

JS Browser

ΜΑΘΗΜΑ 3.1

WINDOW

ΠΕΡΙΕΧΟΜΕΝΑ:

1. Το αντικείμενο Window
2. Properties
 1. Μέγεθος Ορατής Περιοχής
 2. Παράθυρο σε Σχέση με την Οθόνη
 3. Scroll
3. Μέθοδοι
 1. Κίνηση Παραθύρου
 2. Αλλαγή Διαστάσεων Παραθύρου
 3. Scrolling

ΜΑΘΗΜΑ 3.1: WINDOW

1. Το αντικείμενο window

Το αντικείμενο window:

- Είναι το “καθολικό αντικείμενο” (global object) που αντιπροσωπεύει ένα ανοιχτό tab στον browser.
 - Το έχουμε συναντήσει σε διάφορα σημεία (π.χ. είδαμε ότι οι μεταβλητές που δηλώνονται με var, γίνονται μέλη αυτού του αντικειμένου) και τώρα θα το δούμε αναλυτικά
- Περιέχει (υπερβολικά) πολλά πράγματα, μεταξύ των οποίων βρίσκουμε:

Περιέχει	Επεξήγηση
document	Το αντικείμενο που αντιπροσωπεύει το DOM
location	Αντικείμενο με πληροφορίες για το URL και πλοήγηση
history	Αντικείμενο με το ιστορικό του session (back, forward)
navigator	Αντικείμενο με πληροφορίες συσκευής και browser
screen	Αντικείμενο με πληροφορίες οθόνης
timers	Μέθοδοι για ορισμό χρονισμού στην εκτέλεση κώδικα
dialogs	Μέθοδοι για διάλογο με το χρήστη
window control	Διαχείριση tabs
console	Αντικείμενο που ελέγχει την κονσόλα
storage	Αντικείμενα που ελέγχουν την αποθήκευση στον browser
speech	Αναγνώριση και Σύνθεση Ομιλίας

