



EXAMEN

Semestre : 1 ☒ 2 ☐

Session : Principale ☒ Rattrapage ☐

Module : Technologies Web2.0

Enseignants : UP Web

Classes : 3A

Documents autorisés : OUI ☒ NON ☐ Nombre de pages : 3

Calculatrice autorisée : OUI ☐ NON ☒ Internet autorisée : OUI ☐ NON ☒

Date : 05/11/2022

Heure : 09h00

Durée : 01h30

1.

- a. .env.....
- b. DATABASE\_URL="mysql://root:@127.0.0.1:3306/qatar2022?"
- c. "doctrine/orm": "^2.13"

2.

Entité Joueur	Entité Vote
<pre>#[ORM\Entity(repositoryClass: JoueurRepository::class)] class Joueur {     #[ORM\Id]     #[ORM\GeneratedValue]     #[ORM\Column]     private ?int \$id = null;      #[ORM\OneToMany(mappedBy: 'joueur', targetEntity: Vote::class)]     private Collection \$votes;</pre>	<pre>#[ORM\Entity(repositoryClass: VoteRepository::class)] class Vote {     #[ORM\Id]     #[ORM\GeneratedValue]     #[ORM\Column]     private ?int \$id = null;      #[ORM\ManyToOne(inversedBy: 'votes')]     private ?Joueur \$joueur = null;</pre>

3.

JoueurController	Joueur/list.html.twig
<pre>#[Route('/joueur', name: 'app_joueur')] public function listJoueur([JoueurRepository \$jeuerepository ]) {     \$joueurs = \$jeuerepository- &gt;findBy(array(),array('nom'=&gt;'ASC'));     return \$this-&gt;render('joueur/index.html.twig', [         'joueurs' =&gt;\$joueurs     ]); }</pre>	<pre>{% for j in joueurs %} &lt;tr&gt;     &lt;td&gt;{{j.nom upper }}&lt;/td&gt;     &lt;td&gt;{{j.equipe}}&lt;/td&gt;     &lt;td&gt;&lt;a href="{{ path('app_vote_byjoueur',{'id':j.id}) }}"&gt;Détails vote&lt;/a&gt;&lt;/td&gt; &lt;/tr&gt; {% endfor %}</pre>

4.

- Le rendu de la requête : Une liste des 10 joueurs Tunisiens Ordonnés par les noms Asc

5.

Ajouter l'attribut « moyenneVote » avec la commande « make :entity » ou directement dans

l'entité Joueur./Créer le fichier de migration avec la commande php bin/console make :migration

Lancer la migration avec la commande php bin/console doctrine :migration :migrate

6.

VoteType	VoteController
<pre>public function buildForm(FormBuilderInterface \$builder, array \$options): void {     \$builder         -&gt;add('noteVote',ChoiceType::class,[             'choices' =&gt; [                 '1' =&gt; 1,                 '2' =&gt; 2,                 '3' =&gt; 3,                 '4' =&gt; 4,                 '5' =&gt; 5],             'multiple'=&gt;false,             'expanded'=&gt;true         ])         -&gt;add('joueur',EntityType::class,[             'class'=&gt;Joueur::class,             'choice_label'=&gt;'nom'</pre>	<pre>#[Route('/joueur', name: 'app_joueur')] public function listJoueur(...) {     ...     \$vote = new Vote();     \$form = \$this- &gt;createForm(VoteType::class,\$vote);     \$form-&gt;handleRequest(\$request);     \$em = \$doctrine-&gt;getManager();     if(\$form-&gt;isSubmitted()) {         \$vote-&gt; setDate (new \DateTime());         //Cette variable représente la note à donner à un         joueur.         \$noteVote = \$vote-&gt;getNoteVote();         // Cette variable représente le joueur à voter.         \$joueurVote = \$vote-&gt;getJoueur();         //Ajouter un vote         \$em-&gt;persist(\$vote);         \$em-&gt;flush();</pre>

<pre> ) -&gt;add('vote',SubmitType::class) ; } </pre>	<pre> <b>Ou \$repositoryVote-&gt;add(\$vote,True) ;</b> //Calculer la moyenne du vote d'un joueur.     \$joueurVote-&gt;setMoyenneVote (<b>\$vr-</b> <b>&gt;getSommeVotebyJoueur(\$votejoueur-&gt;getId())</b> /(\$joueurVote -&gt;getVotes()-&gt;count()));     <b>\$em-&gt;flush();</b>     return \$this-&gt;redirectToRoute('app_joueur'); } return \$this- <b>&gt;renderForm</b>('joueur/index.html.twig', [     ...,     <b>'form'=&gt;\$form</b> ]); } </pre>
---	--

7.

a.

VoteController	VoteRepository
<pre> public function detailVotesParJoueur(<b>VoteRepository</b> <b>\$repository,\$id,JoueurRepository \$repo</b>) {     <b>\$votes = \$repository-&gt;getVotesByJoueur(\$id);</b> <b>\$joueur = \$repo-&gt;find(\$id);</b>     return \$this-&gt;render('vote/votesByJoueur.html.twig', [         ...\$votes,         ...\$joueur     ]); } </pre>	<pre> public function getVotesByJoueur(\$id) : array {     return \$this-     &gt;createQueryBuilder('v')         <b>-&gt;join('v.joueur','j')</b>         <b>-&gt;where('j.id = :id')</b>         <b>-&gt;setParameter('id',\$id)</b>         -&gt;getQuery()         -&gt;getResult(); } </pre>

b.

Joueur/detail.html.twig
<pre> { % if votes is empty % } &lt;h5&gt;Pas de vote&lt;/h5&gt; { % else % } &lt;table class="table-bordered" &gt;     ...     &lt;tr&gt;         &lt;td&gt;{ { v.date <b>date</b> } }&lt;/td&gt;         &lt;td&gt;{ { v.nombreVote } }&lt;/td&gt;     &lt;/tr&gt;     ... &lt;/tbody&gt; &lt;/table&gt; { % endif % } </pre>