

API

method	body	URL	response
POST	{ name, price }	/items	Item.itemsID
GET		/items/:itemsID	Item
GET		/items/:name	Item
POST		/carts	Cart.cartsID
POST	{ discount?, taxes? }	/carts	Cart.cartsID
GET		/carts/:cartsID	Cart
PATCH	{ discount?, taxes? }	/carts/:cartsID	Cart
DELETE		/carts/:cartsID	
GET		/carts/:cartsID/items	Item[]
PUT		/carts/:cartsID/items/:itemsID	Cart
DELETE		/carts/:cartsID/items/:itemsID	

Types Reference

Reponse

```
interface Item {
  itemsID: number;
  name: string; /* Must be unique and cannot be a valid integer, e.g., "1337" */
  price: number;
}

interface Cart {
  cartsID: number;
  subtotal: number;
  discount: number;
  taxes: number;
  total: number;
}
```

Ignored values

ID values are automatically generated in creation (POST) requests, and thus an ID value passed in the body of such a request will be ignored.

Likewise, any attributed passed but not specified in the documentation will be ignored.

Optional values

Null values default to zero for numbers and empty array for arrays. API calls (such as GET requests) will always populate optional values with their defaults, so a GET request returning a type `Cart` will always return an object with a `Cart.subtotal` field, a `Cart.total` field, etc.

Errors

Unless specified in the documentation, the response body will never be empty/null on a successful request. Instead, errors will be reported via HTTP status codes.