Παράδειγμα 1: window

```
console.dir(window);
```

```
▼ Window 1 window.html:44
  ▶ alert: f alert()
  ▶ atob: f atob()
  ▶ blur: f blur()
  ▶ btoa: f btoa()
  ▶ caches: CacheStorage {}
  ▶ cancelAnimationFrame: f cancelAn
  ▶ cancelIdleCallback: f cancelIdle
  ▶ captureEvents: f captureEvents()
  ▶ chrome: {loadTimes: f, csi: f}
  ▶ clearInterval: f clearInterval()
  ▶ clearTimeout: f clearTimeout()
  ▶ clientInformation: Navigator {ve
  ▶ close: f close()
    closed: false
  ▶ confirm: f confirm()
  ▶ cookieStore: CookieStore {onchan
  ▶ createImageBitmap: f createImage
    credentials: false
    crossOriginIsolated: false
  ▶ crypto: Crypto {subtle: SubtleCr
  ▶ customElements: CustomElementReg
  ▶ devicePixelRatio: 1.5
  ▶ document: document
  ▶ documentPictureInPicture: Docume
    event: undefined
  ▶ external: External {}
    fence: null
  ▶ fetch: f fetch()
  ▶ find: f find()
  ▶ focus: f focus()
  ▶ frameElement: null
  ▶ frames: Window {window: Window,
  ▶ getComputedStyle: f getComputedS
  ▶ getScreenDetails: f getScreenDet
  ▶ getSelection: f getSelection()
  ▶ history: History {length: 5, scr
  ▶ indexedDB: IDBFactory {}
  ▶ innerHeight: 1304
  ▶ innerWidth: 683
  ▶ isSecureContext: true
  ▶ launchQueue: LaunchQueue {}
    length: 0
  ▶ localStorage: Storage {length: 0
  ▶ location: Location {ancestorOrig
  ▶ locationbar: BarProp {visible: t
  ▶ matchMedia: f matchMedia()
  ▶ menubar: BarProp {visible: true}
  ▶ moveBy: f moveBy()
  ▶ moveTo: f moveTo()
    name: ""
  ▶ navigation: Navigation {currentE
  ▶ navigator: Navigator {vendorSub:
    onabort: null
    onafterprint: null
    onanimationend: null
    onanimationiteration: null
    onanimationstart: null
    onappinstalled: null
    onauxclick: null
    onbeforeinput: null
    onbeforeinstallprompt: null
    onbeforematch: null
    onbeforeprint: null
    onbeforetoggle: null
    onbeforeunload: null
    onbeforexrselect: null
    onblur: null
    oncancel: null
    oncanplay: null
    oncanplaythrough: null
    onchange: null
    onclick: null
    onclose: null
    oncontentvisibilityautostatechan
    oncontextlost: null
    oncontextmenu: null
    oncontextrestored: null
    oncuechange: null
    ondblclick: null
    ondevicemotion: null
    ondeviceorientation: null
    ondeviceorientationabsolute: nul
    ondrag: null
    ondragend: null
    ondragenter: null
    ondragleave: null
    ondragover: null
    ondragstart: null
    ondrop: null
    ondurationchange: null
    onemptied: null
    onended: null
    onerror: null
    onfocus: null
    onformdata: null
    ongotpointercapture: null
    onhashchange: null
    oninput: null
    oninvalid: null
    onkeydown: null
    onkeypress: null
    onkeyup: null
    onlanguagechange: null
    onload: null
    onloadeddata: null
    onloadedmetadata: null
    onloadstart: null
    onlostpointercapture: null
    onmessage: null
    onmessage
    parent: Window {window: Window,
    performance: Performance {timeOr
    personalbar: BarProp {visible: t
    postMessage: f postMessage()
    print: f print()
    prompt: f prompt()
    queryLocalFonts: f queryLocalFon
    queueMicrotask: f queueMicrotask
    releaseEvents: f releaseEvents()
    reportError: f reportError()
    requestAnimationFrame: f request
    requestIdleCallback: f requestId
    resizeBy: f resizeBy()
    resizeTo: f resizeTo()
    scheduler: Scheduler {}
    screen: Screen {availWidth: 2560
    screenLeft: -14
    screenTop: 0
    screenX: -14
    screenY: 0
    scroll: f scroll()
    scrollBy: f scrollBy()
    scrollTo: f scrollTo()
    scrollX: 0
    scrollY: 0
    scrollbar: BarProp {visible: tr
    self: Window {window: Window, se
    sessionStorage: Storage {length:
    setInterval: f setInterval()
    setTimeout: f setTimeout()
    sharedStorage: SharedStorage {wo
    showDirectoryPicker: f showDirec
    showOpenFilePicker: f showOpenFi
    showSaveFilePicker: f showSaveFi
    speechSynthesis: SpeechSynthesis
      status: ""
    statusBar: BarProp {visible: tru
    stop: f stop()
    structuredClone: f structuredClo
    styleMedia: StyleMedia {type: 's
    toolbar: BarProp {visible: true}
    top: Window {window: Window, sel
    trustedTypes: TrustedTypePolicyF
    visualViewport: VisualViewport {
    webkitCancelAnimationFrame: f we
    webkitRequestAnimationFrame: f w
    webkitRequestFileSystem: f webki
    webkitResolveLocalFileSystemURL:
    window: Window {window: Window,
      Infinity: Infinity
    AbortController: f AbortControlL
    AbortSignal: f AbortSignal()
    AbsoluteOrientationSensor: f Abs
    AbstractRange: f AbstractRange()
    Accelerometer: f Accelerometer()
    AggregateError: f AggregateError
    AnalyserNode: f AnalyserNode()
    Animation: f Animation()
```

