

Semestre : 1 ☒ 2 ☐

Session : Principale ☒ Rattrapage ☐

Module : Technologies Web 2.0

Enseignant(s) : Equipe Web

Classe(s) : 4INFOB 1->4

Documents autorisés : OUI ☐ NON ☒

Calculatrice autorisée : OUI ☐ NON ☒

Date : 14/11/2019 Heure 11H

Nombre de pages : 5

Internet autorisée : OUI ☐ NON ☒

Durée : 1H

Partie 1 : QCM (7 points)

Veuillez choisir la ou les bonnes réponse(s)

Q1. Quelle est la syntaxe de commentaire dans Twig?

- A. **{# Commentaire #}**
- B. **{{- Commentaire -}}**
- C. **{{# Commentaire #}}**
- D. **{{- Commentaire -}}**
- E. **{[Commentaire]}**
- F. Je ne sais pas

Q2. Dans quel répertoire nous trouverons le fichier « parameters.yml » ?

- A. app/
- B. **app/config/**
- C. config/app/
- D. Aucune de ces réponses n'est vraie
- E. Je ne sais pas

Q3. Dans quel emplacement les ressources (css, javascript, images...) doivent être?

- A. **web/**
- B. app/
- C. **A l'intérieur du bundle**
- D. Dans la racine du projet
- E. Je ne sais pas

Q4. Quelle est l'extension d'un fichier de Template dans Symfony?

- A. .tpl.php
- B. .twig.html
- C. .json.twig
- D. **.html.twig**
- E. Je ne sais pas

Q5. Quelle est la commande pour créer un bundle ?

- A. php bin/console bundle:generate
- B. **php bin/console generate:bundle**
- C. php bin/console generate:create:bundle
- D. php bin/console doctrine:generate:bundle
- E. Je ne sais pas

Q6. Dans Symfony, la configuration des routes peut être écrite en_____?

- A. YAML
- B. PHP
- C. XML
- D. Annotations
- E. Toutes les réponses sont vraies
- F. Je ne sais pas

Q7. Quelle est la commande qui permet de faire la mise à jour d'une base de données après une modification de l'entité ?

- A. php bin/console doctrine:schema:create
- B. php bin/console doctrine:schema:update --force
- C. php bin/console generate:schema:update --force
- D. php bin/console schema:update --force
- E. je ne sais pas

Q8. Est-on obligé de passer par la console pour créer un Contrôleur ?

- A. Oui, pour générer les méthodes et les routes à l'aide du console
- B. Oui, parce que le contrôleur crée des fichiers très difficile de les faire à la main
- C. Non, parce que le contrôleur hérite les méthodes de la part du Controller
- D. Non, parce qu'on peut créer le contrôleur manuellement
- E. Je ne sais pas

Q9. Qu'est-ce que l'architecture MVC ?

- A. Une architecture utilisée par Symfony seulement.
- B. Une architecture pour organiser un code Java
- C. Une architecture pour organiser un code pour n'importe quel langage.
- D. Une architecture pour organiser un code pour le Framework Symfony
- E. Je ne sais pas

Q10. L'objectif de chaque contrôleur de Symfony est de renvoyer un _____?

- A. Objet Response
- B. Objet Request
- C. Objet PHP
- D. Aucune de ces réponses n'est vraie.
- E. Je ne sais pas

Q11. Comment récupérer la valeur du paramètre 'id' en utilisant une requête de type « POST »?

- A. \$request->server->get('id');
- B. \$request->get('id');
- C. \$request->post('id');
- D. \$request->request->get('id');
- E. Aucune de ces réponses n'est vraie
- F. Je ne sais pas

Q12. Convertir le code PHP suivant en Twig.

```
<?php
echo "Nom du produit:" . $product->getName();
?>
```

- A. {{ "Nom du produit:" . product.getName() }}
- B. {{ "Nom du produit:" ~ product.name }}

- C. {{ "Nom du produit:" + product.name }}
- D. Aucune de ces réponses n'est vraie
- E. Je ne sais pas

Q13. Quelle est la syntaxe pour afficher le contenu d'une variable « myvariable » dans une vue twig ?

- A. {{ myvariable.contenu|upper }}
- B. {% myvariable.contenu|upper %}
- C. {{ myvariable (message.contenu) }}
- D. {{ myvariable (\$message.contenu) }}
- E. Je ne sais pas

Q14. Je veux ajouter un attribut 'username' dans l'entité user. Cet attribut n'est pas obligatoire. Quelle est l'annotation à mettre dans l'entité user?

