

**Discipline** : MDW  
**Matière** : Atelier Framework Côté Serveur  
**Unité d'Ensg.** : Programmation objet avancée  
**Niveau** : 3<sup>ème</sup> Année (Licence appliquée : LMD)

## Workshop 4 : Custom Repository method and Report with Query Builder and Doctrine Query Language

- 1- Create a new controller named : Report  
E:\TP\_JEE\Etudiant\PHP\test>php bin/console make:controller Report

### Using simple query

- 2- Add the following query in BorrowingRepository to find the most borrowing books:

```
public function findMostPopularBooks()
{
    $sql = "SELECT b2.title, COUNT(*) AS howMany
    FROM borrowing b1
    INNER JOIN book b2 ON b2.id=b1.book_id
    GROUP BY b2.title
    ORDER BY howMany DESC";

    $statement = $this->_em->getConnection()->prepare($sql);
    $result = $statement->executeQuery()->fetchAllAssociative();
    return $result;
}
```

You can verify that \_em is defined in ServiceEntityRepository witch extends EntityRepository

- 3- Call the method in ReportController
  - a. Add the repository to index method and Import it

<pre>&lt;?php  namespace App\Controller;  use Symfony\Bundle\FrameworkBundle\Controller; use Symfony\Component\HttpFoundation\Response; use Symfony\Component\Routing\Annotation\Route;  class ReportController extends AbstractController {     #[Route('/report', name: 'app_report')]     public function index(BorrowingRepository \$repository)     {         return \$this-&gt;render('report/index.html', [             'controller_name' =&gt; 'ReportCont         ]);     } }</pre>	<table border="1"><tr><td>Atteindre la définition</td><td>F12</td></tr><tr><td>Atteindre les références</td><td>Maj+F12</td></tr><tr><td>Aperçu</td><td>&gt;</td></tr><tr><td>Import Class</td><td>Ctrl+Alt+I</td></tr><tr><td>Import All Classes</td><td>Ctrl+Alt+A</td></tr><tr><td>Expand Class</td><td></td></tr><tr><td>Sort Imports</td><td>Ctrl+Alt+S</td></tr><tr><td>Highlight Not Imported Classes</td><td>Ctrl+Alt+N</td></tr><tr><td>Highlight Not Used Classes</td><td>Ctrl+Alt+U</td></tr><tr><td>Generate namespace for this file</td><td>Ctrl+Alt+G</td></tr><tr><td>Kite: Find Related Code From Line</td><td></td></tr><tr><td>Rechercher toutes les références</td><td>Maj+Alt+F12</td></tr><tr><td>Modifier toutes les occurrences</td><td>Ctrl+F2</td></tr><tr><td>Mettre en forme le document avec...</td><td></td></tr><tr><td>Mettre le document en forme</td><td>Maj+Alt+F</td></tr></table>	Atteindre la définition	F12	Atteindre les références	Maj+F12	Aperçu	>	Import Class	Ctrl+Alt+I	Import All Classes	Ctrl+Alt+A	Expand Class		Sort Imports	Ctrl+Alt+S	Highlight Not Imported Classes	Ctrl+Alt+N	Highlight Not Used Classes	Ctrl+Alt+U	Generate namespace for this file	Ctrl+Alt+G	Kite: Find Related Code From Line		Rechercher toutes les références	Maj+Alt+F12	Modifier toutes les occurrences	Ctrl+F2	Mettre en forme le document avec...		Mettre le document en forme	Maj+Alt+F
Atteindre la définition	F12																														
Atteindre les références	Maj+F12																														
Aperçu	>																														
Import Class	Ctrl+Alt+I																														
Import All Classes	Ctrl+Alt+A																														
Expand Class																															
Sort Imports	Ctrl+Alt+S																														
Highlight Not Imported Classes	Ctrl+Alt+N																														
Highlight Not Used Classes	Ctrl+Alt+U																														
Generate namespace for this file	Ctrl+Alt+G																														
Kite: Find Related Code From Line																															
Rechercher toutes les références	Maj+Alt+F12																														
Modifier toutes les occurrences	Ctrl+F2																														
Mettre en forme le document avec...																															
Mettre le document en forme	Maj+Alt+F																														

b. Add the definition of \$books proprieties

```
$books = $repository->findMostPopularBooks();  
return $this->render('report/index.html.twig', [  
    'books' => $books,  
]);
```