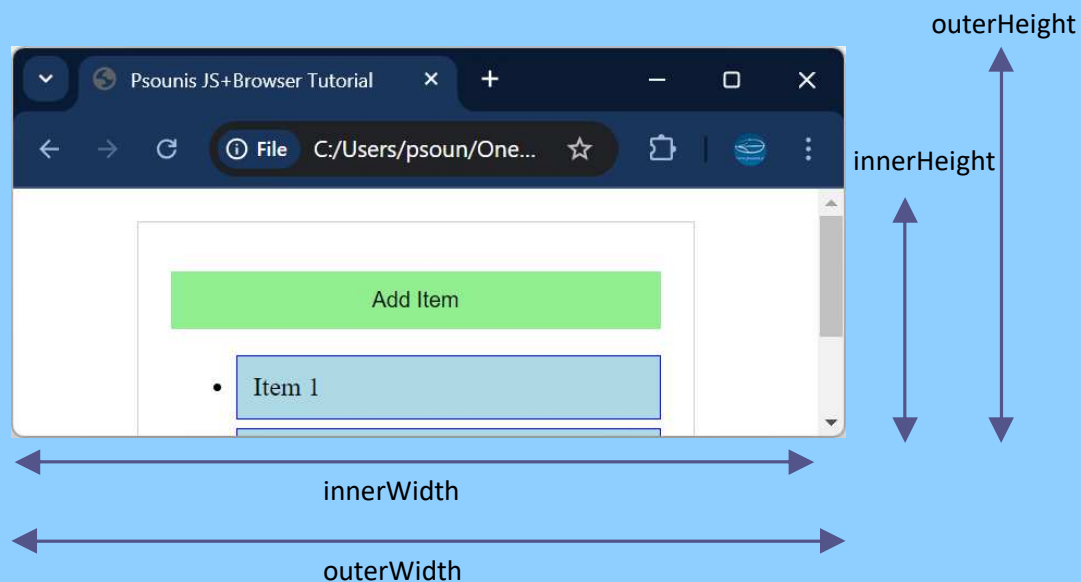
ΜΑΘΗΜΑ 3.1: WINDOW

2.1. Μέγεθος Ορατής Περιοχής

Μέγεθος Ορατής Περιοχής:

- Με τα παρακάτω properties βλέπουμε το μέγεθος της ορατής περιοχής (**viewport**):

Property	Επεξήγηση
innerWidth	Πλάτος viewport (χωρίς toolbars, scrollbars)
innerHeight	Ύψος viewport (χωρίς toolbars, scrollbars)
outerWidth	Πλάτος viewport (με toolbars, scrollbars)
outerHeight	Ύψος viewport (με toolbars, scrollbars)



Παράδειγμα 2: viewport-size.html

```
<div id="info">Resize the window to change the background color!</div>
```

```
function updateBackgroundColor() {  
  const width = window.innerWidth;  
  const height = window.innerHeight;  
  const info = document.getElementById('info');  
  
  // Determine background color based on window size  
  if (width > 1000 && height > 700) {  
    document.body.style.backgroundColor = 'red';  
    info.textContent = `Large Viewport: ${width}px x ${height}px`;  
  } else if (width > 700 && height > 500) {  
    document.body.style.backgroundColor = 'green';  
    info.textContent = `Medium Viewport: ${width}px x ${height}px`;  
  } else {  
    document.body.style.backgroundColor = 'blue';  
    info.textContent = `Small Viewport: ${width}px x ${height}px`;  
  }  
}  
  
// Initial call to set background color based on current size  
updateBackgroundColor();  
  
// Event listener for window resize  
window.addEventListener('resize', updateBackgroundColor);
```

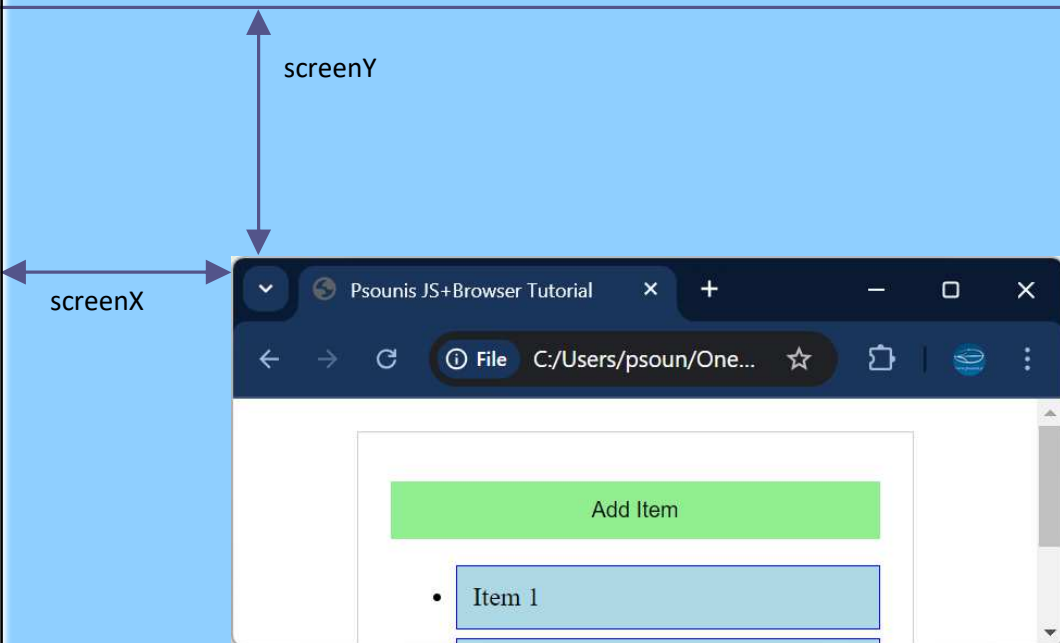
ΜΑΘΗΜΑ 3.1: WINDOW

2.2. Παράθυρο σε Σχέση με την Οθόνη

Παράθυρο σε Σχέση με την Οθόνη:

