**Exercise03\_02\_01 – Step 1**

fluxGen.js

const output = [];

------------------------------------

fluxGen.js

const output = [];  
 let current = seed;

------------------------------------

fluxGen.js

for (var i = 0; i < times; i++) {

let change = (Math.random() \* variability).toFixed(0);

**Exercise03\_02\_01 – Step 2**

handlers/pizza.js

this.startingDate = startingDate;

[

this.ticker,

this.name,

this.startingQuote

] = pizzaProps;

------------------------------------

pizza.js

this.startingQuote,

this.variability = getRand(),

this.positivity = getRand()

] = pizzaProps;

**Exercise03\_02\_01 – Step 3**

handlers/pizza.js

const { ticker } = request.params;

------------------------------------

models/pizza.js

{ startingDate: curDate } = this;

**Exercise03\_02\_01 – Step 4**

popGen.js

module.exports = {

getPopularSlices,

getMostPopular,

getNewestSlice,

getMostImproved

};

------------------------------------

dataStore.js

module.exports = {

init,

getQuotes,

getAllQuotes,

getPizzas,

getPizza

};

**Exercise03\_02\_01 – Step 5**

market.js

module.exports = {

run (socket) {

runInterval = setInterval(function () {

quoteManager.updateQuotes(function (err, newData) {

socket.emit('new\_data', JSON.stringify(newData));

});

}, 1000);

},

stop () {

clearInterval(runInterval);

}

};

------------------------------------

quoteManager.js

module.exports = {

updateQuotes (callback) {

**Exercise03\_02\_01 – Step 6**

server.js

server.start(err => {

if (err) throw err;

console.log(`Connected on ${server.info.uri}`);

});

------------------------------------

quoteManager.js

console.log(`${newData} updating quotes`);

------------------------------------

quoteManager.js

console.log(`${JSON.stringify(newData)}