**Cách chạy project:**

php app/console assets:install

php app/console assetic:dump

**Xóa cache: app/cache: xóa 2 folder dev và prod**

**Chuyển từ String -> Date time (PHP)**

new \DateTime(date('H:i:s',strtotime($request->get("start\_time"))))

**Chuyển từ DateTime -> String (PHP)**

date\_format($c["startTime"], 'H:i:s');

**Dịnh kiểu DateTime (Twig)**

{{ item,startDate|date(‘d M Y’) }} // 09 Nov 2018

l: Monday , Tuesday

**Chang format date JS**

var dd = ("0" + (date.getDate())).slice(-2);

var mm =("0" + (date.getMonth() + 1)).slice(-2); //January is 0!

**Change string to date and calculate date (PHP)**

$time = strtotime($date);

$now = date('Y-m-d',$time);

$from = strtotime(date("Y-m-d", strtotime($now)) . " -1 month");

$from = strftime("%Y-%m-%d", $from);

**Lấy ngày đâu của tuần hiện tại:**

var curr = new Date; // get current date

var first = curr.getDate() - curr.getDay();

var last = = curr.getDate() - curr.getDay() + 6 ;

var firstdate = new Date(curr.setDate(first));

**Các hàm datetime trong php :**

<http://php.net/manual/ru/class.datetime.php>

**Period Time:**

$startTime = $doctorTimeslot->getStartTime();

$endTime = $doctorTimeslot->getEndTime();

$endTime = $endTime->modify( '+1 hour' );

$spec = 'PT1H'; // distance 1h

$interval = new \DateInterval($spec);

$period = new \DatePeriod($startTime, $interval, $endTime);

$arrPeriod[] = $period;

**datepicker**

$('.optionDate').datepicker()

.on('changeDate', function(event)

DocTrine giữ liên kết

$em = $this->getEntityManager();

$repo = $em->getRepository('UtilBundle:Appointment');

$result = $repo->createQueryBuilder('ap')

->where('ap.status = :status')

->setParameter('status', $status);

$q = clone $result;

$q->setFirstResult($startRecord)

->setMaxResults($perPage);

$totalResult = count($result->getQuery()->getResult());

$result = $q->getQuery()->getResult();

**//Phân trang**

Note:

**-reponsitory:**

$perPage = ($params['perPage'] > 0 || $params['perPage'] == -1)? $params['perPage']: Constant::PER\_PAGE\_DEFAULT;

if($perPage != -1) {

$q->setFirstResult($startRecord)

->setMaxResults($perPage);

}

**-Controller:**

if($request->get('appointment\_fill') == -1 ) {

$params['perPage'] = $totalResult;

}

'all' => $request->get('appointment\_fill',0)

**HTML:**

{% if all == -1 %}

{% set currentPage = 0 %}

{% endif %}

**Doctrine**:

$repo = $em->getRepository('UtilBundle:Appointment');

$result = $repo->createQueryBuilder('ap')

->innerJoin('UtilBundle:Patient','pa','WITH','pa.id = ap.patient')

->innerJoin('UtilBundle:PersonalInformation','per','WITH','per.id = pa.personalInformation')

->where('ap.status = :status')->setParameter('status', $status);

//Phân trang

$q = clone $result;

if($perPage != -1) {

$q->setFirstResult($startRecord)

->setMaxResults($perPage);

}

$totalResult = count($result->getQuery()->getResult());

//Data

$result = $q->getQuery()->getResult();

**Update schema:**

Php app/console doctrine:generate:entities UtilBundle:Product

Php app/console doctrine:schema:update –force (update db)

