Endpoint

- Add a person to the mailing list

```
POST: http://localhost:8080/mail
Body:
{
    "email":string
}
Content type: application/json
Responses:
Content type: application/json
HTTP Code:200
{
    "code": 1,
    "data": {
        "createDate": 1555035322691,
        "email": "1"
    },
    "message": "SUCCESS"
}
HTTP Code 405: Validation exception
{
  "code": 0,
  "message": "Email address cannot be empty!"
}
HTTP Code 403: business exception
{
  "code": 0,
  "message": "Email address already exists!"
}
HTTP Code 500: other exception
{
  "code": 0,
  "message": the error message
}
```

- Get all current products

```
GET: http://localhost:8080/product/getAllCurrent
Responses:
Content type: application/json
HTTP Code:200
{
  "code": 1,
  "data": [
      {
          "createDate": 1555037640000,
          "description": "a smartphone",
          "id": 1,
          "isDelete": false,
          "isRelease": true,
          "name": "iphone8",
          "updateDate": 1555037645000
      },
      {
          "createDate": 1555037671000,
          "description": "a smartphone",
          "id": 2,
          "isDelete": false,
          "isRelease": true,
          "name": "iphone7",
          "updateDate": 1555037676000
      }
   ],
   "message": "SUCCESS"
}
HTTP Code: 500 other exception
{
   "code": 0,
   "message": error message
}
```

- Get soon to be released products

```
GET: http://localhost:8080/product/getComingSoon
Responses:
Content type: application/json
HTTP Code:200
{
    "code": 1,
    "data": [
        {
            "createDate": 1555046671000,
            "description": "a new shape",
            "id": 16,
            "isDelete": false,
            "isRelease": false,
            "name": "iphoneXI",
            "updateDate": 1555046676000
        }
    ],
    "message": "SUCCESS"
}
HTTP Code: 500 other exception
{
    "code": 0,
    "message": error message
}
```

- Remove a person from the mailing list

```
DELETE: http://localhost:8080/mail/{email}
Responses:
Content type: application/json
HTTP Code:200
{
    "code": 1,
    "message": "SUCCESS"
}
HTTP Code: 403 business exception
{
    "code": 0,
    "message": "Email address does not exist!"
}
HTTP Code: 500 other exception
{
    "code": 0,
    "message": error message
}
```

- add a current product

```
POST: http://localhost:8080/product/current
Body:
{
    "name": string,
    "description": string
}
Content type: application/json
Responses:
Content type: application/json
HTTP Code:200
{
    "code": 1,
    "data": {
        "createDate": 1555150130685,
        "description": "a smartphone",
        "id": 6,
        "isDelete": false,
        "isRelease": true,
        "name": "iphone6",
        "updateDate": 1555150130685
    },
    "message": "SUCCESS"
}
HTTP Code: 403 business exception
{
    "code": 0,
    "message": "Product name already exists!"
}
HTTP Code: 405 Validation exception
{
    "code": 0,
    "message": "Product name is empty!"
}
HTTP Code: 500 other exception
{
    "code": 0,
    "message": error message
}
```

- add a soon to be released product

```
POST: http://localhost:8080/product/comingSoon
Body:
{
    "name": string,
    "description": string
}
Content type: application/json
Responses:
Content type: application/json
HTTP Code:200
{
    "code": 1,
    "data": {
        "createDate": 1555150130685,
        "description": "a new smartphone",
        "id": 7,
        "isDelete": false,
        "isRelease": false,
        "name": "iphoneXI",
        "updateDate": 1555150130685
    },
    "message": "SUCCESS"
}
HTTP Code: 403 business exception
{
    "code": 0,
    "message": "Product name already exists!"
}
HTTP Code: 405 Validation exception
{
    "code": 0,
    "message": "Product name is empty!"
}
HTTP Code: 500 other exception
{
    "code": 0,
    "message": error message
}
```

- remove a current product

```
DELETE: http://localhost:8080/product/current/{productId}
Responses:
Content type: application/json
HTTP Code:200
{
    "code": 1,
    "message": "SUCCESS"
}
HTTP Code: 400 productId invalid!
{
    "code": 0,
    "message": string
}
HTTP Code: 403 business exception
{
    "code": 0,
    "message": "Product does not exist or is not released!"
}
HTTP Code: 405 Validation exception
```

