49. Aggregate ашиглан категорийн номын дундаж үнийг тооцоолох Mongoose static функц болон post save middleware ашиглах

Models-book.js

BookSchema.statics.computeCategoryAveragePrice = async function (catId) {

const obj = await this.aggregate([

{ $match: { category: catId } },

{ $group: { \_id: "$category", avgPrice: { $avg: "$price" } } },

]);

console.log(obj);

await this.model("Category").findByIdAndUpdate(catId, {

averagePrice: obj[0].avgPrice,

});

return obj;

};

BookSchema.post("save", function () {

this.constructor.computeCategoryAveragePrice(this.category);

});

BookSchema.pre("remove", function () {

this.constructor.computeCategoryAveragePrice(this.category);

});

Models-category.js өмнө random-оор үүсгэж өгч байсан.

// this.averagePrice = Math.floor(Math.random() \* 100000) + 3000;

Нэг category-ын дундаж үнэ харах туршиж үзий улаанаар бичсэн дотор н оруулж бичиж өгсөн.

Controller – book.js

exports.getBook = asyncHandler(async (req, res, next) => {

const book = await Book.findById(req.params.id);

if (!book) {

throw new MyError(req.params.id + " ID-тай ном байхгүй байна.", 404);

}

const avg = await Book.computeCategoryAveragePrice(book.category);

res.status(200).json({

success: true,

data: book,

dundaj: avg, // 49 туршилт дундаж үнэ дуудаж байна.

});

});