

## TP 2 : Les contrôles serveurs

**Objectif :** Utiliser les contrôles serveur

### **Définition :**

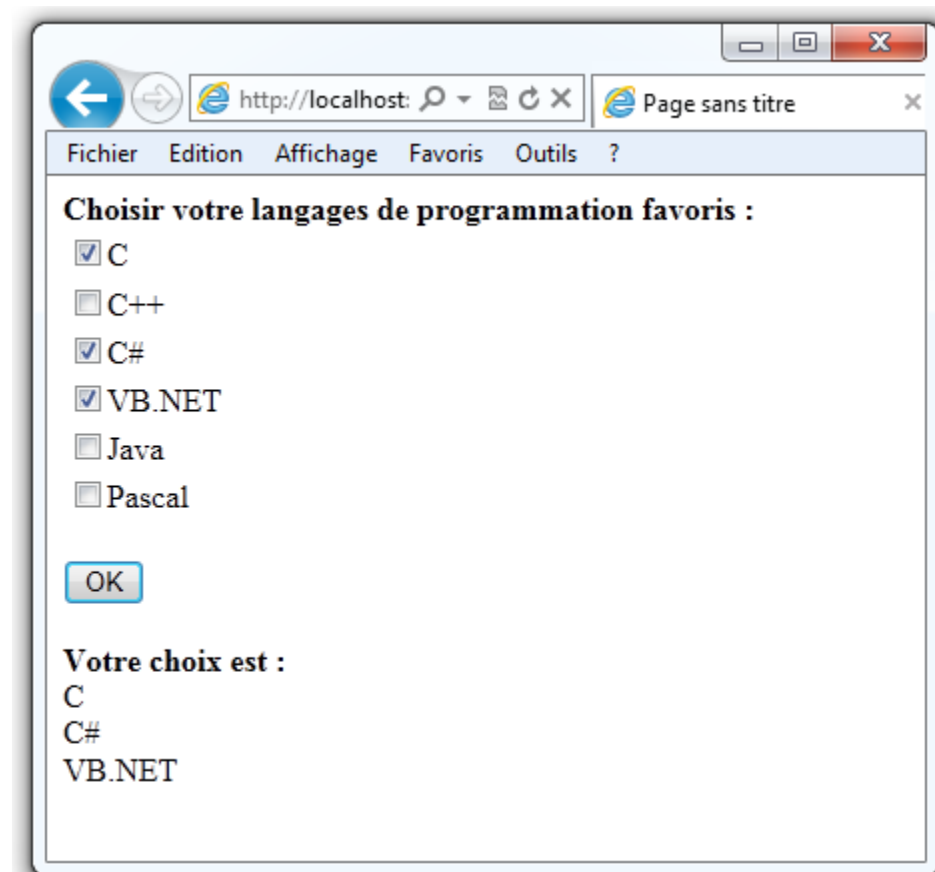
Un contrôle serveur est une balise (HTML ou ASP) associé à la propriété `runat="server"`. A cette balise sera associé un objet lors de l'exécution du code par le serveur, il sera donc instancié au chargement de la page et peut être appelé par le code behind grâce à son `id`. Un contrôle serveur est programmable depuis le code behind pour répondre à des événements (comme un clic, le chargement du contrôle et d'autres).

Les Contrôles sont des classes héritant de la classe Control.

La classe Control contient un attribut appelé « **ViewState** ». C'est un dictionnaire d'état des objets, ainsi tous les états des contrôles sont sauvegardés (dans un champ caché qui est visible dans la source HTML en chiffré). L'intérêt de ASP.NET, à partir de ça, restaure automatiquement les états des contrôles seulement quand la page est rechargée. On appelle ce phénomène le **PostBack**.

### Exercice 1 :

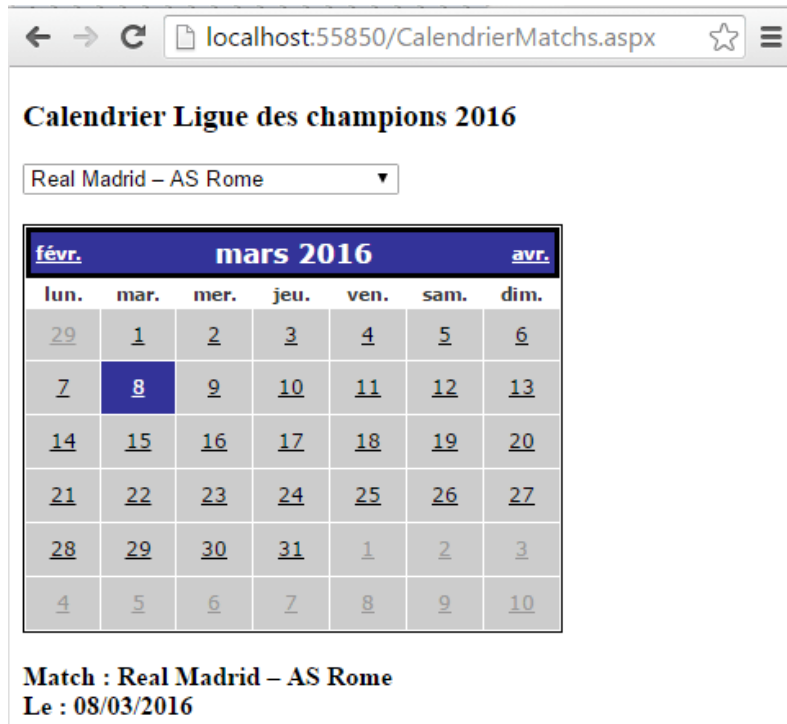
Réaliser l'interface Web suivante :



### Exercice 2 :

L'objectif de cet exercice est de réaliser une page web qui permet la consultation et la modification des dates des matchs d'un championnat en utilisant un calendrier.

