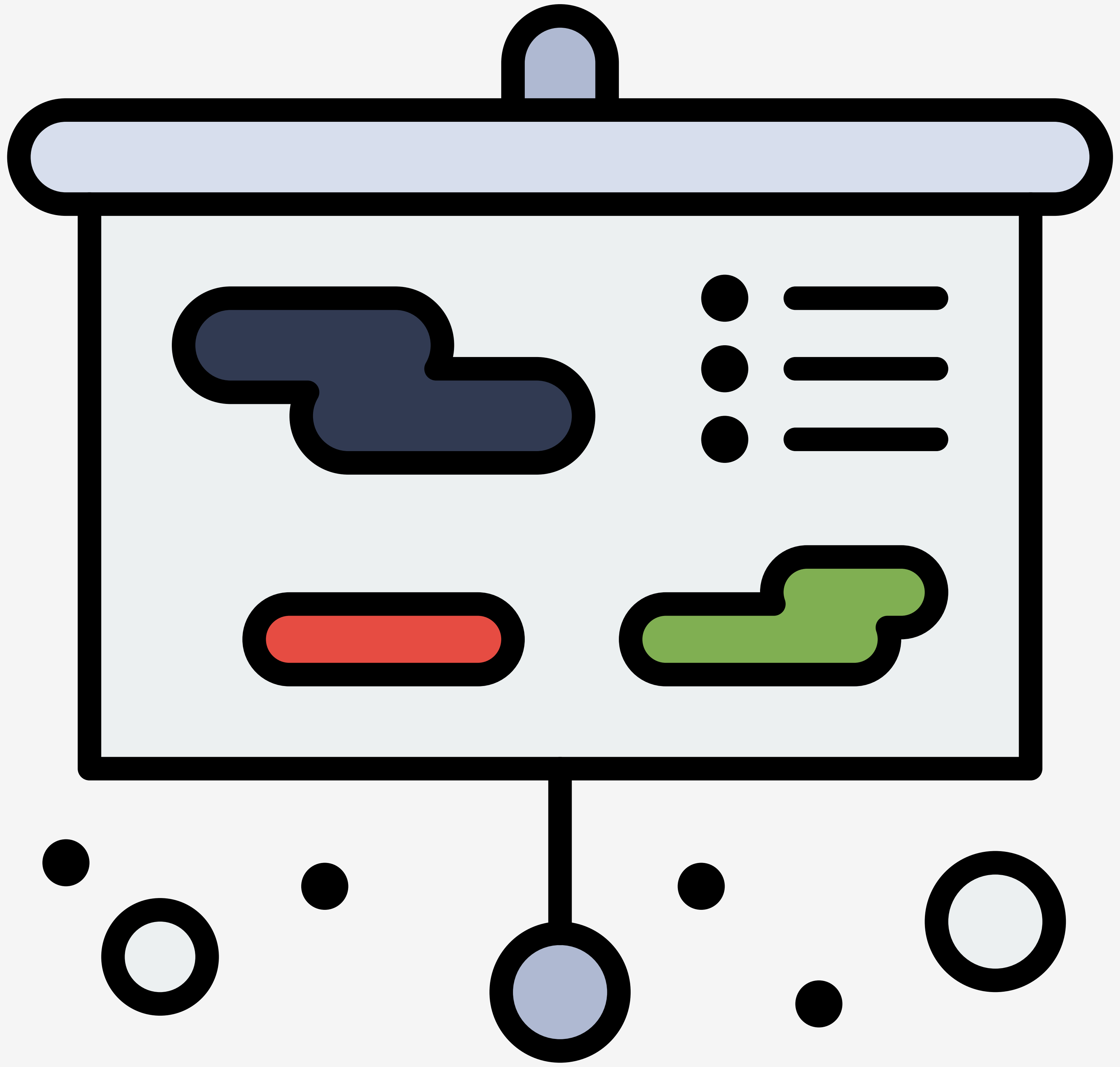
INTRODUCTION TO JAVASCRIPT

Lecture 3

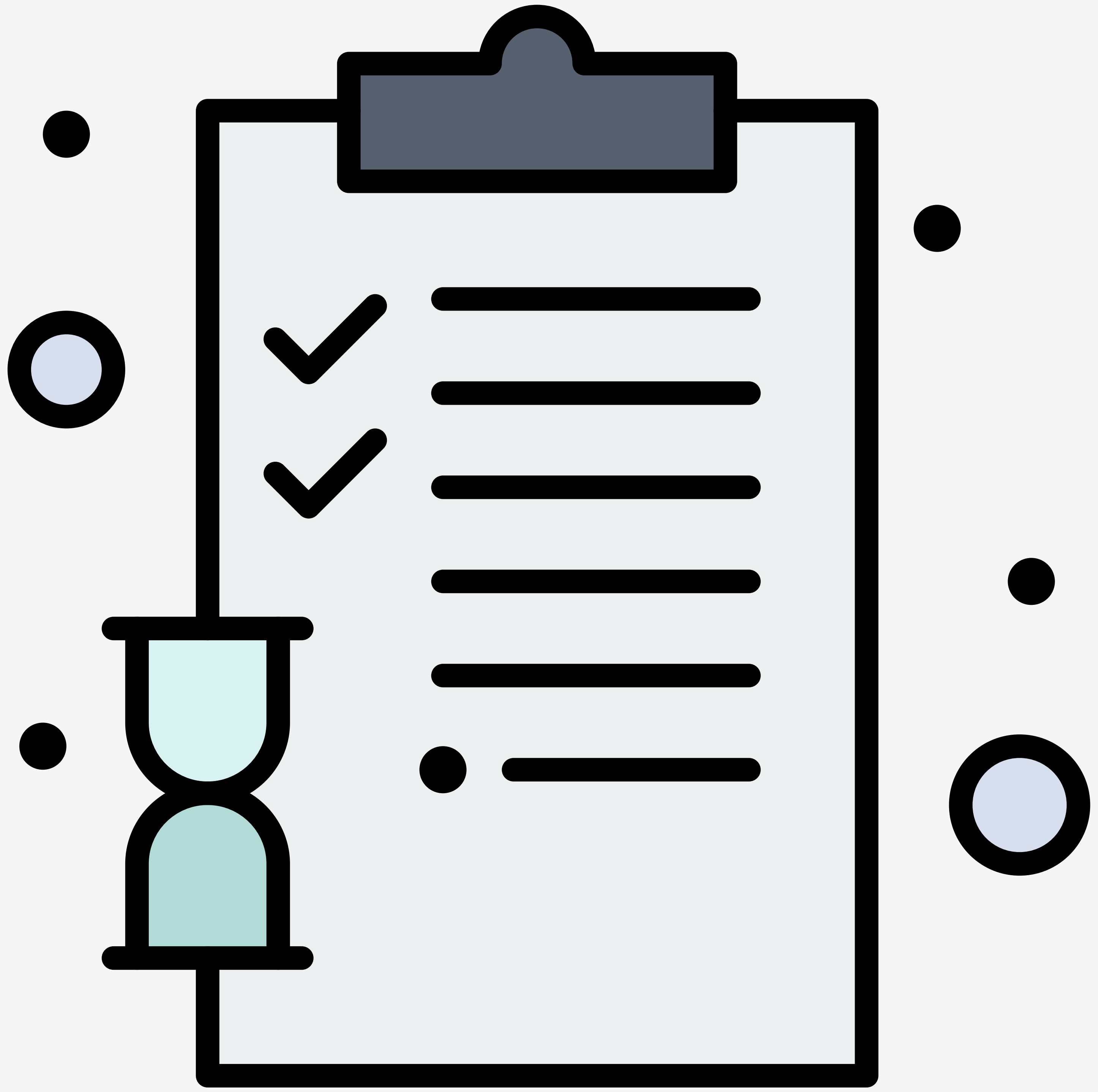
Graphics by Flatart on freepik.com

TODAY'S TOPICS



- Arrays
- Objects
- Participation: Crazy Cats

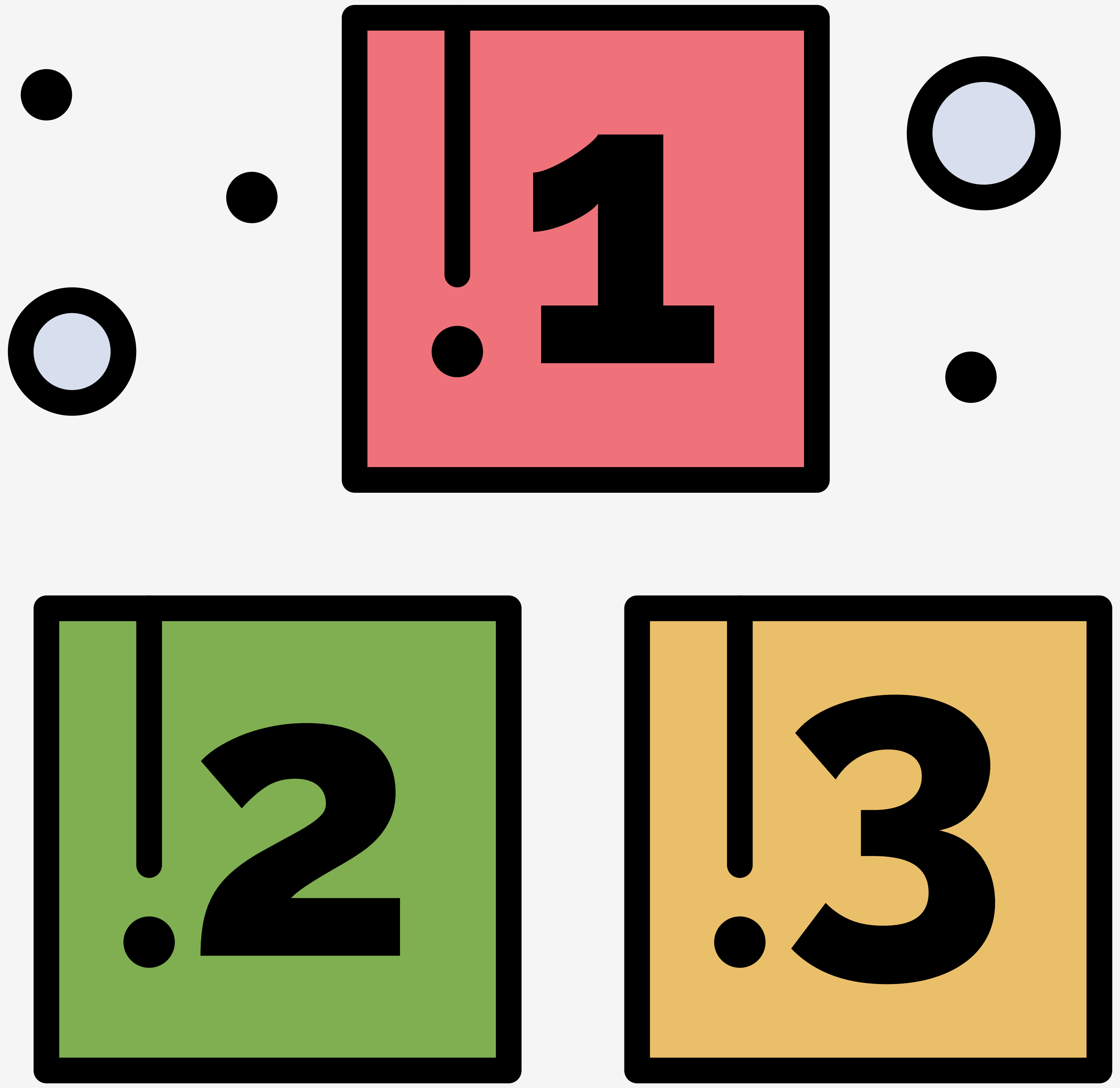
ANNOUNCEMENTS



- Sign-in Sheet
- Assignment Submissions
- Recordings

JAVASCRIPT ARRAYS

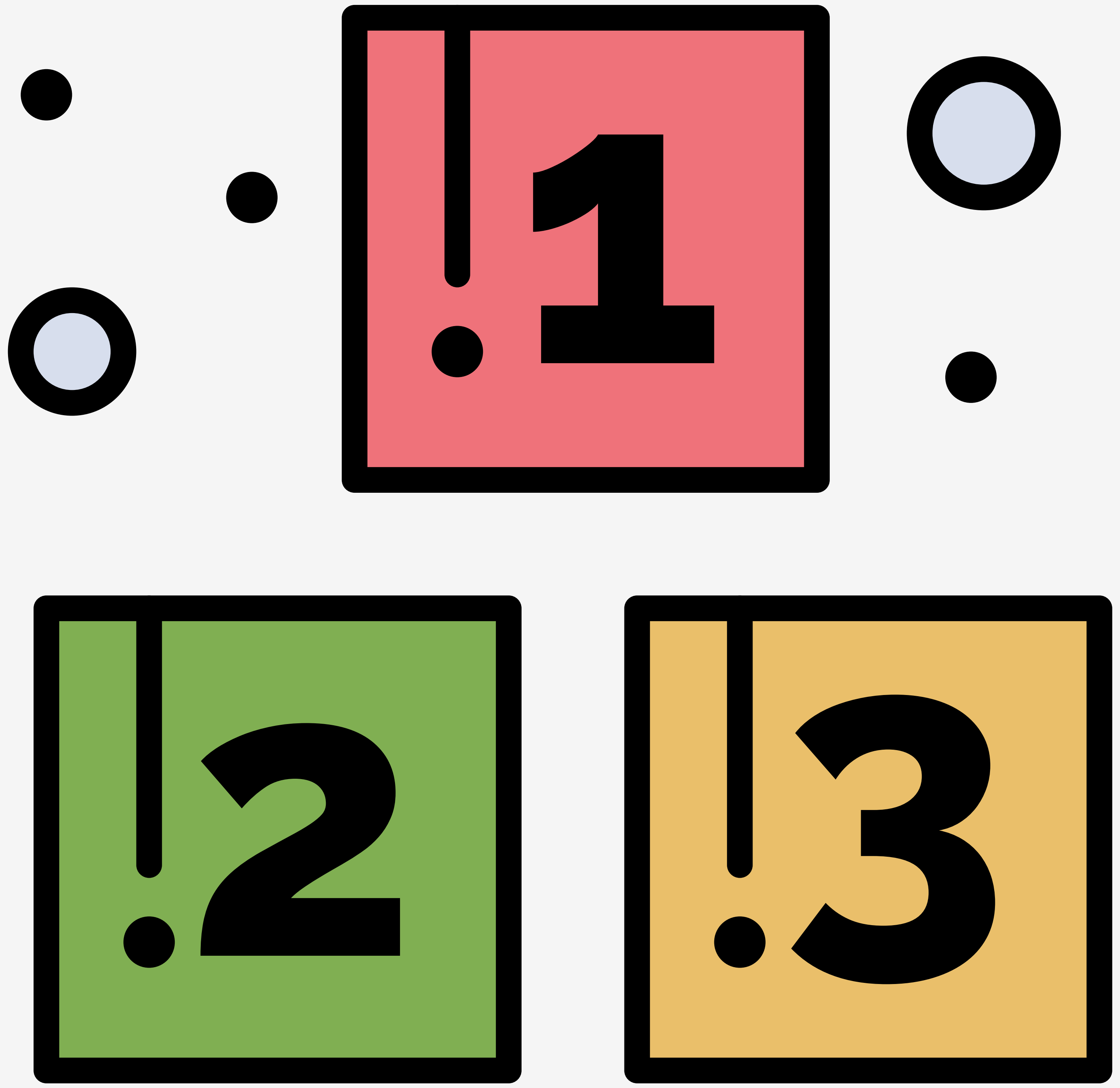
JAVASCRIPT ARRAYS



- An array is a comma separated list of items
