# Article.class

Artikel anlegen:

createArticle(Article article)

mittels POST an die URL:

http://localhost:8080/rest/article/

JSON Example:

{"name":"Milch", "unit":"Pck"}

alle Artikel anzeigen:

public Boolean createArticle(Article article)

mittels GET an die URL:

http://localhost:8080/rest/article/

bestimmten Artikel anzeigen:

public Article findArticle(Long id)

mittels GET an die URL:

http://localhost:8080/rest/article/1

(wobei 1 die jeweilige Artikel ID ist)

bestimmten Artikel löschen:

public void deleteArticle(Article article)

mittels DELETE an die URL:

http://localhost:8080/rest/article/1

(wobei 1 die jeweilige Artikel ID ist)

bestimmten Artikel bearbeiten:

public void editArticle(Long id, Article article)

mittels POST an die URL:

http://localhost:8080/rest/article/1

(wobei 1 die jeweilige Artikel ID ist)

JSON Example:

{"unit":"kg"}

(wobei die zu bearbeitenden Daten gefüllt sein müssen)

# User.class

Nutzer anlegen:

public void createUser(User user)

mittels POST an die URL:

http://localhost:8080/rest/user/

JSON Example:

{"name":"sven", "keyNumber":"123abc"}

alle Benutzer anzeigen:

public List<User> findAllUser()

mittels GET an die URL:

http://localhost:8080/rest/user/

bestimmten Benutzer anzeigen:

public User findUser(Long id)

mittels GET an die URL:

http://localhost:8080/rest/user/1

(wobei 1 die jeweilige Benutzer ID ist)

bestimmten Benutzer löschen:

public void deleteUser(User user)

mittels DELETE an die URL:

http://localhost:8080/rest/user/1

(wobei 1 die jeweilige Nutzer ID ist)

bestimmten Benutzer bearbeiten:

public void editUser(Long id, User user)

mittels POST an die URL:

http://localhost:8080/rest/user/1

(wobei 1 die jeweilige Nutzer ID ist)

JSON Example:

{"keyNumber":"123c"}

(wobei die zu bearbeitenden Daten gefüllt sein müssen)

ShoppingListen eines bestimmten Nutzers anzeigen:

public String getShoppingListFromUser(@PathParam("id")Long id)

mittels GET an die URL: http://localhost:8080/rest/user/sl/1

(wobei 1 die jeweilige Benutzer ID ist)

# ShoppingList.class

ShoppingList anlegen:

public void createShoppingList(ShoppingList shoppingList)

mittels POST an die URL:

http://localhost:8080/rest/sl/

JSON Example:

{"name":"schnulli", "status":"0", "userList":[{"id":"2"}]}

(hier muss eine userList mit gesenden werden, da eine ManyToMany Verknüpfung mit User.class besteht)

alle ShoppingLists anzeigen:

public List<ShoppingList> findAllShoppingLists()

mittels GET an die URL:

http://localhost:8080/rest/sl/

bestimmten ShoppingLists anzeigen:

public ShoppingList findShoppingList(Long id)

mittels GET an die URL:

http://localhost:8080/rest/sl/1

(wobei 1 die jeweilige ShoppingList ID ist)

bestimmten ShoppingLists löschen:

public void deleteShoppingList(ShoppingList shoppingList)

mittels DELETE an die URL:

http://localhost:8080/rest/sl/1

(wobei 1 die jeweilige ShoppingList ID ist)

zu beachten ist, dass natürlich alle enthaltenen ListEntries gelöscht werden

bestimmten Nutzer einer ShoppingList hinzufügen:

public void addUserToList(ShoppingList shoppingList)

mittels POST an die URL:

http://localhost:8080/rest/article/1

(wobei 1 die jeweilige ShoppingList ID ist)

JSON Example:

{"id":"1", "userList":[{"id":"2"}]}

(die jeweilige id reicht zum hinzufügen des Nutzers)

bestimmten Nutzer von einer ShoppingList löschen:

public void deleteUserFromList(Long id, Long slID)

# ListEntry.class

ListEntry anlegen:

public void createEntry(ListEntry listEntry)

mittels POST an die URL:

http://localhost:8080/rest/le/

JSON Example:

{"article":{"id": "1"}, "shoppingList":{"id":"1"}, "user":{"id":"1"}, "quantity": "3", "addDate": "", "buyDate":""}

alle ListEntries anzeigen:

public List<ListEntry> findAllListEntries()

mittels GET an die URL:

http://localhost:8080/rest/le/

bestimmten ListEntries anzeigen:

public ListEntry findListEntry(Long id)

mittels GET an die URL:

http://localhost:8080/rest/le/1

(wobei 1 die jeweilige ListEntry ID ist)

bestimmten ListEntries löschen:

public void deleteListEntry(ListEntry listEntry)

mittels DELETE an die URL:

http://localhost:8080/rest/le/1

(wobei 1 die jeweilige ListEntry ID ist)

bestimmten ListEntries bearbeiten:

public void editListEntry(Long id, ListEntry listEntry)

mittels POST an die URL:

http://localhost:8080/rest/le/1

(wobei 1 die jeweilige ListEntry ID ist)

JSON Example:

{"article":{"id": "1"}, "quantity": "1"}

(nicht zu ändernde Felder bleiben leer)

# Recepe.class

# RecepeEntry.class