- Με τα παρακάτω properties βλέπουμε την θέση του παραθύρου σε σχέση με την οθόνη:

Property	Επεξήγηση
screenX	Μετατόπιση στον x-άξονα (από αριστερά)
screenY	Μετατόπιση στον y-άξονα (από πάνω)



Παράδειγμα 3: screenX-Y.html

```
<div id="message">Move the window to match the target position!</div>
<div id="target"></div>
<div id="position">Current Position: X: <span id="posX">0</span>, Y:
<span id="posY">0</span></div>

// Random target position within screen bounds
const targetX = Math.floor(Math.random() * 300);
const targetY = Math.floor(Math.random() * 300);
// Display the target coordinates
document.getElementById('target').textContent = `Target Position: X: ${targetX}, Y: ${targetY}`;

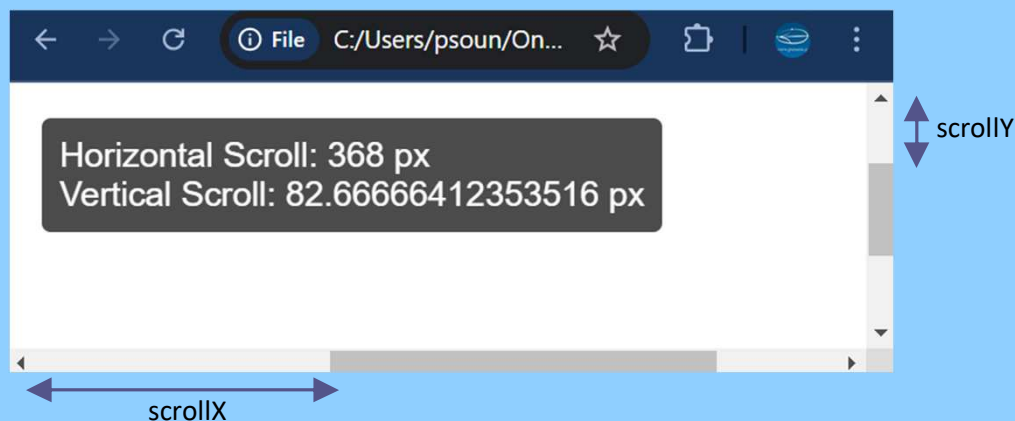
// Function to update the current position of the window
function updatePosition() {
  const posX = window.screenX;
  const posY = window.screenY;
  document.getElementById('posX').textContent = posX;
  document.getElementById('posY').textContent = posY;
  // Check if the window position matches the target
  if (Math.abs(posX - targetX) < 10 && Math.abs(posY - targetY) < 10) {
    document.getElementById('message').textContent = 'You hit the target!';
    document.getElementById('message').style.color = 'green';
  }
}

// Set interval to update the position every 100 milliseconds
setInterval(updatePosition, 100);
// Initial position check
updatePosition();
```

Properties για Scroll:

- Με τα παρακάτω properties βλέπουμε το μέγεθος της σελίδας που έχει γίνει scroll:

Property	Επεξήγηση
scrollX	Pixels που η σελίδα έχει μετατοπιστεί κάθετα
scrollY	Pixels που η σελίδα έχει μετατοπιστεί οριζόντια
pageXOffset	Συνώνυμο του scrollX
pageYOffset	Συνώνυμο του scrollY



Παράδειγμα 4: scroll.html

```
<div class="scroll-info">
  Horizontal Scroll: <span id="scrollX">0</span> px<br>
  Vertical Scroll: <span id="scrollY">0</span> px
</div>
```

```
function updateScrollPosition() {
  // Get the current scroll positions
  const scrollX = window.scrollX;
  const scrollY = window.scrollY;

  // Display the scroll positions
  document.getElementById('scrollX').textContent = scrollX;
  document.getElementById('scrollY').textContent = scrollY;
}

// Update the scroll position on scroll event
window.addEventListener('scroll', updateScrollPosition);

// Initial update to set positions on page load
updateScrollPosition();
```


Μέθοδοι για Κίνηση Παραθύρου:

- Με τις παρακάτω μεθόδους μπορούμε να κινήσουμε το παράθυρο:

Μέθοδος	Επεξήγηση
moveBy(x,y)	Το παράθυρο μετακινείται κατά το πλήθος pixel στους άξονες
moveTo(x,y)	Μετακινεί το παράθυρο σε συγκεκριμένες συντεταγμένες

