

# App hybrides

Merci de vous installer  
par binôme de niveau  
différent

# SK

# Programme (1/2)

- Introduction: panorama du dev mobile
- Le web et le HTML
  - Évaluation 1 (QCM)
- La mise en page avec CSS
- Les bases de Javascript
- Les bases de PHP / MySQL
  - Évaluation 2 (Projet)
- Panorama du monde de l'informatique

# Programme (2/2)

- Le HTML / CSS pour le mobile hybride
- Asynchronous Javascript And Xml (Ajax)
  - Angular
  - JQuery
- Découverte de Phonegap Cordova
  - Eclipse
  - xCode
  - Mise en ligne sur les store
- Découverte de iOnic
  - Évaluation finale

- Retour sur le html
- Javascript de base
  - Les variables
  - Structure conditionnelles
  - Les fonctions
- Bonus: La librairie JQuery
- Annexe: Le framework AngularJS

- Insertion du javascript comme avec le CSS:
  - soit à l'intérieur d'une balise spécifique dans le `<head>`:

```
<script type="text/javascript">  
// mon code javascript  
</script>
```

- soit dans un fichier externe:

```
<script type="text/javascript" src="fichier.js"> </script>
```

# Exercice 16: les alert JS

- Insérez le code javascript suivant à l'intérieur de votre page:

```
alert("Hello world");
```

- Résultat attendu: votre page doit afficher une pop-up « Hello World » lorsque vous l'ouvrez (ou la rechargez)

- Variable: élément dont la valeur peut varier en fonction du contexte. Ce « contexte » peut être:
  - des paramètres propre à mon site
  - des informations entrées par l'utilisateur
  - le résultat d'une opération
  - etc..
- Se définit de cette manière:
  - `var ma_variable = "valeur_de_ma_variable";`
  - exemple:  
*`var langueDuSite = "fr";`*  
*`alert("Mon site est en langue"+ langueDuSite);`*  
affichera en pop-up: *Mon site est en langue fr*

# Exercice 17: les variables en JS

- Modifier le code précédent en remplaçant « world » par une variable *mon\_prenom* qui aura pour valeur votre prénom
- Référez-vous à l'exemple de la page précédente
- Résultat attendu: dans mon cas, ma page doit afficher une pop-up « Hello Thibault » lorsque vous l'ouvrez (ou la rechargez)



➤ Permet de rendre son site dynamique. Le site se comportera différemment selon les valeurs prises par le contexte (les variables).

➤ La structure si... alors... sinon...

```
if(condition){  
  // code qui s'exécute si la condition est réunie  
}  
else if(autre condition){  
  // code qui s'exécute si l'autre condition est réunie  
}  
else{  
  // code qui s'exécute si aucune condition n'est réunie  
}
```

# Les structures conditionnelles (2)

## ➤ Exemple

```
var mon_chiffre = "4";  
if(mon_chiffre < 2){  
    alert("mon chiffre est plus petit que 2");  
}  
else if(mon_chiffre > 2){  
    alert("mon chiffre est plus grand que 2");  
}  
else{  
    alert("mon chiffre doit probablement être égal à 2");  
}
```

# Exercice 18: validation du form

- Vérifier lors de l'envoi que votre formulaire comporte bien un message
  - Utiliser la variable propre au champ:  
`document.forms["nomDuFormulaire"]  
["nomDuChamp"].value;`
  - Utiliser une structure conditionnelle pour tester si le champ n'est pas vide `== null ou == ```
  - Afficher un message dans le cas ou le message est vide `alert();`
  - Imbriquez l'ensemble dans une fonction qui sera appelée à l'envoi du formulaire

➤ Les fonctions encapsulent des morceaux de programme qu'on veut exécuter à un moment précis.

Elle se définit sous la forme:

```
fonction nomDeMaFonction(){  
    // code à exécuter  
}
```

➤ Cette fonction s'appelle de cette manière:

```
nomDeMaFonction()
```

Par exemple, *alert()*; est une fonction

# Exercice 18 suite

- Imbriquez l'ensemble dans une fonction qui sera appelée à l'envoi du formulaire

La librairie JQuery <https://jquery.com/>

- Documentation officielle: <http://api.jquery.com/>
- Améliorez la transition de la page index à la page contact avec un de ces plugin jquery:
  - 1: <http://tympanus.net/Development/PageTransitions/>
  - 2: <http://tympanus.net/Development/FullscreenLayoutPageTransitions/>
  - 3: <http://tympanus.net/Tutorials/FullscreenBookBlock/>
  - 4: <http://tympanus.net/Development/DraggableDualViewSlideshow/>

Regardez du côté de:

- La programmation orientée objet
- Les design patterns
- L'architecture MVC

Annexe:

- Le framework AngularJS pour des SPA - applications monopage (adapté au mobile)
  - Tutoriel: <https://docs.angularjs.org/tutorial>
  - Documentation: <https://docs.angularjs.org/api>
- Cordova: application hybride (HTML, CSS et JS transformé en application native) [cordova.apache.org/](http://cordova.apache.org/)
- iOnic (Cordova + Angular): [ionicframework.com/](http://ionicframework.com/)