Assignment 3- Retrieve Revenue by OrderType & LocationId

In this assigment, lets refactor the @GetMapping("/revenue/{order_type}") endpoint in the controller to accept the locationId as a request parameter and retrieve the revenue for that location id.

Assignment Instructions:
1. Update the controller function revenueByOrderType () to accept the location_id as a request parameter.
2. Create a nee function in the named getRevenueByLocationId() in the OrderService to accept the locationId and orderType as a input.
    a. The return type will be OrderRevenueDTO.
3. Retrieve the revenue for the passed LocationId and Return it in the OrderRevenueDTO as a return type

Questions for this assignment
1) Make changes in the OrdersController to accept the locationId for the revenueByOrderType function.

2) Make Changes in the OrderService to implement the getRevenueByLocationId() function.

Instructor's Example:
1) Make changes in the OrdersController to accept the locationId for the revenueByOrderType function.

2) Make Changes in the OrderService to implement the getRevenueByLocationId() function.

OrderService:
```
public OrderRevenueDTO getRevenueByLocationId(String orderType, String locationId)
{
    var revenueStoreByType =getRevenueStore(orderType);

    var totalRevenue = revenueStoreByType.get(locationId);
    if (totalRevenue != null) {
        return new OrderRevenueDTO(locationId,mapOrderType(orderType),
totalRevenue);
    } else {
        return null;
    }
}
```

OrderController:
```
@GetMapping("/revenue/{order_type}")
public ResponseEntity<?> revenueByOrderType(
        @PathVariable("order_type") String orderType,
        @RequestParam(value = "location_id", required = false) String locationId
) {
```

```
    if (StringUtils.hasLength(locationId)) {
        return ResponseEntity.ok(orderService.getRevenueByLocationId(orderType,
locationId));
    } else {

        return ResponseEntity.ok(orderService.revenueByOrderType(orderType));

    }
}
```