- Each item is assigned a numbered index
- Arrays can contain items of many different data types
- An array is created using a set of square brackets ([ ])

```
// Creates an empty array
const empty = []

// Creates an array of colors
const colors = ['blue', 'green', 'yellow', 'red']

// Creates an array of numbers
const numbers = [1, 17, 8, 24]

// Creates a mixed array
const mixed = ['dog', 13, ['yes', 'no', 'maybe']]
```

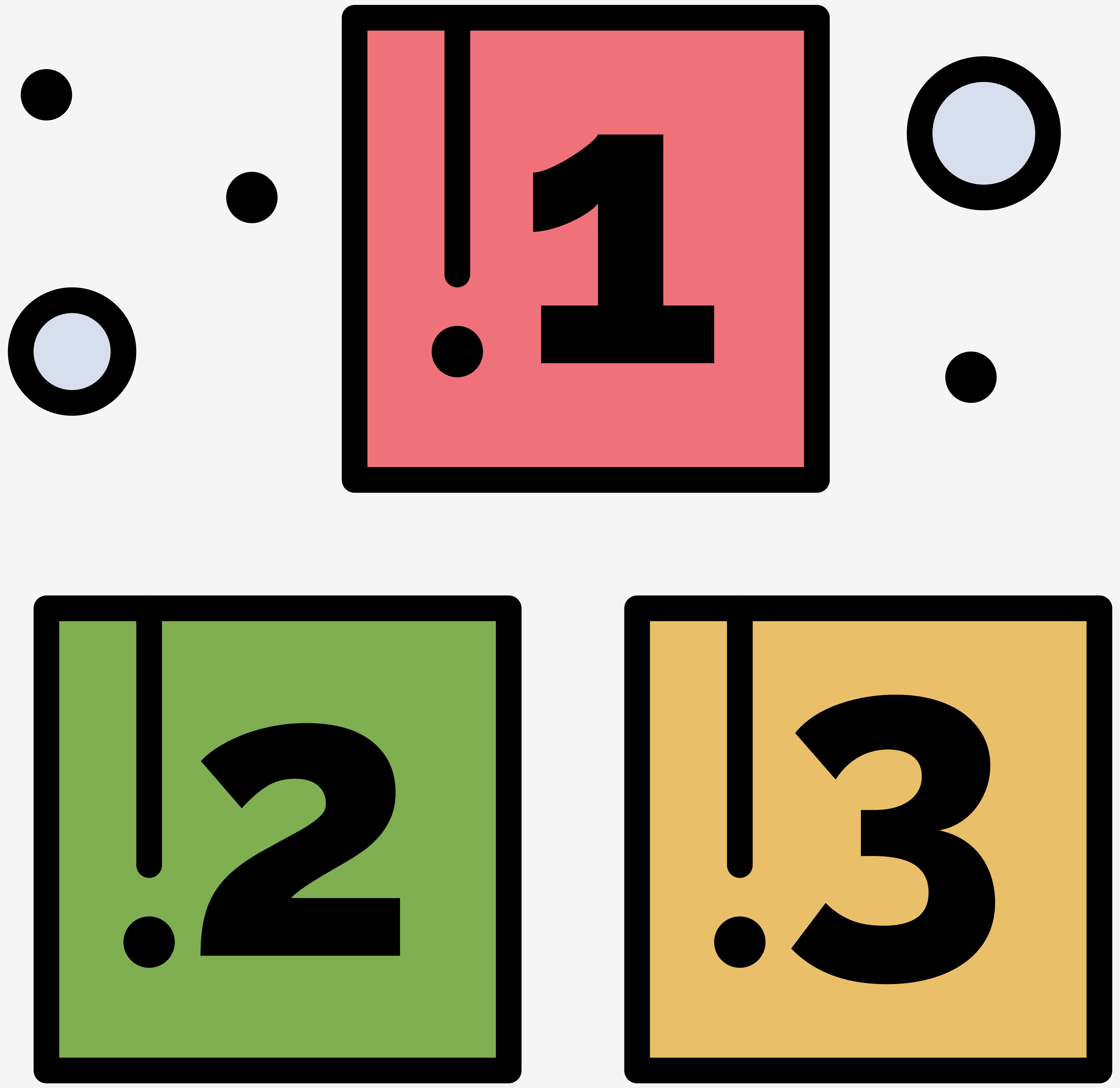
RETRIEVING ITEMS



- Items are retrieved from a array using the numbered index
- Indexes start with `0`, not `1`
- The index is placed inside a set of square brackets following the array variable
- This is known as Bracket Notation

```javascript
// Creates an array of colors
const colors = ['blue', 'green', 'yellow', 'red']

// Logging the FIRST color
console.log(colors[0]) // blue

// Logging the THIRD color
console.log(colors[2]) // yellow

// Logging the FIFTH color
console.log(colors[4]) // undefined
```

