Api Documentation

Overview

Api Documentation

Version information

Version : 1.0

License information

License : Apache 2.0
License URL : http://www.apache.org/licenses/LICENSE-2.0
Terms of service : urn:tos

URI scheme

Host : localhost:8082
BasePath : /

Tags

- customer-controller : Customer Controller
- order-controller : Order Controller
- order-detail-controller : Order Detail Controller
- payment-controller : Payment Controller
- product-controller : Product Controller
- supplier-controller : Supplier Controller

Paths

Creates a new Customer.

```
POST /customer
```

Parameters

Type	Name	Description	Schema
Body	**newCustomer** *required*	newCustomer	Customer

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- customer-controller

Returns a Customer by Order identifier. 404 if does not exist.

```
GET /customer/order/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- order-controller

Returns list of all Customers in the system.

```
GET /customers
```

Responses

HTTP Code	Description	Schema
200	OK	< Customer > array
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- application/json

Tags

- customer-controller

Returns a specific Customer by their identifier. 404 if does not exist.

```
GET /customers/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- `application/json`

Tags

- customer-controller

Deletes a Customer from the system. 404 if the customer's identifier is not found.

```
DELETE /deleteCustomer/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
204	No Content	No Content
401	Unauthorized	No Content
403	Forbidden	No Content

Produces

- `application/json`

Tags

- customer-controller

Creates a new Order detail.

```
POST /order-details
```

Parameters

Type	Name	Description	Schema
Body	**order** *required*	order	OrderDetail

Responses

HTTP Code	Description	Schema
200	OK	OrderDetail
201	Created	No Content

HTTP Code	Description	Schema
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `*/*`

Tags

- order-detail-controller

Returns list of all Order details in the system.

```
GET /order-details
```

Responses

HTTP Code	Description	Schema
200	OK	< OrderDetail > array
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- `*/*`

Tags

- order-detail-controller

Returns a specific Order detail by their identifier. 404 if does not exist.

```
GET /order-details/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	OrderDetail
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- order-detail-controller

Updates Order detail.

```
PUT /order-details/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Type	Name	Description	Schema
Body	**order** *required*	order	OrderDetail

Responses

HTTP Code	Description	Schema
200	OK	OrderDetail
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `*/*`

Tags

- order-detail-controller

Deletes an Order detail from the system. 404 if the order detail's identifier is not found.

```
DELETE /order-details/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
204	No Content	No Content
401	Unauthorized	No Content
403	Forbidden	No Content

Produces

- `application/json`

Tags

- order-detail-controller

Returns Order of a specific Order detail by their identifier. 404 if does not exist.

```
GET /order/order-details/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
401	Unauthorized	No Content
403	Forbidden	No Content

HTTP Code	Description	Schema
404	Not Found	No Content

Produces

- **/**

Tags

- order-detail-controller

Creates a new Order.

```
POST /orders
```

Parameters

Type	Name	Description	Schema
Body	**order** *required*	order	Order

Responses

HTTP Code	Description	Schema
200	OK	Order
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- application/json

Produces

- */*

Tags

- order-controller

Returns list of all Orders in the system.

```
GET /orders
```

Responses

HTTP Code	Description	Schema
200	OK	< Order > array
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- order-controller

Returns list of Orders by Customer identifier. 404 if does not exist.

```
GET /orders/customer/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	< Order > array
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- order-controller

Returns number of orders per each day from this month.

```
POST /orders/monthly/{truth}
```

Parameters

Type	Name	Description	Schema
Path	**truth** *required*	truth	integer (int64)
Body	**ids** *required*	ids	StatisticsDTO

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
201	Created	No Content

HTTP Code	Description	Schema
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `*/*`

Tags

- order-controller

Returns list of Orders by Supplier identifier. 404 if does not exist.

```
GET /orders/supplier/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	< Order > array
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- order-controller

Returns number of orders per each day from this week.

```
POST /orders/weekly/{truth}
```

Parameters

Type	Name	Description	Schema
Path	**truth** *required*	truth	integer (int64)
Body	**ids** *required*	ids	StatisticsDTO

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- application/json

Produces

- */*

Tags

- order-controller

Returns a specific Order by their identifier. 404 if does not exist.

