

# Тестовое задание

## Описание тестируемых методов метода

### 1. Метод добавления товара.

Метод используется для добавления товара с параметрами:  
идентификатор, название, цена

### Примеры данных:

Запрос

```
{"id": 76699046, "name": "apple", "price": 12.10}
```

Ответ системы в случае успешного добавления товара

```
{"code": 200, "description": "success"}
```

Произошла ошибка во время добавления товара

```
{"code": 500, "description": "Internal Server Error"}
```

В запросе отсутствуют обязательные параметры

```
{"code": 400, "description": "Bad Request"}
```

### 2. Метод получения данных о товаре по

## идентификатору.

Метод получает идентификатор и возвращает данные о соответствующем товаре: идентификатор, название товара, цена товара.

## Примеры данных

Запрос

```
{"id": 76699046}
```

Ответ метода в случае успешного получения товара

```
{"id": 76699046, "name": "apple", "price": 12.10}
```

Товар не найден

```
{"code": 404, "description": "Not Found"}
```

В запросе отсутствуют обязательные параметры

```
{"code": 400, "description": "Bad Request"}
```

все ответы метода выполняются в рамках HTTP 200

## Задание №1 - Создание тест кейсов

- Создайте набор тест кейсов для методов добавления и получения товаров в [TestRail](#) или [Google таблицах](#)

- Добавьте тестовые данные на основе выбранной техники тест дизайна
- Выберите тест кейсы для автоматизации

## Задание №2 - Реализация stub

Реализуйте stub, имитирующий следующие ответы метода

- Ответ метода в случае успешного получения товара
- Товар не найден
- В запросе отсутствуют обязательные параметры
- Произошла ошибка во время добавления товара

## Задание №3 - Автоматизация тестирования

- Реализуйте автотесты метода создания и получения товаров, используя Java. Опишите инфраструктуру.
- Выложите stub и тесты в GitHub репозиторий, опишите инструкцию по запуску, пришлите ссылку на репозиторий.

## Схемы

Идентификатор товара

```
{  
  "title": "Id",  
  "type": "object",  
  "properties": {  
    "id": {  
      "description": "The unique identifier for a object",  
      "type": "integer"  
    }  
  }  
}
```

```
    }  
  },  
  "required": [  
    "id"  
  ]  
}
```

## Товар

```
{  
  "title": "Product",  
  "type": "object",  
  "properties": {  
    "id": {  
      "description": "The unique identifier for a product",  
      "type": "integer"  
    },  
    "name": {  
      "description": "Name of the product",  
      "type": "string"  
    },  
    "price": {  
      "type": "number",  
      "exclusiveMinimum": 0  
    }  
  },  
  "required": [  

```

```
    "id",  
    "name",  
    "price"  
  ]  
}
```

## Ошибка

```
{  
  "title": "Error",  
  "type": "object",  
  "properties": {  
    "code": {  
      "type": "integer"  
    },  
    "description": {  
      "type": "string"  
    }  
  }  
}
```