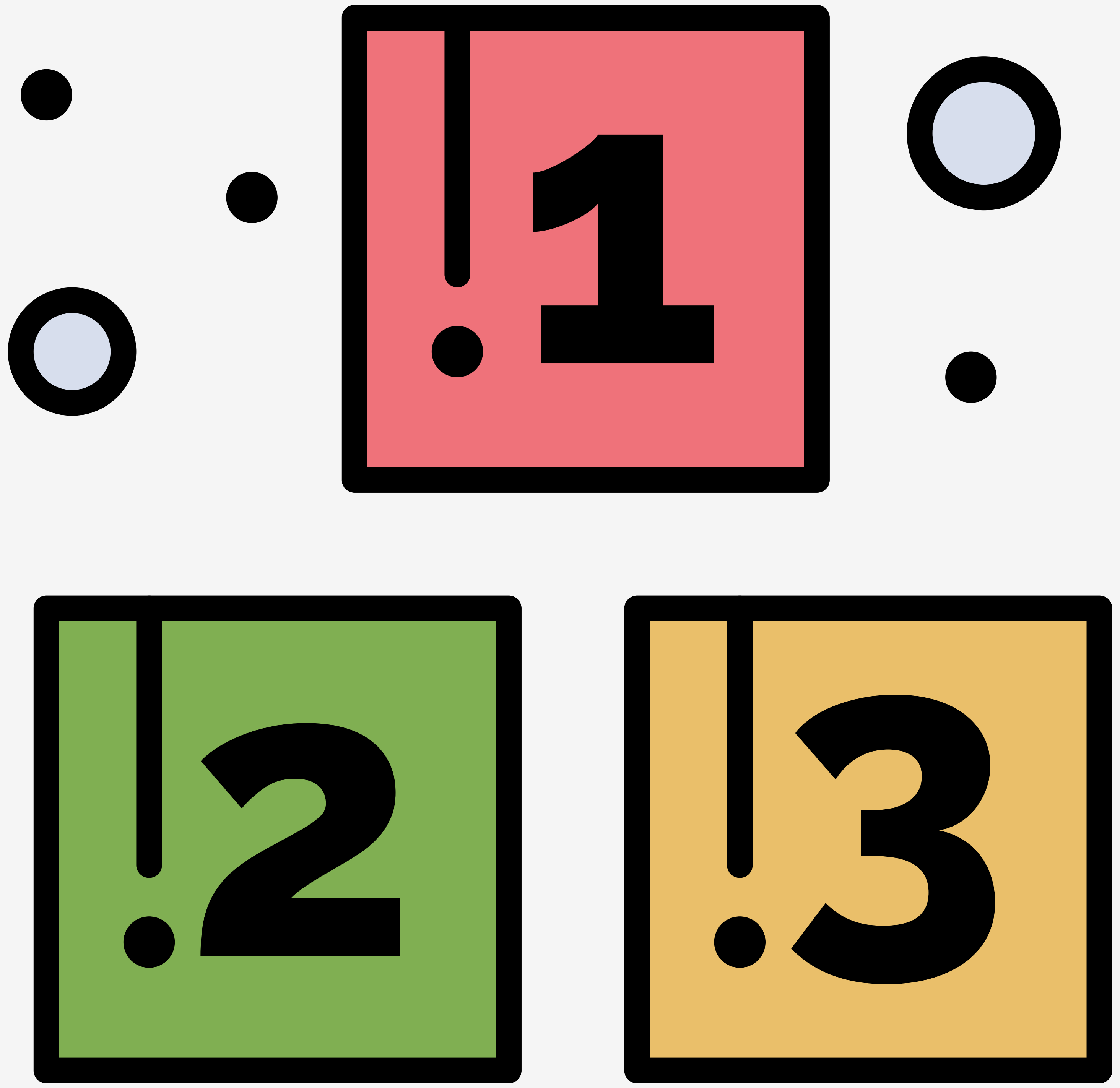
UPDATING ARRAY



- Items can be changed or added using bracket notation
- If an index does exist, the item will be updated
- If the index does NOT exist, the item will be added

```
// Creates a groceries list
const groceries = ['Milk', 'Eggs']

// Replaces the FIRST item
groceries[0] = 'Juice'

// Add an item
groceries[3] = 'Bread' Bad!
```

