|  |  |
| --- | --- |
|  |  |
|  | C:\Users\hamadi\AppData\Local\Microsoft\Windows\INetCache\Content.Word\logo_inti_vector-04.jpg  Institut des Nouvelles Technologies  de l’Information |

QCM

RIA

Session 2016

## Guide d'utilisation

* Notez qu'en fonction de la question, il est possible de cocher une ou plusieurs réponses. Ce sera à vous de déterminer le nombre de bonnes réponses possibles.

## Quelques conseils de mise en œuvre

* Ne restez pas bloqué sur une question pour laquelle vous n'avez pas de réponse. Passez à la question suivante. Vous reviendrez par la suite sur les questions sans réponse.
* Identifiez les questions auxquelles vous avez répondu sans certitude. Que ces réponses soient justes ou non, cela vous permettra de mémoriser les questions sur lesquelles vous devrez revenir dans la correction du TEST.
* Ne cherchez pas à aller le plus vite possible. Le temps imparti doit être utilisé au maximum pour vous permettre d'effectuer une synthèse des points abordés dans le module.

**QCM**

**1. Comment sélectionner, avec *jQuery*, les paragraphes se trouvant dans les éléments de classe *el* ?**

* $(‘p > .el’)
* $(‘#el p’)
* $(‘.el, p’)
* $(‘.el p’)

**2. Que doit avoir un composant JSF pour qu'il puisse être affecté par un appel Ajax ?**

* Un attribut id
* Un attribut name
* Un attribut render
* Rien de particulier

**3. Que va faire le code *jQuery* suivant ?**

$(element).append(‘<p>paragraphe</p>’).css({ color: ‘red’, fontSize: 30});

* Ajouter un paragraphe à *element* puis changer la couleur des textes d’*element* en rouge et leur taille à 30 pixels
* Ajouter *element* dans le paragraphe et changer la couleur des textes de celui-ci et leur taille à 30 pixels
* Ajouter à *element* un paragraphe dont le texte sera rouge et la taille de la police à 30px
* Rien de tout cela

**4. Avec un appel ajax JSF, quel attribut faut il rajouter pour mettre à jour un composant ?**

* recall
* render
* target
* component

**5. *jQuery* est-il un standard W3C ?**

* Oui
* Non

**6. *jQuery* est une librairie exécutée...**

* Côté client
* Côté serveur
* Après compilation
* Les trois

**7. Qui développe AngularJs ?**

* Oracle
* Mozilla
* W3C
* Google

**8. Quel est le principe d’*AngularJs* ?**

* Utiliser le javascript côté serveur
* Fournir des outils de manipulation du DOM
* Établir un modèle MVC dont les traitements sont effectués côté client
* Les trois

**9. Comment déclarer une application *AngularJs* ?**

* var application = new Angular(‘appplication’, []);
* angular.newApp(‘application’, []);
* var application = angular.module(‘application’, []);
* var application = angular(‘application’, []);

**10. Dans cette déclaration, à quoi servent les [ ] ?**

* A rien
* A déclarer les autres modules dont l’application dépend
* A déclarer la liste des contrôleurs de l’application
* A déclarer les paramètres de configuration de l’application

**11. Dans *AngularJs*, quelle variable permet d’accéder aux variables d’un contrôleur et des vues qui en dépendent ?**

* scope
* $scope
* $controleur
* Toutes les variables dans le corps du contrôleur sont accessibles dans la vue et inversement

**12. Dans *AngularJs*, quelle variable permet les appels *Ajax* ?**

* $ajax
* $scope
* $http
* angular.ajax

**13. L’*Ajax* d’*AngularJs* est conçu pour faire ses requêtes vers quoi par défaut ?**

* Vers des services SOAP
* Vers n’importe quel serveur
* Vers des Servlets
* Vers des services Rest

**14. Comment déclarer un contrôleur *AngularJs* mini fiable ?**

* .controller(‘dependance1’, ‘dependance2’, function(dependance1, dependance2){ });
* .controller(‘nomControleur’, function(dependance1, dependance2) { });
* .controller(‘nomControleur’, ‘depdance1’, ‘dependance2’, function(dependance1, depdance2) { });
* .controller(‘nomControleur’, [‘depdance1’, ‘dependance2’, function(dependance1,

dependance2) { }]);

**15. Quel attribut permet de faire une boucle dans le *template HTML Angular* ?**

* ng­repeat
* ng­loop
* data­repeat
* data­loop

**16. Dans cet attribut, comment effectuer une boucle sur une liste laListe ?**

* objet : laListe
* objet in laListe
* laListe : objet
* objet from laListe

**17. Comment lier en lecture et écriture la valeur d’un *input* dans la vue à une variable du contrôleur en *AngularJs* ?**

* <input type=”text” value=”{{variable}}” />
* <input type=”text” data­ng­bind=”variable” />
* <input type=”text” data­ng­model=”variable” />
* <input type=”text” data­ng­value=”variable” />

**18. Où les modifications du DOM doivent être effectuées avec *AngularJs* ?**

* N’importe où
* Dans les contrôleurs
* Dans les directives
* Dans les services

**19. Avec JSF, comment utiliser de l’*ajax* au *click* sur un bouton ?**

* <h:commandButton event=”click” action=”#{mb.methode()}” />
* <h:commandButton action=”#{mb.methode()}”><f:ajax event=”click”/></h:commandButton>
* <h:commandButton action=”#{mb.methode()}” type=”ajax” />
* <h:commandButton action=”#{mb.methode()}”></h:commandButton>

**20. Quelles sont deux des librairies de composants JSF majeures ?**

* JavaServerFaces et LimeFaces
* AngularJs et jQuery
* RichFaces et PrimeFaces
* PrimeJsf et LimeFaces

FIN