Σημείωση:

- Οι παραπάνω μέθοδοι μπλοκάρονται από τον Google Chrome και δεν έχουν κάποια επίδραση στη σελίδα (για λόγους ασφαλείας και ευχάριστης αλληλεπίδρασης με τη σελίδα)
- Ωστόσο μπορούμε να τις χρησιμοποιήσουμε αν ανοίξουμε ένα popup window με τη μέθοδο του Window:

Μέθοδος	Επεξήγηση
open(url, name)	Ανοίγει νέο παράθυρο με το url και έχει το όνομα name Θέτοντας την κενή συμβολοσειρά και στα δύο ορίσματα, έχουμε ένα απλό popup για χρήση από τη σελίδα μας. Επίσης έχει ορίσματα με όνομα width και height για τις διαστάσεις του popup

Παράδειγμα 5: moveBy.html

<button id="openWindow">Open and Move Window</button>

```
let newWindow;

document.getElementById('openWindow').addEventListener('click', () => {
  // Open a new window with a small size
  newWindow = window.open("", "", 'width=300,height=300');

  // Check if the new window was created successfully
  if (newWindow) {
    // Add another button inside the pop-up to move it by (50, 50)
    newWindow.document.body.innerHTML = '<button id="moveWindow">Move Window by (50, 50)</button>';

    // Add an event listener to the button in the new window

    newWindow.document.getElementById('moveWindow').addEventListener('click', () => {
      newWindow.moveBy(50, 50); // Moves the new window 50 pixels right and down
    });
  } else {
    alert('Pop-up blocked! Please allow pop-ups for this site.');
```

Μέθοδοι για Αλλαγή Διαστάσεων Παραθύρου:

- Με τις παρακάτω μεθόδους μπορούμε να αλλάξουμε τις διαστάσεις του παραθύρου:

Μέθοδος	Επεξήγηση
resizeBy(x,y)	x: Αλλαγή Διαστάσεων κατά τον x-άξονα y: Αλλαγή Διαστάσεων κατά τον y-άξονα
resizeTo(width,height)	Αλλάζει τις διαστάσεις στο καθορισμένο width και height

Σημείωση:

- Επίσης αυτές οι μέθοδοι ισχύουν μόνο για popup
- Ενδέχεται άλλοι browsers να μπλοκάρουν τελείως τα popups.

Παράδειγμα 6: resizeTo.html

<button id="openWindow">Open and Resize Window</button>

```

let newWindow;

document.getElementById('openWindow').addEventListener('click', () => {
    // Open a new pop-up window with initial size 300x300
    newWindow = window.open("", "", 'width=300,height=300');

    // Check if the new window was created successfully
    if (newWindow) {
        // Add a button inside the pop-up to resize the window
        newWindow.document.body.innerHTML = '<button
id="resizeWindow">Resize to 500x500</button>';

        // Add an event listener to the button inside the new window

        newWindow.document.getElementById('resizeWindow').addEventListener('click',
        () => {
            newWindow.resizeTo(500, 500); // Resizes the new window to 500x500
pixels
        });
    } else {
        alert('Pop-up blocked! Please allow pop-ups for this site.');
```

Μέθοδοι για Scrolling:

- Με τις παρακάτω μεθόδους μπορούμε να κάνουμε scrolling στη σελίδα:

Μέθοδος	Επεξήγηση
scroll(x,y)	x: Scrolling κατά τον x-άξονα y: Scrolling κατά τον y-άξονα (σε σχέση με την αρχή)
scrollBy(x,y)	x: Scrolling κατά τον x-άξονα y: Scrolling κατά τον y-άξονα (σε σχέση με την τρέχουσα θέση)
scrollTo(x,y)	Ίδια νε την scroll(x,y)

- Εναλλακτικά, μπορούμε να χρησιμοποιήσουμε την εκδοχή των παραπάνω μεθόδων:

Μέθοδος	Επεξήγηση
scroll*(object)	object είναι αντικείμενο με μέλη: top: αντιστοιχεί στον x-άξονα, left: αντιστοιχεί στον y-άξονα, behavior: Παίρνει τιμές 'auto' και 'smooth' (που είναι αισθητικά πιο ευχάριστη)

Παράδειγμα 7: scrollMethod.html

```
<button id="scrollButton">Scroll to (100, 200)</button>
```

```
document.getElementById('scrollButton').addEventListener('click', () => {  
  // Scroll to 100 pixels horizontally and 200 pixels vertically  
  window.scroll(100, 200);  
});
```