ARRAY MANIPULATION

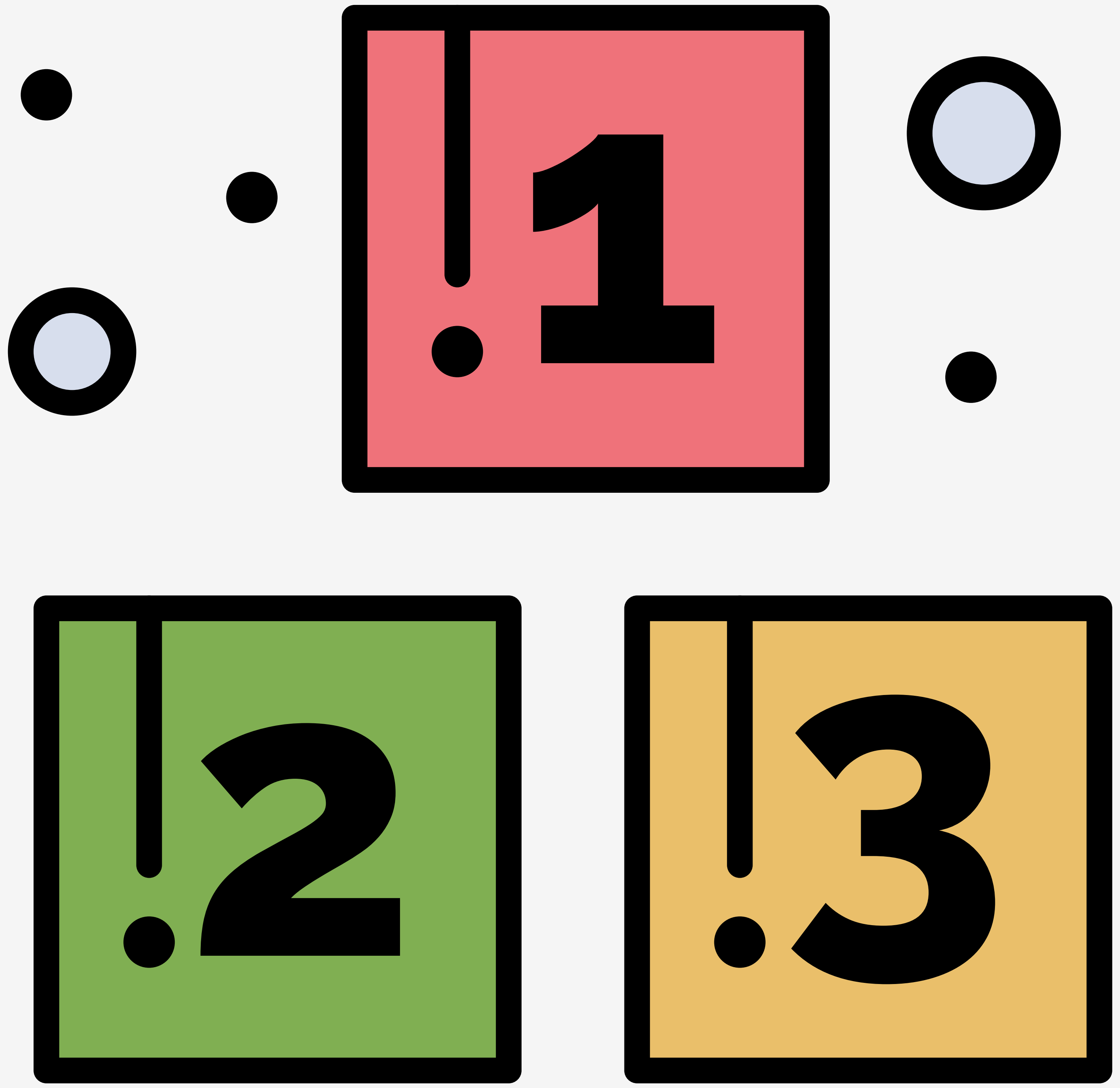


- Use `push()` to add items to the end of the array
- Use `pop()` to remove the last item in the array
- Use `shift()` to remove the first item in the array
- Use `unshift()` to add items to the beginning of the array
- Use `splice()` to add and / or remove items

```javascript
// Creates a groceries list
const groceries = ['Milk', 'Eggs']

// Add an item
groceries.push('Bread')

// Add multiple items
groceries.push('Sugar', 'Flour')

// ['Milk', 'Eggs', 'Bread', 'Sugar', 'Flour']
// Creates a groceries list
const groceries = ['Milk', 'Eggs']

// Remove the last item
groceries.pop()

// ['Milk']
// Creates a groceries list
const groceries = ['Milk', 'Eggs']

// Remove the first item
groceries.shift()

// ['Eggs']
// Creates a groceries list
const groceries = ['Milk', 'Eggs']

// Add an item
groceries.unshift('Bread')

// Add multiple items
groceries.unshift('Sugar', 'Flour')

// ['Sugar', 'Flour', 'Bread', 'Milk', 'Eggs']
```

```javascript
// Creates a groceries list
const groceries = ['Milk', 'Eggs', 'Bread']

// Add an item
groceries.splice(1, 0, 'Sugar')

// Remove an item
groceries.splice(2, 1)

// Add and Remove items
groceries.splice(0, 2, 'Flour')

// ['Flour', 'Bread']
```

ARRAY SORT



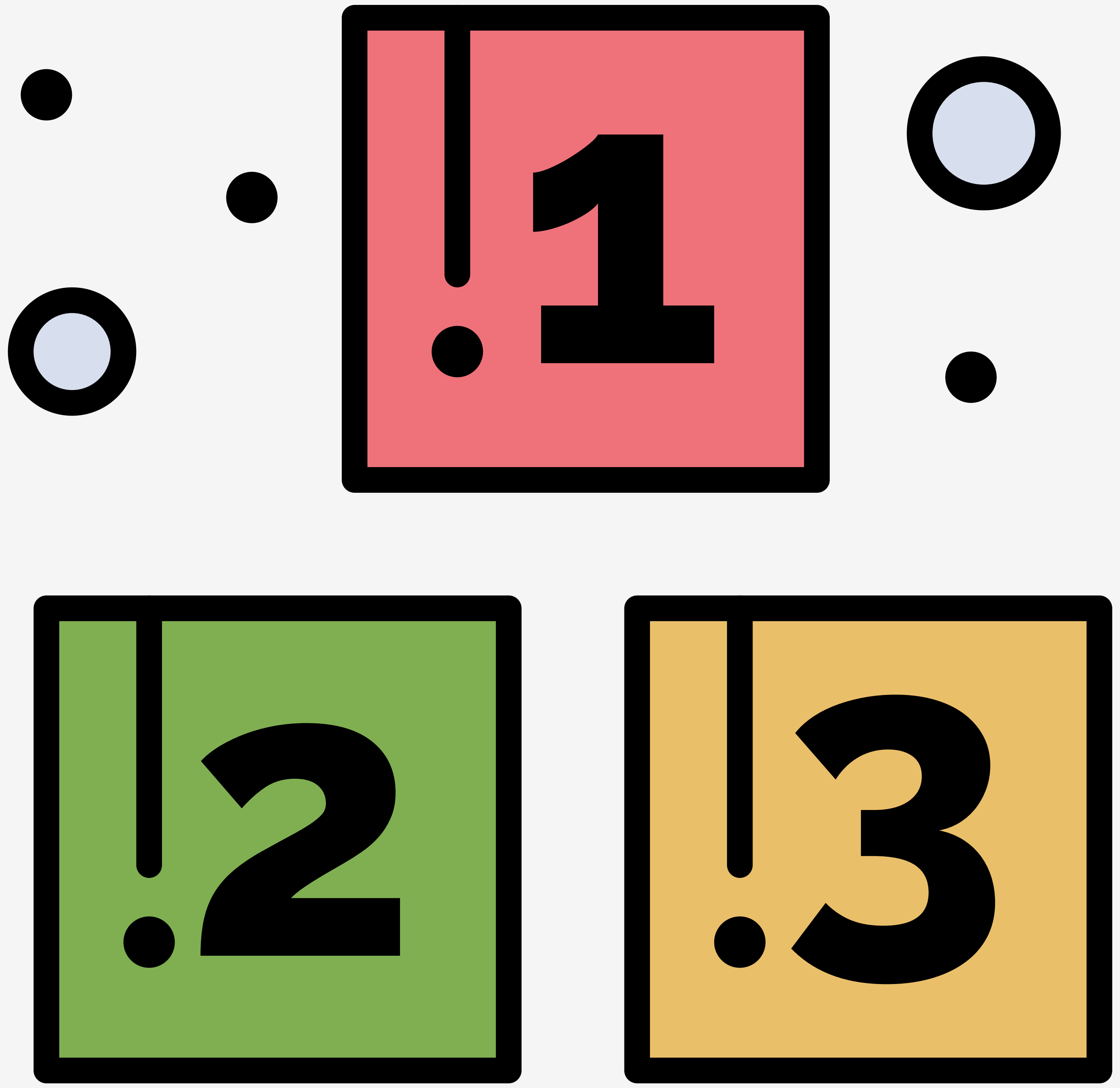
- Use `sort()` to sorts items by their UTF-16 code value
- Use `reverse()` to reverse the order of the items

```
const primary = ['red', 'yellow', 'blue']
primary.sort()

// ['blue', 'red', 'yellow']

const secondary = ['orange', 'Purple', 'green']
secondary.sort()

// ['Purple', 'green', 'orange']
```

```
const numbers = [1, 20, 4, 100]
numbers.sort()

// [1, 100, 20, 4]

// Adding a compare function to compare numbers
numbers.sort(function (a, b) {
  return a - b
})

// [1, 4, 20, 100]
```

```javascript
const numbers = [1, 2, 3, 4, 5]
numbers.reverse()

// [5, 4, 3, 2, 1]

const colors = ['red', 'blue', 'green', 'yellow']
colors.sort().reverse()

// ['yellow', 'red', 'green', 'blue']
```

