

JavaScript Conditional Statement Exercises

Level 1

1. Viết hàm `isOddNumber(number)` nhận một giá trị đầu vào `number` là một số nguyên bất kỳ, kiểm tra và trả về kết quả số đó có phải là số lẻ (`true`) hay không (`false`)
2. Viết hàm `maxOfTwo(a, b)` nhận hai giá trị đầu vào `a, b` là 2 số bất kỳ, kiểm tra và trả về số lớn hơn

3. Viết hàm `divisibleByThreeAndFive(number)` nhận một giá trị đầu vào `number` là một số nguyên bất kỳ, kiểm tra và trả về kết quả cho biết số đó có chia hết cho cả 3 và 5 (`true`) hay không (`false`)
4. Viết hàm `season(month)` nhận một giá trị đầu vào `month` là một tháng trong năm (`number`), kiểm tra và in ra mùa tương ứng với tháng đó:
 - Winter (12, 1, 2)
 - Spring (3, 4, 5)
 - Summer (6, 7, 8)
 - Fall (9, 10, 11)
5. Viết hàm `trafficLight(light)` nhận một giá trị đầu vào `light` là tín hiệu đèn giao thông (Red, Yellow, Green), in ra thông báo tương ứng với tín hiệu đèn:
 - Green: Được phép di chuyển
 - Yellow: Giảm tốc độ và dừng lại
 - Red: Dừng lại
6. Viết hàm `calcCommissions(totalSales)` nhận một giá trị đầu vào `totalSales` là tổng doanh số bán hàng của một đại lý, kiểm tra và **trả về kết quả** là tiền hoa hồng mà đại lý nhận được quy định như sau:
 - Nếu $totalSales \leq 100\text{triệu}$ thì hoa hồng nhận được là 5%
 - Nếu $100\text{triệu} < totalSales \leq 300\text{triệu}$ thì hoa hồng là 10%
 - Nếu $totalSales > 300\text{triệu}$ thì hoa hồng là 20%

Level 2

- Viết hàm `isLeafYear(year)` nhận một giá trị đầu vào `year` là một năm bất kỳ, kiểm tra và trả về kết quả năm đó có phải là năm nhuận (`true`) hay không (`false`)



Tính theo lịch [Gregory](#)

- Viết hàm `daysOfMonth(month, year)` nhận một giá trị đầu vào `month`, trả về số ngày của tháng đó:
 - Các tháng 1, 3, 5, 7, 8, 10, 12 có 31 ngày
 - Các tháng 4, 6, 9, 11 có 30 ngày
 - Tháng 2 có 29 ngày nếu là năm nhuận, nếu không là 28 ngày
- Viết hàm `calcGrade(point)` nhận một giá trị đầu vào `point` là điểm trung bình của sinh viên, kiểm tra và **trả về kết quả** xếp loại tốt nghiệp của sinh viên theo số điểm trung bình (`point`). Quy định như sau:
 - $point < 4.0$: Xếp loại F
 - $4.0 \leq point < 5.5$: Xếp loại D
 - $5.5 \leq point < 7.0$: Xếp loại C
 - $7.0 \leq point < 8.5$: Xếp loại B
 - $point \geq 8.5$: Xếp loại A
- Viết hàm `calcTaxiFee(km)` nhận một giá trị đầu vào `km` là số km đã di, tính và **trả về kết quả** là số tiền phải trả cho khách đi taxi theo số kilomet (km) đã di chuyển. Quy định như sau:
 - Phí mở cửa 10.000
 - Giá cho 30km đầu tiên là 11.000đ/km
 - Giá cho km thứ 31 trở đi là 9.500đ/km
 - Tổng tiền = Phí mở cửa + Tổng tiền trên số km
- Viết hàm `maxOfThree(a, b, c)` nhận ba giá trị đầu vào `a, b, c` là 3 số bất kỳ. Kiểm tra và in ra kết quả số lớn nhất trong 3 số (lưu ý trường hợp các số có giá trị bằng nhau)
- Viết hàm `solveEquation(a, b, c)` nhận ba giá trị đầu vào `a, b, c` tương ứng là 3 tham số của phương trình bậc 2 $ax^2 + bx + c = 0$, tìm và trả về nghiệm của phương trình:
 - Nếu $a = 0$ và $b = 0$ và $c = 0$ thì phương trình có vô số nghiệm (Infinity)
 - Nếu $a = 0$ và $b = 0$ thì phương trình vô nghiệm (trả về `null`)
 - Nếu $a = 0$ thì phương trình có một nghiệm $-\frac{c}{b}$
 - Nếu $a \neq 0$ thì tính $delta = b^2 - 4ac$
 - Nếu $delta < 0$ thì phương trình vô nghiệm (trả về `null`)
 - Nếu $delta \geq 0$ thì phương trình có 2 nghiệm: $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$
 - Lưu ý kết quả trả về nếu có nghiệm là một mảng chứa các nghiệm của phương trình, cú pháp `[x1, x2]`