**$qb->select('agent.id, per.firstName, per.lastName, agent.agentCode, afm.agentFee, afm.areaType, afm.takeEffectOn')**

**->from('UtilBundle:AgentFeeMedicine', 'afm')**

**->innerJoin('afm.agent', 'agent')**

**->innerJoin('agent.personalInformation', 'per')**

**->where('afm.agentFee <> IFELSE(afm.areaType = 1, :localFee, :overseaFee)')**

**->andWhere('agent.isActive = 1')**

**->andWhere('agent.deletedOn is NULL')**

**->andWhere('afm.isActive = 1')**

**->setParameter('localFee', $params['localFee'])**

**->setParameter('overseaFee', $params['overseaFee'])**

**->groupBy('afm.agent');**

**Tạo Entity**

**php app\console doctrine:mapping:import UtilBundle annotation --filter=DrugAudit**

**php app\console doctrine:generate:entities UtilBundle/Entity/DrugAudit --no-backup --path=src/Core**

**sửa cái DrugAudit lại theo tên entity em muốn tạo.**

**Send form type:file to controller from js**

**form\_data.append('mainLogo',$('#staff\_admin\_mainLogo')[0].files[0]);**

**replace id from list**

**<input type='hidden' value="{{ path('admin\_staff\_edit',{'id':'id'})}}" id='edit-path' />**

**var path = $("#edit-path").val();**

**var curPath = path.replace('id', data[i].id);**

**redirect Back page**

**window.history.back();**

**html file**

**<div id="upload\_wa" style="padding-top:20px" >**

**<div class="upload-btn-wrapper" style="position:relative;overflow:hidden;display: inline-block">**

**<button class="btn-upload" style="border: 2px solid gray;color: gray;background-color: white;padding: 8px 20px;border-radius: 8px;font-weight: bold;">Upload photo to rx</button>**

**<input style="font-size: 100px;position: absolute;left: 0;top: 0; opacity: 0;" type="file" name="myfile" id="myfile">**

**</div>**

**</div>**

**Ajax submit input:file**

**form\_data = new FormData();**

**form\_data.append('myfile',$('input[name=new-reference-rx]')[0].files[0]);**

**$.ajax({**

**type:"POST",**

**data: form\_data,**

**url:url,**

**contentType: false,**

**processData: false,**

**success:function(res){**

**console.log('aaaaa');**

**},**

**});**

**Sử dụng biến ngoài ajax**

**async: false,**

**Invalid image type in Jquery**

**Import:**

**'@AdminBundle/Resources/public/assets/global/plugins/jquery-validation/js/jquery.validate.min.js'**

**'@AdminBundle/Resources/public/assets/global/plugins/jquery-validation/js/additional-methods.min.js'**

**$('#myfile').change(function(){**

**var reader = new FileReader();**

**var fileExtension = ['image/jpeg', 'application/pdf', 'image/png', 'image/gif'];**

**if($.inArray($('#myfile')[0].files[0].type, fileExtension) == -1){**

**$('#img-error').html('Invalid Format Type. Please Check Again');**

**$('#img-error').attr('style','color:red');**

**return false;**

**} else {**

**$('#img-error').html('');**

**}**

**html = '';**

**reader.onload = function (e) {**

**if($('#myfile')[0].files[0].type == 'application/pdf') {**

**$('#img\_pdf').attr('src',e.target.result);**

**$('#img\_pdf').attr('style','display:block;padding-bottom:10px');**

**$('#img\_file').hide();**

**} else {**

**$('#img\_file').attr('src', e.target.result);**

**$('#img\_file').show();**

**$('#img\_pdf').hide();**

**}**

**}**

**reader.readAsDataURL($('#myfile')[0].files[0]);**

**$('#img\_file').show();**

**})**

**Check undefined JS**

**typeof data !== 'undefined'**

**Autocomplete**

**jsCommon.suggestion(url\_doctor, data\_doctor, 'text');**

**jsCommon.suggestion(url, data\_patient, 'text', $('.on-suggestion-doctor'));**

**suggestion: function(url, data, display\_name, $el) {**

**var tag = $el || $('.on-suggestion');**

**if(tag.length > 0) {**

**data = typeof data !== 'undefined' ? data : {};**

**display\_name = typeof display\_name !== 'undefined' ? display\_name : 'name';**

**var sources = function (term, sync) {**

**data['term'] = term;**

**$.ajax({**

**url: url,**

**data: data,**

**dataType: "json",**

**async: false,**

**type: "POST",**

**success: function (res) {**

**return sync(res);**

**}**

**});**

**};**

**tag.typeahead({**

**highlight: false,**

**hint: true,**

**minLength: 3**

**}, {**

**display: display\_name,**

**source: sources**

**}).on('keyup', this, function (e) {**

**if (e.keyCode == 13) {**

**tag.typeahead('close');**

**}**

**});**

**}**

**},**

**Show scroll horizon table html:**

table {

display: block;

overflow-x: auto;

white-space: nowrap;

}