4- Customize the template

```
{% extends 'base.html.twig' %}  
  
{% block title %}Most popular books{% endblock %}  
  
{% block body %}  
<h1>The most popular books</h1>  
<table class="table table-striped">  
<tr>  
    <th>Title</th>  
    <th>Borrowed</th>  
</tr>  
    {% for book in books %}  
        <tr><td>{{ book.title }}</td><td>{{ book.howMany }}</td></tr>  
    {% endfor %}  
</table>  
{% endblock %}
```

5- Change the name root

```
#[Route('/most-popular-book', name: 'most_popular_book')]
```

Result:

127.0.0.1:8000/most-popular-book

Mohamed

UVT: Tous les cours

Google

DuckDuckGo — Co...

Ecole Ouverte des

Librarian

Borrowing

Books

Students ▾

Logout

The most popular books

Title	Borrowed
the book	3
Symfony	1

## Using QueryBuilder

- a. Add new request builder in BorrowinRepository

```
public function findMostPopularBooksQb()
{
    return $this->createQueryBuilder('b')
        ->addSelect('bk.title, COUNT(b) AS howMany')
        ->join('b.book', 'bk')
        ->groupBy('bk.title')
        ->orderBy('howMany', 'DESC')
        ->getQuery()
        ->getResult();
}
```

- b. Add now the new definition of \$books proprieties in ReportController

```
$books = $repository->findMostPopularBooksQb();
```

The result is the same

## Using Doctrine Query Language

- a. Add new function named findMostPopularBooksDql() in BorrowinRepository

```
public function findMostPopularBooksDql()
{
    $query = $this->_em->createQuery('SELECT bk.title, COUNT(b) AS howMany
                                     FROM App\Entity\Borrowing b
                                     JOIN b.book bk
                                     GROUP BY bk.title
                                     ORDER BY howMany DESC');

    $borrowings = $query->getResult();
    return $borrowings;
}
```

- b. Add now the new definition of \$books proprieties in ReportController

```
$books = $repository->findMostPopularBooksDql();
```

The result is the same

## Search by book

1. In order to search for all borrowings of a given book, add BookSearch in App\Entity

```
<?php
namespace App\Entity;
```

```

use Doctrine\ORM\Mapping as ORM;

class BookSearch
{
    private $book;
    public function getBook(): ?Book
    {
        return $this->book;
    }
    public function setBook(?Book $book): self
    {
        $this->book = $book;
        return $this;
    }
}

```

2. Create then in the Form section, BookSearchType class which is a builder for all books

```

<?php
namespace App\Form;

use App\Entity\Book;
use App\Entity\BookSearch;
use Symfony\Component\Form\AbstractType;
use Symfony\Component\Form\FormBuilderInterface;
use Symfony\Component\OptionsResolver\OptionsResolver;
use Symfony\Bridge\Doctrine\Form\Type\EntityType;

class BookSearchType extends AbstractType
{
    public function buildForm(FormBuilderInterface $builder, array $options): void
    {
        $builder
            ->add('book',EntityType::class,['class' => Book::class,
                'choice_label' => 'title' ,
                'label' => 'Book' ]);
    }
    public function configureOptions(OptionsResolver $resolver): void
    {
        $resolver->setDefaults([
            'data_class' => BookSearch::class,
        ]);
    }
}

```

```
}
```

3. For example add the root `/BorrowingBook` that will create an instance of `BookSearch` and will call `BookSearchType` to select all borrowings of a given book in a form

```
public function BorrowingBook(Request $request, BorrowingRepository $repository) {
    $bookSearch = new BookSearch();
    $form = $this->createForm(BookSearchType::class,$bookSearch);
    $form->handleRequest($request);
    $borrowings= [];

    if($form->isSubmitted() && $form->isValid()) {
        $book = $bookSearch->getBook();
        if ($book!="")
            $borrowings=$repository->findBy( array('book' => $book) );
        else
            $borrowings= $repository->findAll();
    }

    return $this->render('report/BorrowingBook.html.twig',
        ['form' => $form->createView(),'borrowings' => $borrowings]);
}
```

Librarian Borrowing Books Students [My compte \(123456\)](#) [Logout](#)

Book the book ▼

the book

Symfony

123

the sun

Search

Student surname	Date borrowed	Book returned
admin	2017-01-01	No
sss	2021-01-01	No
sss	2022-01-01	No

back to list