**BÀI TẬP CÓ ĐÁNH GIÁ KẾT QUẢ**

Mã sinh viên: 1871020089

Họ và tên sinh viên: Lê Trọng Chiến

Lớp: CNTT 18-05

**Câu 1: Duyệt và thao tác trên mảng (3 điểm)**

Công ty bạn vừa nhận được danh sách khách hàng cần xử lý, danh sách này được lưu dưới dạng mảng như sau:

const customers = [

{ name: 'Alice', orders: [120, 250, 320] },

{ name: 'Bob', orders: [50, 100, 40] },

{ name: 'Charlie', orders: [400, 500] },

];

**Yêu cầu:**

1. Viết một đoạn mã JavaScript để tính tổng giá trị đơn hàng của từng khách hàng và lưu kết quả vào một mảng mới. Kết quả mảng mới có dạng:

[

{ name: 'Alice', totalOrders: 690 },

{ name: 'Bob', totalOrders: 190 },

{ name: 'Charlie', totalOrders: 900 },

]

1. Sắp xếp mảng mới theo tổng giá trị đơn hàng (totalOrders) từ cao đến thấp.

**LỜI GIẢI CÓ GIẢI THÍCH CHI TIẾT Ở ĐÂY:**

|  |
| --- |
| //1 Tạo mảng mới chứa tên và tổng đơn hàng của mỗi khách hàng  //Dùng map() để tạo mảng mới  const TongDonHang = customers.map(customer => ({      name: customer.name,      //Dùng reduce() để tính tổng các đơn hàng      totalOrders: customer.orders.reduce((Tong, DonHang) => Tong + DonHang, 0)  }));  //2 Sắp xếp giảm dần theo totalOrders  //Dùng sort() để sắp xếp  const DanhSachDaXep = TongDonHang.sort((a, b) => b.totalOrders - a.totalOrders);  //In ra kết quả  console.log(DanhSachDaXep); |

**Câu 2: Xử lý dữ liệu đối tượng (3 điểm)**

Bạn được cung cấp một đối tượng lưu thông tin về số lượng sách của một thư viện như sau:

const library = {

Science: { total: 120, borrowed: 30 },

Literature: { total: 80, borrowed: 25 },

Math: { total: 100, borrowed: 10 },

History: { total: 90, borrowed: 50 },

};

**Yêu cầu:**

1. Viết đoạn mã để tính tổng số sách hiện có sẵn trong thư viện (không bao gồm sách đã được mượn).
2. Tìm ra loại sách có tỷ lệ mượn cao nhất (tính bằng borrowed / total). Kết quả trả về là tên của loại sách và tỷ lệ mượn.

**LỜI GIẢI CÓ GIẢI THÍCH CHI TIẾT Ở ĐÂY:**

|  |
| --- |
| // 1. Tính tổng số sách có sẵn  const SachKhaDung = Object.values(library).reduce((Tong, DanhMuc) =>      Tong + (DanhMuc.total - DanhMuc.borrowed), 0);  // 2. Tìm loại sách có tỷ lệ mượn cao nhất  let TyLeCaoNhat = 0;  let DanhMucCaoNhat = '';  //Dùng Object.entries() và forEach() để duyệt qua từng phần tử của mảng  Object.entries(library).forEach(([DanhMuc, DuLieu]) => {      //Tính tỷ lệ mượn      const TyLe = DuLieu.borrowed / DuLieu.total;      //Nếu tỷ lệ mượn lớn hơn TyLeCaoNhat thì gán TyLeCaoNhat = TyLe và DanhMucCaoNhat = DanhMuc      if (TyLe > TyLeCaoNhat) {          TyLeCaoNhat = TyLe;          DanhMucCaoNhat = DanhMuc;      }  });  //In ra kết quả  console.log(`Số sách có sẵn: ${SachKhaDung}`);  console.log(`Loại sách có tỷ lệ mượn cao nhất: ${DanhMucCaoNhat} (${(TyLeCaoNhat \* 100).toFixed(2)}%)`); |

**Câu 3: Kết hợp mảng và đối tượng (4 điểm)**

Một nhà phát triển đã viết một đoạn mã để lưu trữ danh sách sinh viên và điểm số của họ dưới dạng mảng như sau:

const students = [

{ name: 'Anna', scores: { math: 85, english: 78, science: 92 } },

{ name: 'Brian', scores: { math: 90, english: 88, science: 76 } },

{ name: 'Clara', scores: { math: 60, english: 70, science: 80 } },

];

**Yêu cầu:**

1. Viết đoạn mã để tính điểm trung bình của mỗi sinh viên và lưu vào đối tượng tương ứng dưới dạng thuộc tính mới averageScore. Kết quả mảng sau khi xử lý:

[

{ name: 'Anna', scores: { math: 85, english: 78, science: 92 }, averageScore: 85 },

{ name: 'Brian', scores: { math: 90, english: 88, science: 76 }, averageScore: 84.67 },

{ name: 'Clara', scores: { math: 60, english: 70, science: 80 }, averageScore: 70 },

]

1. **Tìm sinh viên có điểm trung bình cao nhất và in ra kết quả dưới dạng:**

"Sinh viên có điểm trung bình cao nhất là: Brian (84.67)"

**LỜI GIẢI CÓ GIẢI THÍCH CHI TIẾT Ở ĐÂY:**

|  |
| --- |
| // 1. Tính điểm trung bình cho mỗi sinh viên  //Dùng map() để tính điểm trung bình cho mỗi sinh viên  const DanhSachDiemTB = students.map(student => {      const DiemSo = Object.values(student.scores); // Lấy tất cả điểm số      const DiemTB = DiemSo.reduce((Tong, Diem) => Tong + Diem, 0) / DiemSo.length; // Tính trung bình      return {          ...student, // Giữ lại thông tin cũ          averageScore: Number(DiemTB.toFixed(2)) // Làm tròn 2 chữ số thập phân      };  });  // 2. Tìm sinh viên có điểm trung bình cao nhất  //Dùng reduce() để tìm sinh viên có điểm trung bình cao nhất  const SinhVienXuatSac = DanhSachDiemTB.reduce((CaoNhat, HienTai) =>      HienTai.averageScore > CaoNhat.averageScore ? HienTai : CaoNhat  );  //In ra kết quả  console.log(DanhSachDiemTB);  console.log(`Sinh viên có điểm trung bình cao nhất là: ${SinhVienXuatSac.name} (${SinhVienXuatSac.averageScore})`); |

**Yêu cầu về bài làm:**

1. Giải thích ngắn gọn logic của bạn cho mỗi bài.
2. Viết mã một cách rõ ràng, có chú thích giải thích.
3. Bài làm cần đảm bảo tính thực tế và thể hiện được khả năng tư duy logic, xử lý dữ liệu.