ARRAY JOIN

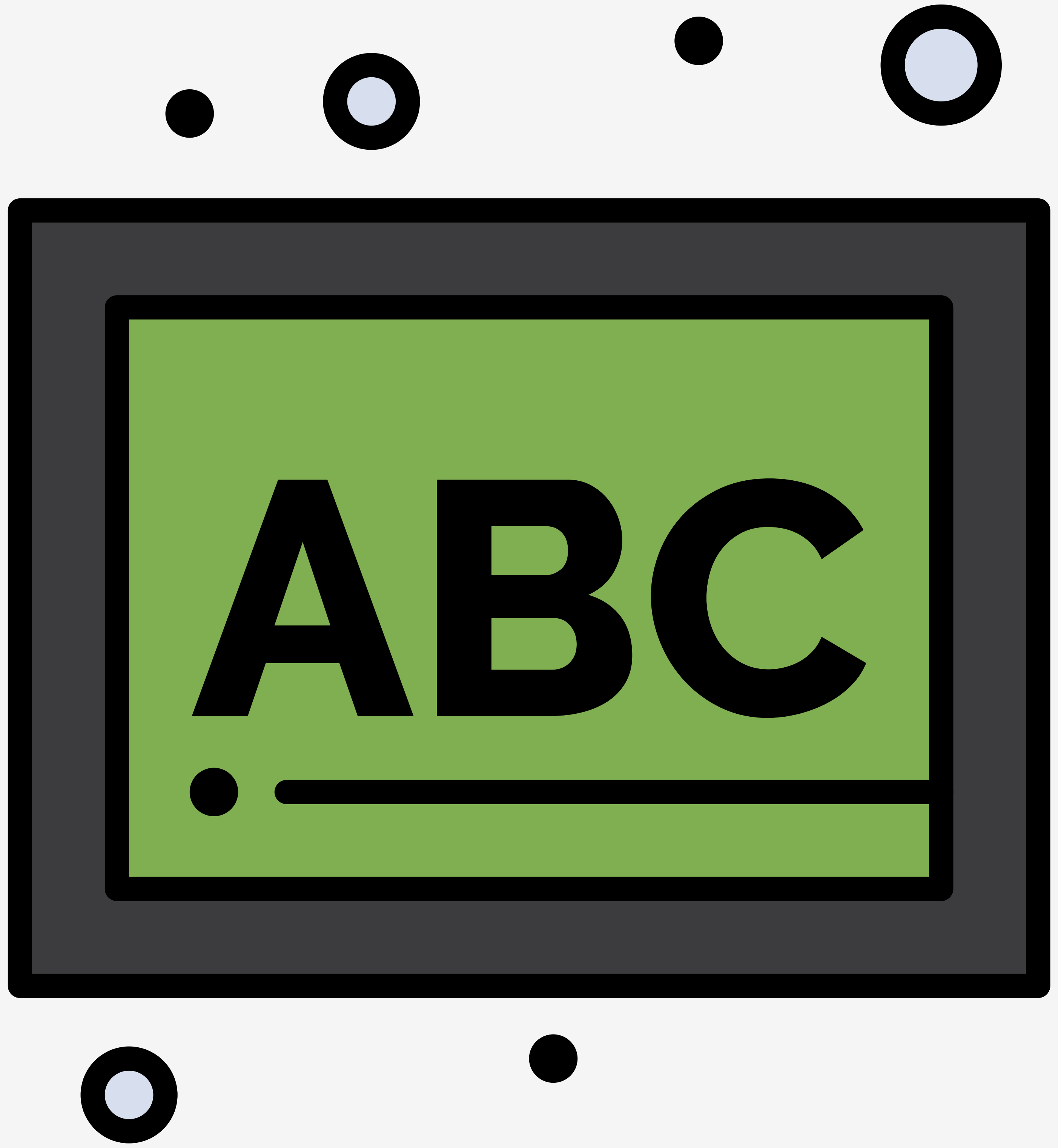


- Use `join()` combine all array items into a string
- A separator can be provided to place between each item.
- If no separator is given, a comma (`,`) will be used

```
// Creates a list of fruit
const fruits = ['Apples', 'Bananas', 'Cherries']

console.log(fruits.join())
// Apples,Bananas,Cherries

console.log(fruits.join(''))
// ApplesBananasCherries

console.log(fruits.join(' | '))
// Apples | Bananas | Cherries
```

