1) Для практикума из занятия 7 продумать, где можно применить замыкания

Замыкание можно было бы применить для переменной score создав для нее отдельную функцию counter() со своим окружением например let count = 0, которая возвращала бы функцию которая использует эту переменную count.

И еще можно замкнуть функции:  
isSnakeUnit(unit)  
isBarrier(newUnit)  
haveFood(unit)

внутри функции moveSnake(), т.е. просто объявив их в теле функции moveSnake() {....}

Не знаю стоит так делать, но можно вообще весь код игры поместить в отдельную функцию mySnake() которая через return возвращает доп.функцию runSnake() со сценарием вызовов функций игры, и просто потом поместить runSnake() со всем своим окружением в init(), тем самым мы замкнем все наши переменные и функции внутри mySnake().

2) Не выполняя кода, ответить, что выведет браузер и почему

if (!("a" in window)) {

var a = 1;

}

alert(a); // undefined

Так как после завершения оператора if() c ним из памяти исчезает и его цепочка видимости.

var b = function a(x) {

x && a(--x);

};

alert(a); // a – undefined

так как var b = function a(x) - это функциональное выражение, объект function a(x) стирается из памяти стразу после возвращения значения в переменную b, дальнейший вызов a приведет к ошибке a is not defined

function a(x) {

return x \* 2;

}

var a;

alert(a); // выведет в консоль текстовое представление function a(x) {……..}

function b(x, y, a) {

arguments[2] = 10;

alert(a);

}

b(1, 2, 3);

выведет число 10

через объект/псевдо-массив arguments обращаемся к переданным параметрам функции   
через св-во/псевдо-индекс.

function a() {

alert(this);

}

a.call(null); - object window   
не понимаю ведь контекст вызова объект null, почему object window?

3) \* Подумать о создании и развитии своего собственного сайта/сервиса

в работе.