- 1) Créer la page web suivante qui contient un contrôle DropDownList **listMatch** pré-remplie avec une liste des matchs, un contrôle Calendar **calDate** et un contrôle Label **lblMatches** pour afficher la date du match sélectionné :



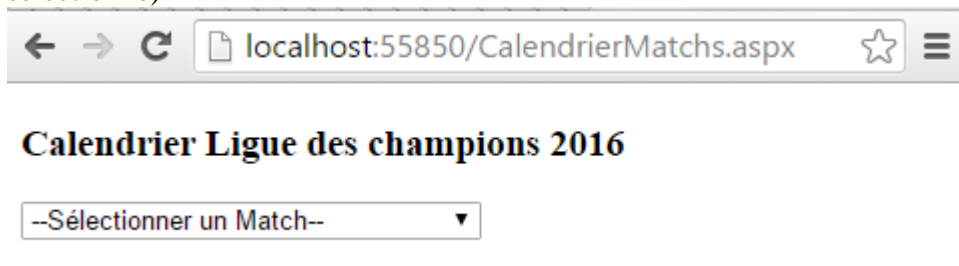
Calendrier Ligue des champions 2016

Real Madrid – AS Rome

mars 2016						
lun.	mar.	mer.	jeu.	ven.	sam.	dim.
29	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

Match : Real Madrid – AS Rome  
Le : 08/03/2016

- 2) Lorsque l'utilisateur sélectionne un match sa date sera sélectionnée dans le calendrier et sera affichée dans le Label (Note : Le calendrier sera caché si aucun match n'est sélectionné)



Calendrier Ligue des champions 2016

--Sélectionner un Match--

- 3) Modifier l'interface pour permettre aussi l'affichage de tous les matchs programmés dans la même date qu'on sélectionne dans le calendrier. L'affichage sera effectué dans un Label et devra prévoir le cas où aucun match n'est programmé.

localhost:55850/CalendrierMatches2.aspx

### Calendrier Ligue des champions 2016

Arsenal – Barcelone

février 2016						
lan.	mar.	mer.	jeu.	ven.	sam.	dim.
25	26	27	28	29	30	31
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	1	2	3	4	5	6

**Matches programmés le 24/02/2016:**  
 - PSV Eindhoven vs Atletico Madrid  
 - Dynamo Kiev vs Manchester City

localhost:55850/CalendrierMatches2.aspx

### Calendrier Ligue des champions 2016

La Gantoise – Wolfsburg

février 2016						
lan.	mar.	mer.	jeu.	ven.	sam.	dim.
25	26	27	28	29	30	31
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	1	2	3	4	5	6

**Aucun match programmé pour cette date**

### Exercice 3 :

L'objectif de cet exercice est de développer l'interface Web suivante :



Cette page Web utilise un contrôle ImageMap **imgMapLog** composé de six zones :

- RectangleHotSpot** : pour naviger dans une nouvelle page (**target="\_blank"**) du site Facebook (**NavigateUrl="http://www.facebook.com"**)
- RectangleHotSpot** : pour ouvrir dans la même page actuelle (**target="\_self"**) le site Twitter (**NavigateUrl="http://www.twitter.com"**)
- CircleHotSpot** : quand l'utilisateur clique sur le premier cercle à gauche un code côté serveur est exécuté pour afficher le texte "WordPress" (**PostBackValue=" WordPress"**) sur un contrôle Label **lblPostBackValue**
- CircleHotSpot** : quand l'utilisateur clique sur le cercle au milieu un code côté serveur est exécuté pour afficher le texte "BMW" (**PostBackValue=" BMW"**) sur le contrôle Label **lblPostBackValue**
- CircleHotSpot** : quand l'utilisateur clique sur le premier cercle à droite un code côté serveur est exécuté pour afficher le texte "Windows" (**PostBackValue=" Windows"**) sur le contrôle Label **lblPostBackValue**

-**PolygonHotSpot** : quand l'utilisateur clique sur cette zone un code côté serveur est exécuté pour afficher le texte "Start" (**PostBackValue=" Start"**) sur le contrôle Label **lblPostBackValue**.

Le contrôle **ImageMap** supporte l'événement **Click** qui permet de capturer les cliques de l'utilisateur sur les différentes zones **HotSpot**. Il reçoit comme argument un objet de type **ImageMapEventArgs** ([ImageMapEventArgs](#) e), cet objet dispose d'une propriété appelée **PostBackValue** (e.PostBackValue) que vous pouvez utiliser dans le code côté serveur lors de la publication.