```
GET /orders/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	Order
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- order-controller

Updates Order.

```
PUT /orders/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)
Body	**order** *required*	order	Order

Responses

HTTP Code	Description	Schema
200	OK	Order
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `*/*`

Tags

- order-controller

Deletes an Order from the system. 404 if the order's identifier is not found.

```
DELETE /orders/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
204	No Content	No Content
401	Unauthorized	No Content
403	Forbidden	No Content

Produces

- `application/json`

Tags

- order-controller

Returns Payment of a specific Order detail by their identifier. 404 if does not exist.

```
GET /payment/order-details/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity

HTTP Code	Description	Schema
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- **/**

Tags

- order-detail-controller

Creates a new Payment.

```
POST /payments
```

Parameters

Type	Name	Description	Schema
Body	**payment** *required*	payment	Payment

Responses

HTTP Code	Description	Schema
200	OK	Payment
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `*/*`

Tags

- payment-controller

Returns list of all Payments in the system.

```
GET /payments
```

Responses

HTTP Code	Description	Schema
200	OK	< Payment > array
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- `*/*`

Tags

- payment-controller

Returns a specific Payment by their identifier. 404 if does not exist.

```
GET /payments/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	Payment
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- payment-controller

Updates Payment.

```
PUT /payments/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)
Body	**payment** *required*	payment	Payment

Responses

HTTP Code	Description	Schema
200	OK	Payment
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `*/*`

Tags

- payment-controller

Deletes a Payment from the system. 404 if the payment's identifier is not found.

```
DELETE /payments/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
204	No Content	No Content

HTTP Code	Description	Schema
401	Unauthorized	No Content
403	Forbidden	No Content

Produces

- `application/json`

Tags

- payment-controller

Returns Product of a specific Order detail by their identifier. 404 if does not exist.

```
GET /product/order-details/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- `*/*`

Tags

- order-detail-controller

Creates a new Product.

```
POST /products
```

Parameters

Type	Name	Description	Schema
Body	**product** *required*	product	Product

Responses

HTTP Code	Description	Schema
200	OK	Product
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- application/json

Produces

- */*

Tags

- product-controller

Returns list of all Products in the system.

```
GET /products
```

Responses

HTTP Code	Description	Schema
200	OK	< Product > array
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- product-controller

Returns a specific Product by their identifier. 404 if does not exist.

```
GET /products/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	Product
401	Unauthorized	No Content

HTTP Code	Description	Schema
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- product-controller

Updates Product.

```
PUT /products/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)
Body	**product** *required*	product	Product

Responses

HTTP Code	Description	Schema
200	OK	Product
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `*/*`

Tags

- product-controller

Deletes a Product from the system. 404 if the product's identifier is not found.

```
DELETE /products/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
204	No Content	No Content
401	Unauthorized	No Content
403	Forbidden	No Content

Produces

- `application/json`

Tags

- product-controller

Returns a Supplier by Order identifier. 404 if does not exist.

```
GET /supplier/order/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- order-controller

Creates a new Supplier.

```
POST /suppliers
```

Parameters

Type	Name	Description	Schema
Body	**supplier** *required*	supplier	Supplier

Responses

HTTP Code	Description	Schema
200	OK	Supplier
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `*/*`

Tags

- supplier-controller

Returns list of all Suppliers in the system.

```
GET /suppliers
```

Responses

HTTP Code	Description	Schema
200	OK	< Supplier > array
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- supplier-controller

Returns a specific Supplier by their identifier. 404 if does not exist.

```
GET /suppliers/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	Supplier
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Produces

- */*

Tags

- supplier-controller

Updates Supplier.

```
PUT /suppliers/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)
Body	**supplier** *required*	supplier	Supplier

Responses

HTTP Code	Description	Schema
200	OK	Supplier
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `*/*`

Tags

- supplier-controller

Deletes a Supplier from the system. 404 if the supplier's identifier is not found.