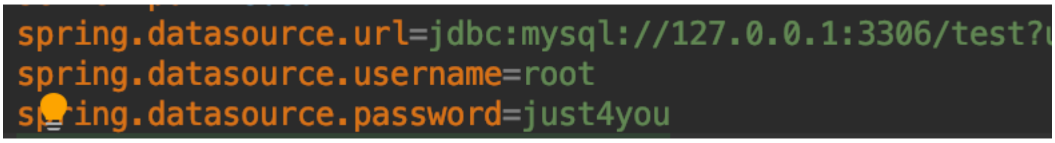
```
{
   "code": 0,
   "message": "Product id cannot be empty and should be greater
than 0!"
}
HTTP Code: 500 other exception
{
   "code": 0,
   "message": error message
}
```

- remove a soon to be released product

```
DELETE: http://localhost:8080/product/comingSoon/{productId}
Responses:
Content type: application/json
Code:200
{
   "code": 1,
   "message": "SUCCESS"
}
Code: 400 productId invalid!
{
   "code": 0,
   "message": string
}
Code: 403 business exception
{
   "code": 0,
   "message": "Product does not exist or is released!"
}
Code: 405 Validation exception
{
   "code": 0,
   "message": "Product id cannot be empty and should be greater
than 0!"
}
Code: 500 other exception
{
   "code": 0,
   "message": error message
}
```

Database

The database information is as follows.



```
spring.datasource.url=jdbc:mysql://127.0.0.1:3306/test?u
spring.datasource.username=root
spring.datasource.password=just4you
```

If you need to change it, please modify the application.properties file by yourself and then re-execute the “mvn package” .

Table statement

```
SET NAMES utf8mb4;
SET FOREIGN_KEY_CHECKS = 0;

-- ----------------------------
-- Table structure for maillist
-- ----------------------------
DROP TABLE IF EXISTS `maillist`;
CREATE TABLE `maillist` (
  `id` int(11) NOT NULL AUTO_INCREMENT COMMENT 'PK',
  `email` varchar(100) NOT NULL COMMENT 'Mail address',
  `create_date` datetime DEFAULT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `idx_email` (`email`) USING BTREE
) ENGINE=InnoDB AUTO_INCREMENT=8 DEFAULT CHARSET=utf8mb4;

-- ----------------------------
-- Records of maillist
-- ----------------------------
BEGIN;
INSERT INTO `maillist` VALUES (3, 'jock@email.com', '2019-04-12 11:06:51');
INSERT INTO `maillist` VALUES (5, 'jodck@s.cd', '2019-04-12 09:19:39');
COMMIT;

-- ----------------------------
-- Table structure for products
-- ----------------------------
DROP TABLE IF EXISTS `products`;
CREATE TABLE `products` (
  `id` int(11) NOT NULL AUTO_INCREMENT COMMENT 'PK',
  `name` varchar(100) NOT NULL COMMENT 'Product name',
  `description` varchar(255) DEFAULT NULL COMMENT 'The description of product',
  `is_release` bit(1) DEFAULT NULL COMMENT '1: released\n0: not release',
  `create_date` datetime DEFAULT NULL,
  `update_date` datetime DEFAULT NULL,
  `is_delete` bit(1) DEFAULT NULL COMMENT '1: deleted\n',
  PRIMARY KEY (`id`),
  UNIQUE KEY `idx_products_name` (`name`) USING BTREE
) ENGINE=InnoDB AUTO_INCREMENT=17 DEFAULT CHARSET=utf8mb4;

-- ----------------------------
-- Records of products
-- ----------------------------
BEGIN;
INSERT INTO `products` VALUES (1, 'iphone8', 'a smartphone', b'1', '2019-04-12 10:54:00', '2019-04-12 10:54:05', b'0');
INSERT INTO `products` VALUES (2, 'iphone7', 'a smartphone', b'1', '2019-04-12 10:54:31', '2019-04-12 10:54:36', b'0');
INSERT INTO `products` VALUES (16, 'iphoneXI', 'a new shape', b'0', '2019-04-12 13:24:31', '2019-04-12 13:24:36', b'0');
COMMIT;

SET FOREIGN_KEY_CHECKS = 1;
```

execute

```
java -jar target/demo-0.0.1-SNAPSHOT.jar
```