- A.


```
/**
 * @ORM\Column(type="string", length=2, options={"fixed":false,
 "comment": "L'attribut n'est pas obligatoire"})
 */
private $username;
```
- B.


```
/**
 * @ORM\Column(type="string", nullable=true, name="username")
 */
private $ username;
```
- C.


```
/**
 * @ORM\Column(type="string", nullable=true, name="username")
 */
public $ username;
```
- D. Je ne sais pas

Partie 2 (13 Points)

Exercice 1 : Corrigez le code suivant (6pts)

A) Pour supprimer un article

```
$em=$this->getDoctrine()->getManager() ;
$em=$p=$this->getRepository(EspritBundle :article)->findAll()
$em->delete($p) ; $em->remove($p)
$em->flush();
```

B) L'annotation de l'attribut CIN en tant que clé primaire

```
/**
 * @ORM\Column(type="integer")
 * @ORM\CIN Id
 * @ORM\GeneratedValue(strategy="AUTO")
 */
private $CIN;
```

C) Ajouter un article

```
Public function addArticleAction(Request $request)
{
    $article = new Article() ;
    $form = $this->createForm(ArticleType ::class, $article);
    $form->handleRequest($request);
    If($form->isSubmitted()){
        $em=$this->getDoctrine->getManager();
        $em->flush($article); $em->persist();
        $em->persist();$em->flush($article);
        Return new Response
    }
    return $this->render('@Article/Article/addArticle.html.twig' ,
    array("form"=>$form;$form->createView()));
}
```

D) Modifier un article

```
Public function updateArticle updateArticleAction (Request $request, $id)
{
    $article = $this->getDoctrine()->getRepository(Article::class)->find($id) ;
    $form = $this->createForm(ArticleType ::class , $article);
    $form->handleRequest($request);
    If($form->isSubmitted()){
        $em=$this->getDoctrine->getRepository()->getmanger()
        $em->flush($article); $em->persist();
        $em->persist();$em->flush($article);
        return $this->redirectToRoute('anyRoute');
    }
    return $this->render('@Article/Article/editArticle.html.twig' , array("form"=>$form $form-
    >createView()));
}
```

E) Supprimer un article

```
Public function deleteArticleAction(Article $id)
{
    $em=$this->getDoctrine->getManager();
    $em->getRepository('Article'ArticleBundle:Article)->deleteelementbyid($id);->remove($p)
    $em->flush();
    return response ("Article supprimé");
}
```

F) Afficher tous les articles

```
Public function getAllArticlesAction()
{
    $em=$this->getDoctrine->getManager(); getRepository('ArticleBundle:Article)
    $results=$em->myFindAll();
    $em->flush();

    return $this->render('@Article/Article/showArticle.html.twig' , array('results'=>$ results));
}
```

Exercice 2 : Convertir le code PHP suivant en Twig. (3pts)

A)

```
<?php
echo $product["name"];
?>
{{ product["name"] }}
ou
{{ product.name }}
```

B)

```
<?php
if ($max > 20)
{
    echo "J'ai dépassé la valeur 20!"; }
?>
{% if max > 20 %}
    J'ai dépassé la valeur 20!
{% endif %}
```

C)

```
<?php
for ($i=0; $i < count($products); $i++) {
}
?>
{% for i in 0..products|length %}

{% endfor %}
```

Exercice 3 (4pts)

A) Soit la fonction suivante :

Dans le bundle **EspritBundle** et dans le contrôleur **articleController** nous avons défini la fonction suivante :

```
Public function voidAction($id)
{
    return new Response("Affichage de l'article qui a un id=" . $id)
}
```

Ajouter la route nécessaire afin d'afficher le résultat de l'action définie ci-dessus

```
.....void.....:
```

```
path: .../route/{id}.....
```

```
defaults: { _controller: EspritBundle :article :void}
```

B) Quelles sont les étapes à faire pour afficher la liste des articles depuis la base de données dans symfony ?

1-créer une action dans le contrôleur

2- créer un fichier twig

3-configurer une route

C) Après avoir créé un nouveau projet, vous avez généré un Bundle, une erreur s'affiche qui nécessite de faire le **c dump autoload** .

Quelle est cette erreur ?

Le bundle va bien être ajouté au kernel mais il ne pourra pas le charger car il ne sera pas dans l'autoloader