JAVASCRIPT OBJECTS

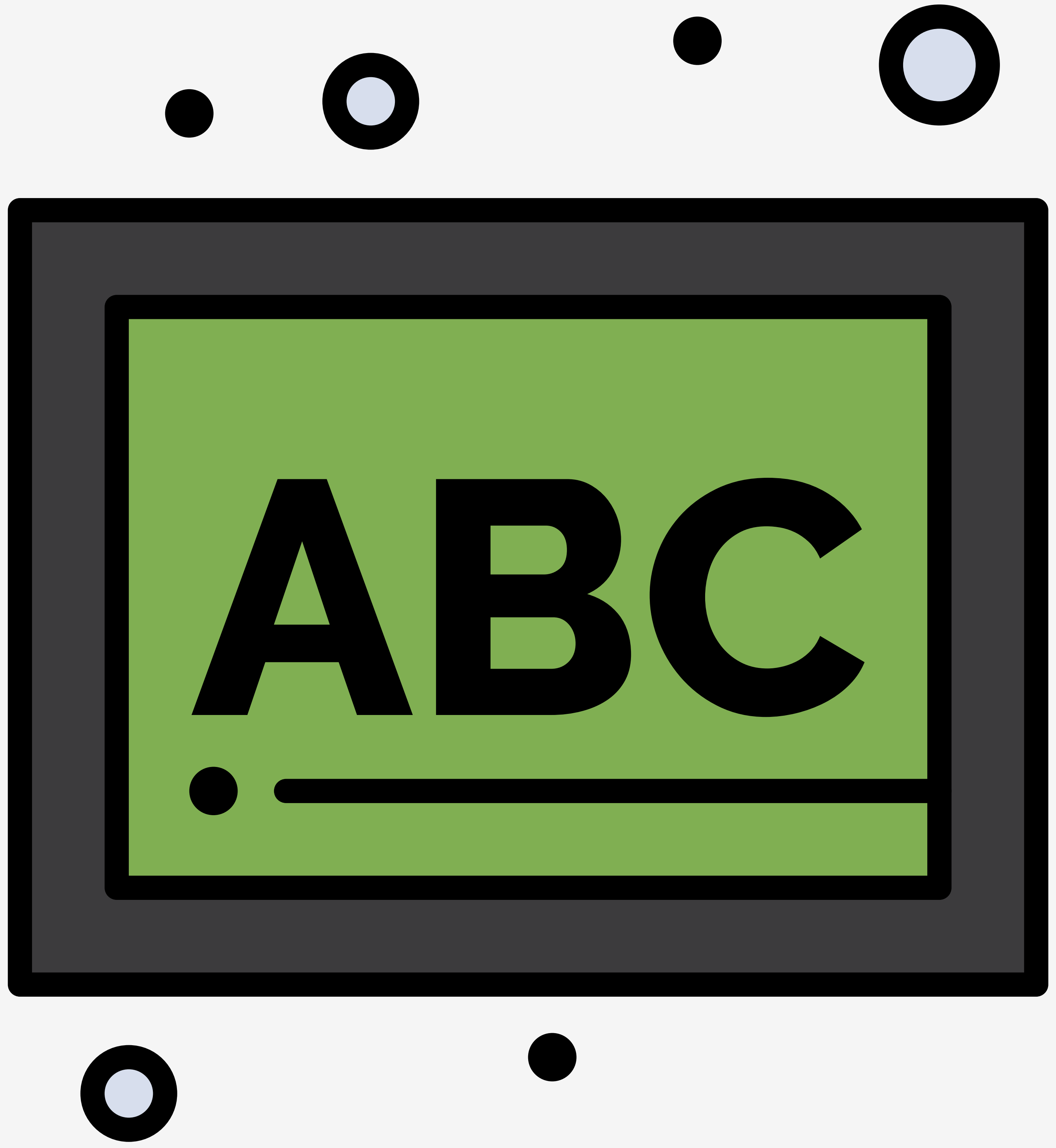
JAVASCRIPT OBJECTS



- An object is a collection of key / value pairs
- These key / values pairs are called properties
- Objects are created using a set of curly braces ({})
- Each property is made up of a key and a value separated by a colon (:)
- Properties are separated by commas

```
// An object with no properties
const empty = {}

// An object with properties
const car = {
  year: 2019,
  make: 'Toyota',
  model: 'Prius'
}
```

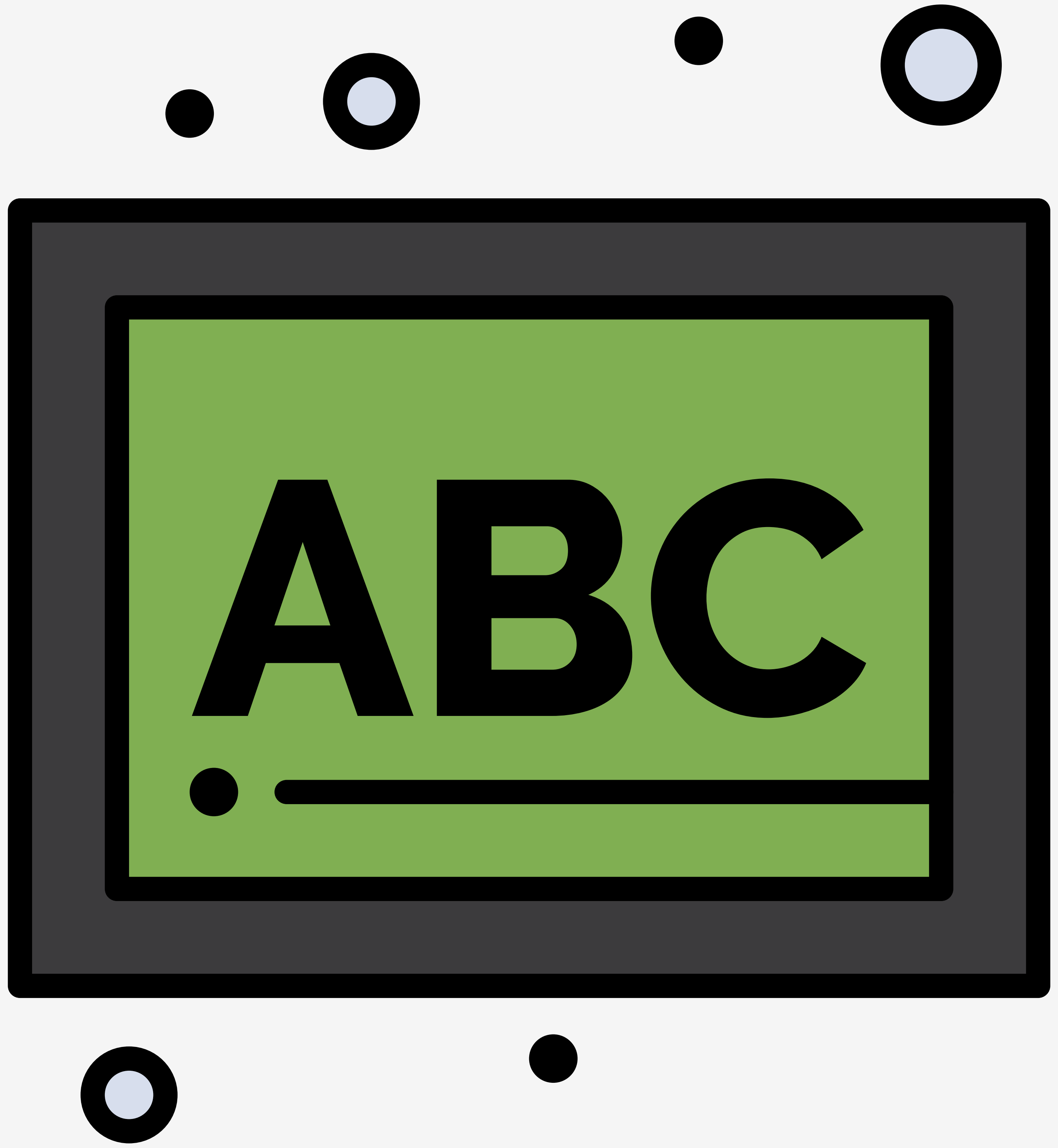
RETRIEVING PROPERTIES



- The property key is used to retrieve the property value
- Bracket Notation with the key placed inside the square brackets as a string
- Dot Notation with the object variable is followed by a period (.) and the key

```
const car = {
  year: 2019,
  make: 'Toyota',
  model: 'Prius'
}

// dot notation
car.make // 'Toyota'

// bracket notation
car['model'] // 'Prius'
```

