

Cours

Programmation avancée pour le web

Credit : 4

Coefficient : 2

Mode d'évaluation :

- Examen (60%) - contrôle continu (40%)

Author:

Chap III HTML5 APIs

L' HTML :

HTML Graphics (HTML5)

L'élément **<canvas>** utilisé pour dessiner du graphique sur une page html moyennant javascript.

<canvas> est un conteneur par défaut sans contour et sans contenu.

```
<!DOCTYPE html>
<html>
<body>
<canvas id="myCanvas" width="200" height="100"
style="border:1px solid #000000;">
Your browser does not support the HTML canvas tag.
</canvas>
</body>
</html>
```

Chap III HTML5 APIs

L' HTML :

HTML Graphics

L'élément `<canvas>` utilisé pour dessiner du graphique sur une page html moyennant javascript.

`<canavas>` est un conteneur par défaut sans contour et sans contenu.



Chap III HTML5 APIs

L' HTML :

HTML Graphics

L'élément **<canvas>** (HTML5) utilisé pour dessiner du graphique sur une page html moyennant javascript.

exemple <script>

```
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.moveTo(0, 0);
ctx.lineTo(200, 100);
ctx.stroke();
</script>
```

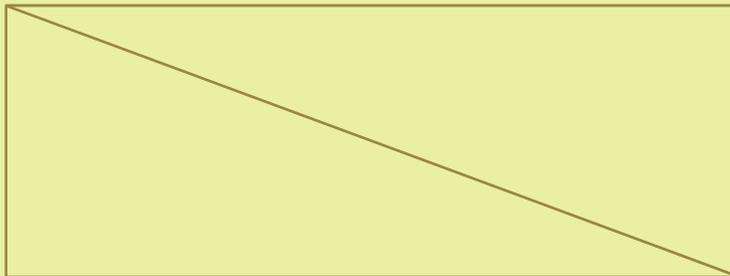
Chap III HTML5 APIs

L' HTML :

HTML Graphics

L'élément **<canvas>** utilisé pour dessiner du graphique sur une page html moyennant javascript.

exemple `<script>`



Chap III HTML5 APIs

L' HTML : HTML APIs

Géolocalisation Cette API permet de repérer la position d'un utilisateur. La méthode à utiliser est : **getCurrentPosition()**.

```
<script>
var x = document.getElementById("demo");
function getLocation() {
  if (navigator.geolocation) {
    navigator.geolocation.getCurrentPosition(showPosition);
  } else {
    x.innerHTML = "Geolocation is not supported by this
browser.";
  }
}
function showPosition(position) {
  x.innerHTML = "Latitude: " + position.coords.latitude +
  "<br>Longitude: " + position.coords.longitude;
}
</script>
```

Chap III HTML5 APIs

L' HTML : HTML APIs

Géolocalisation Cette API permet de repérer la position d'un utilisateur. La méthode à utiliser est :

getCurrentPosition().

Elle renvoie un objet « position » avec les propriétés :

coords.latitude : latitude (décimale renvoyée toujours)

coords.longitude : longitude (décimale renvoyée toujours)

coords.accuracy: accuracy (renvoyée toujours)

coords.altitude: altitude en metres au dessus de la mer (if available)

coords.altitudeAccuracy : altitude accuracy (if available)

coords.heading : en degrés par rapport au nord (if available)

coords.speed : speed (metres/second returned if available)

timestamp :date et heure de la réponse (if available)

Chap III HTML5 APIs

L' HTML : HTML APIs

Drag and drop API glisser-déposer : Cette API permet de déplacer des éléments. La méthode à utiliser est :

```
<!DOCTYPE HTML>
<html>
<head>
<script>
function allowDrop(ev) {
    ev.preventDefault();
}
function drag(ev) {
    ev.dataTransfer.setData("text", ev.target.id);
}
function drop(ev) {
    ev.preventDefault();
    var data = ev.dataTransfer.getData("text");
    ev.target.appendChild(document.getElementById(data));
}
</script>
</head>
<!-- à suivre .... -->
```

Chap III HTML5 APIs

L' HTML : HTML APIs

Drag and drop API glisser-déposer : Cette API permet de déplacer des éléments. La méthode à utiliser est :

```
<!-- la suite ... -->
<body>

<div id="div1" ondrop="drop(event)" ondragover="allowDrop(event)"></div>



</body>
</html>
```