**Controller JS**

**المدخلات الاساسية**

var MemoryGame = {

اوبجيكت لقيم الاعمدة و الصفوف

  settings: {

    rows: 2,

    columns: 3,

    images: 3, // عدد الصور

  },

  // Properties that indicate state

  cards: [], // مصفوفة فيها قيم الكروت

  attempts: 0, // عدد الكروت اللي اتقلبت

  mistakes: 0, // كام مرة غلط

  isGameOver: false,

**فاكشن جوا الاوبجيكت بتحتاج عدد الصفوف و الاعمدة و الصور**

initialize : function(rows, columns, images) {

    var validOptions = true;

    // التأكد من القيم

    if (!(typeof columns === 'number' && (columns % 1) === 0 && columns > 1) ||

        !(typeof rows === 'number' && (rows % 1) === 0) && rows > 1) {

      validOptions = false;

      throw {

        name: "invalidInteger", //إن القيم مش اقل من صفر

        message: "Both rows and columns need to be integers greater than 1."

      };

    }

    if ((columns \* rows) % 2 !== 0) {

      validOptions = false;

      throw {

        name: "oddNumber", //إن القيم باقي قسمة ضربهم تقبل القسمة علي اتنين

        message: "Either rows or columns needs to be an even number."

      };

    }

//تصفير القيم الاساسية و ابدأ فاكشنز الكروت و لخبطتها

    if (validOptions) {

      this.settings.rows = rows;

      this.settings.columns = columns;

      this.settings.images = images;

      this.attempts = 0;

      this.mistakes = 0;

      this.isGameOver = false;

      this.createCards().shuffleCards();

    }

**فاكشن بتعمل الكروت بالترتيب قبل ما نلخبطها**

 createCards: function() {

    var cards = [];

    var values = [];

    var count = 0;

    var maxValue = (this.settings.columns \* this.settings.rows) / 2;

// القيمة بتاعت الصفوف و الاعمدة مقسومة علي اتنين علشان الكروت بتتكرر

    while (count < maxValue) {

      // رقم كارت عشوائي

      var value = this.getRandomCardValue(values);

      // اول كارت

      cards[2 \* count] = new this.Card(value); //نسجل الداتا اللي هتطلع في كونستراكتور عشان هنستخدمها بعد كده

      // تاني كارت (له نفس القيمة)

      cards[2 \* count + 1] = new this.Card(value, true);

      count++;

    }

    this.cards = cards;

    return this; // بترد القيم في الجلوبال سكوب

**الجوريزم بيجيب ارقام عشوائية للفانكشن اللي فوق**

getRandomCardValue: function(values) {

    var valid = false;

    var randomValue = 0;

    while (!valid) {

      randomValue = Math.floor(Math.random() \* this.settings.images) + 1;

      var found = false;

      for (var index = 0; index < values.length; index++) {

        if (randomValue === values[index]) {

          found = true;

          break;

        }

      }

      if (!found) {

        valid = true;

        values.push(randomValue); // بيزود القيم في الاخر بتاعت الراندوم في الفاليوز

      }

    }

  return randomValue;

  },

**الجوريزم بيعمل لخبطة للكروت:**

shuffleCards: function() {

    var cards = this.cards;

    var shuffledCards = [];

    var randomIndex = 0;

    // لوبة بتلخبط الكروت

    while (shuffledCards.length < cards.length) {

      // عدد عشوائي بين صفر و عدد الكروت

      randomIndex  = Math.floor(Math.random() \* cards.length);

      // لو العنصر مش فاضي زود عليه

      if(cards[randomIndex]) {

        // زود عنصر جديد للمصفوفة

        shuffledCards.push(cards[randomIndex]);

        // و ساوي العناصر بفولس بحيث مانسجلش جواه حاجه تاني

        cards[randomIndex] = false;

      }

    }

    this.cards = shuffledCards;

    return this;

  },

**اللوجيك بتاع اللعبة**

play: (function() {

    var cardSelection = [];

    var revealedCards = 0;

    var revealedValues = [];

    return function(index) {

      var status = {};

      var value = this.cards[index].value;

      if (!this.cards[index].isRevealed) {

        this.cards[index].reveal();

// خلي قيمة الكارت كأنها ظاهرة

        cardSelection.push(index);

// سجل قيمة الكارت في الكروت اللي اشتغلت

        if (cardSelection.length == 2) {

          this.attempts++;

          if (this.cards[cardSelection[0]].value !=

              this.cards[cardSelection[1]].value) {

            // لو الكروت قيمة العناصر اللي اخترناها مش شبه بعض يبقي مش ماتش

            this.cards[cardSelection[0]].conceal();

            this.cards[cardSelection[1]].conceal();

// شغل الفكشن اللي في الموديل اللي بتخلي الكارت مقلوب تاني

            /\*\*

             \* الجوريزم بيحدد كام مرة عملنا غلطة و بيرد الناتج ب-1

             \*/

            var isMistake = false;

            if (revealedValues.indexOf(this.cards[cardSelection[0]].value) === -1) {

              revealedValues.push(this.cards[cardSelection[0]].value);

            }

            else {

              isMistake = true;

            }

            if (revealedValues.indexOf(this.cards[cardSelection[1]].value) === -1) {

              revealedValues.push(this.cards[cardSelection[1]].value);

            }

            if (isMistake) {

              this.mistakes++;

            }

// عدد الاخطاء بيتسجل حسب ما الاتنين يكونوا مش شبه بعض

            revealedValues.push(this.cards[cardSelection[0]].value);

// قيمة الكروت اللي اتقلبت بتتسجل في المصفوفة دي

            status.code = 3,

            status.message = 'No Match. Conceal cards.';

            status.args = cardSelection;

// كود 3 معمول للكود مش ماتشيد

          }

          else {

            revealedCards += 2;

            if (revealedCards == this.cards.length) {

              // لو عدد الكروت الظاهرة بيساوي عدد الكروت يبقي اللعبة خلصت

              this.isGameOver = true;

              revealedCards = 0;

              revealedValues = [];

              status.code = 4,

              status.message = 'GAME OVER! Attempts: ' + this.attempts +

                  ', Mistakes: ' + this.mistakes;

            }

// كود 4 بيظهر لما اللعبة تخلص و بيصفر كل حاجه و بيعرض الميستيكس

else {

              status.code = 2,

              status.message = 'Match.';

            }

// لو مش اي احتمال من اللي فوق يبقي القيمتين شبه بعض يبقي يا معلم تقوله الكود ب2 و ترد بماتش

          }

          cardSelection = [];

        }

        else {

          status.code = 1,

          status.message = 'Flip first card.';

// في حالة الضغط علي كارت واحد بس

        }

      }

      else {

        status.code = 0,

        status.message = 'Card is already facing up.';

// لو الكارت اولريدي مرفوع

      }

      return status;

    };

  })()

};