contain meat; 1 = menu item contains meat
- dairy (boolean): 0 = menu item doesn't contain dairy; 1 = menu item contains dairy
- nuts (boolean): 0 = menu item doesn't contain nuts; 1 = menu item contains nuts
- soy (boolean): 0 = menu item doesn't contain soy; 1 = menu item contains soy

## 6 MenuTimes

- timeID (int): Identifier for the time pairing
- menuID (int): Identifier for the menu starting and ending at the specified times
- startTime (int): Time menu starts being available (ex: 1100 = 11:00 AM)
- endTime (int): Time menu stops being available (ex: 1100 = 11:00 AM)

## 7 Options

- optionID (int): Identifier for the option
- optionName (int): Descriptive name of the option
- price (decimal): Price of the option on it's own
- optionStatus (boolean): 0 = inactive; 1 = active (default)

## 8 OrderItem

- orderItemID (int): Identifier for the order item
- menuItemID (int): The menu item associated with the order item
- quantity (int): The quantity ordered of this item
- comments (string): Any additional instructions or requests for the order item
- orderID (int): Identifier for the order containing the order items
- price (decimal): The price of the order item (menu item price \* quantity)

## 9 OrderItemHasOptions

- orderItemOptionID (int): Identifier for the combination of order-item and option
- orderItemID (int): Identifier for the order-item
- optionID (int): Identifier for the option to be associated with the order-item

## 10 Orders

- orderID (int): Identifier for the order
- tableID (int): Identifier for the table the order was placed at
- orderTime (DateTime): Time the order was submitted
- chargeAmount (decimal): Total order cost
- userID (int): ID of the customer that placed the order
- orderStatus (boolean): 0 = not complete, 1 = complete

## 11 Restaurant

- restaurantID (int): Identifier for a restaurant
- restaurantName (string): Name of the restaurant
- description (string): Description for the restaurant
- address (string): Street address for the restaurant
- salesTax (decimal): Tax percentage for the restaurant

- city (string): The restaurant's city
- state (string): The restaurant's state as a postal abbreviation (ex. MN)
- zipCode (int): The restaurant's zip code
- country (string): The restaurant's country, no abbreviation (ex. United States)

## 12 RestaurantManagers

- managerID (int): Identifier for a the manager
- restaurantID (int): Identifier for the restaurant being managed
- userID (int): Identifier for the user entry for the manager

## 13 RestaurantTables

- tableID (int): Identifier for the table
- alexaID (string): Identifier for the Alexa at the table
- qrCode (string): QR code that associates the table to the customer sitting at the table (scannable on the app)
- restuarantID (int): Identifier for the restaurant

## 14 Users

- userID (int): Identifier for the user
- firstName (string): User's first name
- lastName (string): User's last name
- email (string): User's email address
- token (string): Another identifier for the user