```
DELETE /suppliers/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
204	No Content	No Content
401	Unauthorized	No Content
403	Forbidden	No Content

Produces

- `application/json`

Tags

- supplier-controller

Updates Customer.

```
PUT /updateCustomer/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	id	integer (int64)
Body	**customer** *required*	customer	Customer

Responses

HTTP Code	Description	Schema
200	OK	ResponseEntity
201	Created	No Content
401	Unauthorized	No Content
403	Forbidden	No Content
404	Not Found	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- customer-controller

Definitions

Customer

Name	Schema
address *optional*	string
email *optional*	string
first_name *optional*	string
id *optional*	integer (int64)
last_name *optional*	string

Name	Schema
order *optional*	< Order > array
phone *optional*	string

Order

Name	Schema
customerId *optional*	Customer
dateOfOrder *optional*	string (date-time)
id *optional*	integer (int64)
orderDetail *optional*	< OrderDetail > array
orderDetails *optional*	< OrderDetail > array
status *optional*	string
supplierId *optional*	Supplier

OrderDetail

Name	Schema
date *optional*	string (date-time)
id *optional*	integer (int64)

Name	Schema
orderId *optional*	Order
paymentId *optional*	Payment
productId *optional*	Product
quantity *optional*	integer (int32)
size *optional*	integer (int32)
total *optional*	number (double)
unitPrice *optional*	number (double)

Payment

Name	Schema
id *optional*	integer (int64)
orderDetails *optional*	string
paymentType *optional*	string

Product

Name	Schema
description *optional*	string

Name	Schema
id *optional*	integer (int64)
name *optional*	string

ResponseEntity

Name	Schema
body *optional*	object

Name	Schema
statusCode *optional*	enum (100 CONTINUE, 101 SWITCHING_PROTOCOLS, 102 PROCESSING, 103 CHECKPOINT, 200 OK, 201 CREATED, 202 ACCEPTED, 203 NON_AUTHORITATIVE_INFORMATION, 204 NO_CONTENT, 205 RESET_CONTENT, 206 PARTIAL_CONTENT, 207 MULTI_STATUS, 208 ALREADY_REPORTED, 226 IM_USED, 300 MULTIPLE_CHOICES, 301 MOVED_PERMANENTLY, 302 FOUND, 302 MOVED_TEMPORARILY, 303 SEE_OTHER, 304 NOT_MODIFIED, 305 USE_PROXY, 307 TEMPORARY_REDIRECT, 308 PERMANENT_REDIRECT, 400 BAD_REQUEST, 401 UNAUTHORIZED, 402 PAYMENT_REQUIRED, 403 FORBIDDEN, 404 NOT_FOUND, 405 METHOD_NOT_ALLOWED, 406 NOT_ACCEPTABLE, 407 PROXY_AUTHENTICATION_REQUIRED, 408 REQUEST_TIMEOUT, 409 CONFLICT, 410 GONE, 411 LENGTH_REQUIRED, 412 PRECONDITION_FAILED, 413 PAYLOAD_TOO_LARGE, 413 REQUEST_ENTITY_TOO_LARGE, 414 URI_TOO_LONG, 414 REQUEST_URI_TOO_LONG, 415 UNSUPPORTED_MEDIA_TYPE, 416 REQUESTED_RANGE_NOT_SATISFIABLE, 417 EXPECTATION_FAILED, 418 I_AM_A_TEAPOT, 419 INSUFFICIENT_SPACE_ON_RESOURCE, 420 METHOD_FAILURE, 421 DESTINATION_LOCKED, 422 UNPROCESSABLE_ENTITY, 423 LOCKED, 424 FAILED_DEPENDENCY, 425 TOO_EARLY, 426 UPGRADE_REQUIRED, 428 PRECONDITION_REQUIRED, 429 TOO_MANY_REQUESTS, 431 REQUEST_HEADER_FIELDS_TOO_LARGE, 451 UNAVAILABLE_FOR_LEGAL_REASONS, 500 INTERNAL_SERVER_ERROR, 501 NOT_IMPLEMENTED, 502 BAD_GATEWAY, 503 SERVICE_UNAVAILABLE, 504 GATEWAY_TIMEOUT, 505 HTTP_VERSION_NOT_SUPPORTED, 506 VARIANT_ALSO_NEGOTIATES, 507 INSUFFICIENT_STORAGE, 508 LOOP_DETECTED, 509 BANDWIDTH_LIMIT_EXCEEDED, 510 NOT_EXTENDED, 511 NETWORK_AUTHENTICATION_REQUIRED)
statusCodeValue *optional*	integer (int32)

StatisticsDTO

Name	Schema
listElements *optional*	< integer (int32) > array

Supplier

Name	Schema
email *optional*	string
id *optional*	integer (int64)
name *optional*	string
order *optional*	< Order > array
phone *optional*	string