UPDATING OBJECTS



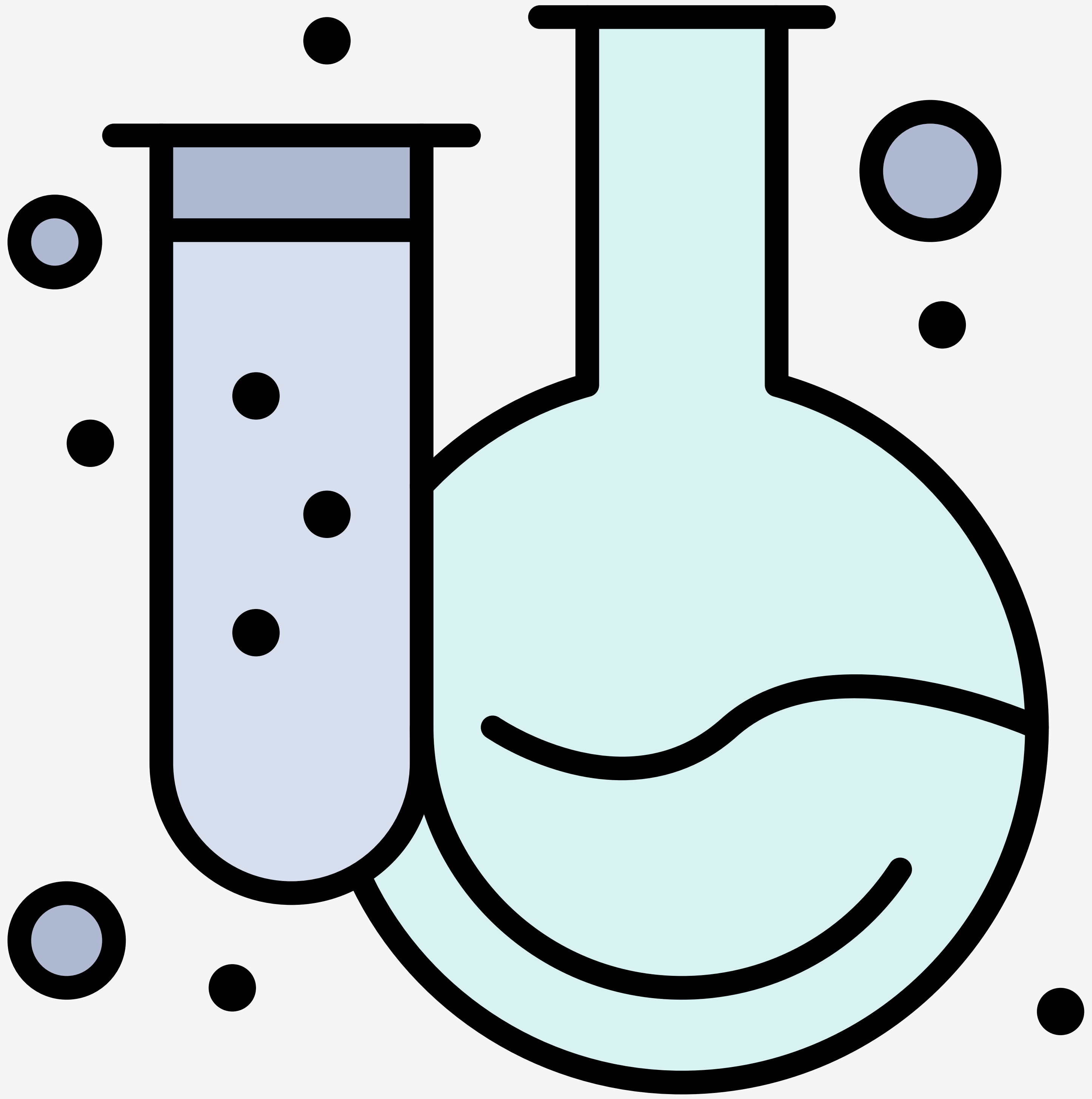
- Properties can be updates or added
- Bracket and dot notation can be used
- If a property does not exists, it will be added
- Use the `delete` keyword to remove properties

```javascript
const car = {
  year: 2019,
  make: 'Toyota',
  model: 'Prius'
}

// Update properties
car.make = 'Telsa'
car['model'] = 'Model 3'

// Add properties
car.color = 'Red'
car['range'] = 500
```

```
const car = {
  year: 2019,
  make: 'Toyota',
  model: 'Prius'
}

// Remove properties
delete car.make
delete car['model']
```

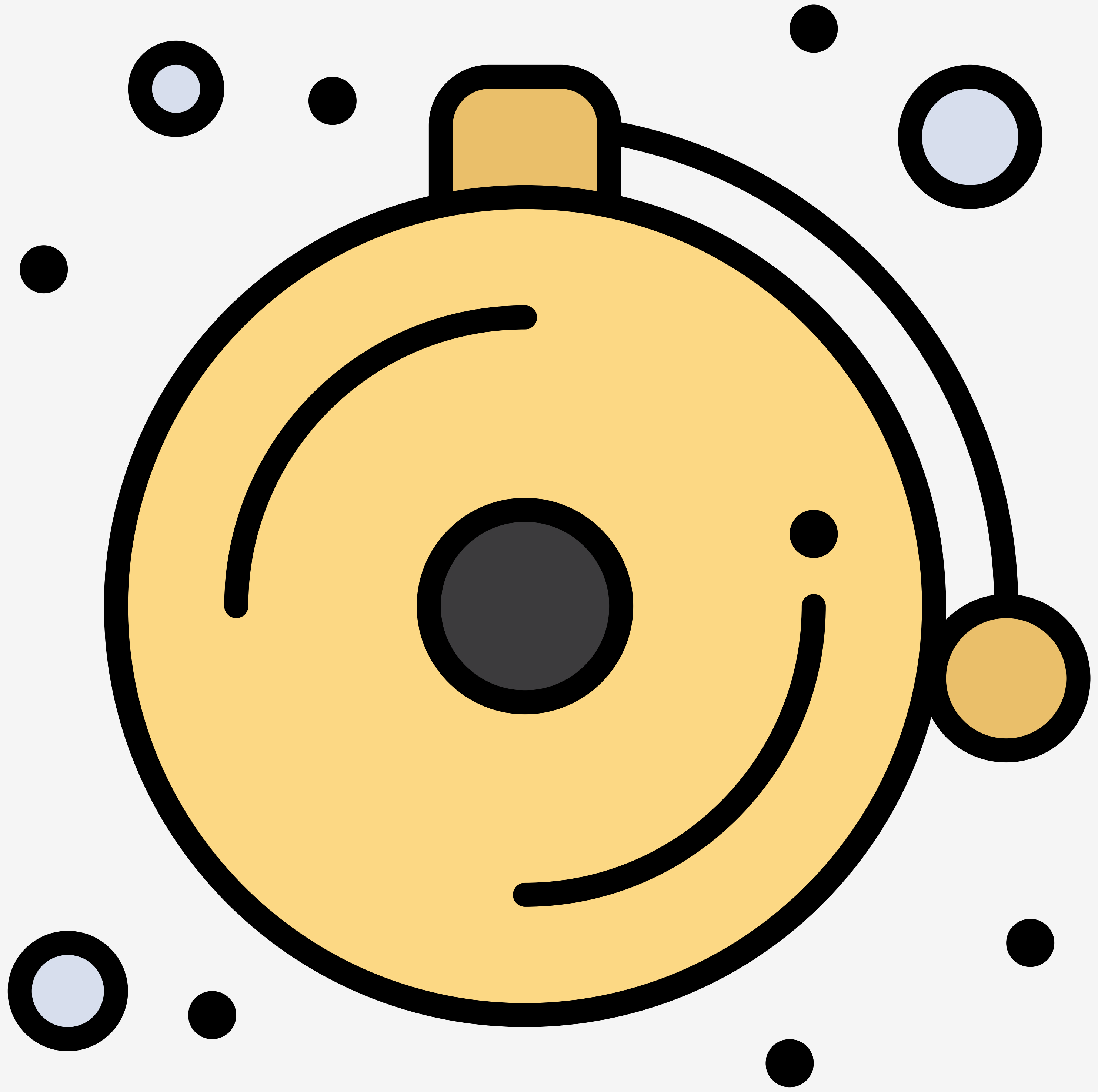
PRACTICE

CRAZY CATS



- *FORK THE PEN!*
- Create objects with the properties: `name`, `size`, `color`, and `age`
- Click the button to create cats
- Submit the URL to your pen
- *DUE:* **Thu. Sep. 19 @ 11:59 PM**

NEXT TIME...



- **Hands-on:** Arrays and Objects
- **Exercise:** Madlibs