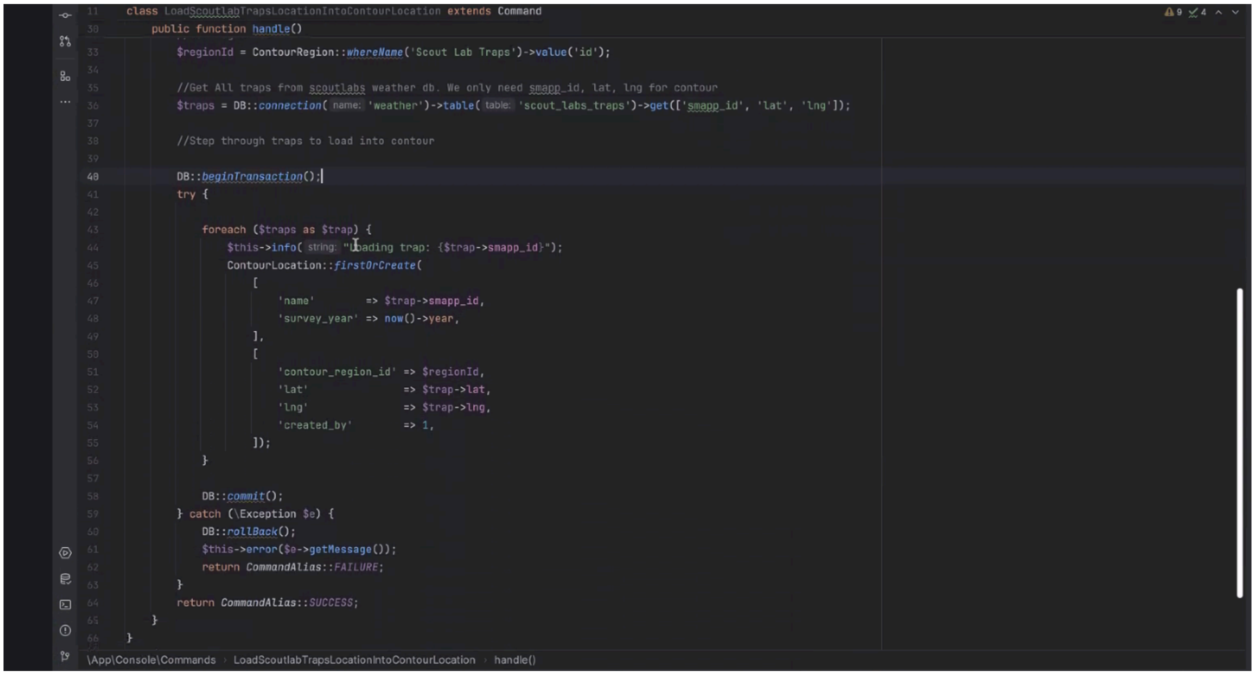
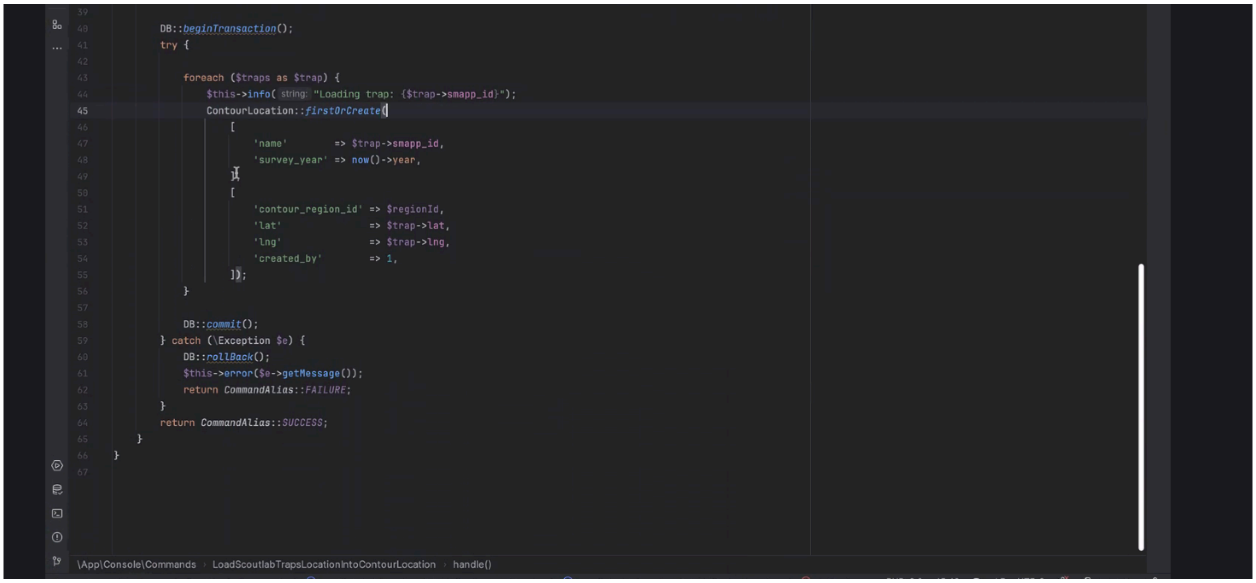
don't want sum of the week, want trap_count column



```php
class LoadScoutlabTrapsLocationIntoContourLocation extends Command
    public function handle()
        $regionId = ContourRegion::whereName('Scout Lab Traps')->value('id');

        //Get All traps from scoutlabs weather db. We only need smapp_id, lat, lng for contour
        $traps = DB::connection('weather')->table('scout_labs_traps')->get(['smapp_id', 'lat', 'lng']);

        //Step through traps to load into contour

        DB::beginTransaction();
        try {

            foreach ($traps as $trap) {
                $this->info("Loading trap: {$trap->smapp_id}");
                ContourLocation::firstOrCreate(
                    [
                        'name'        => $trap->smapp_id,
                        'survey_year' => now()->year,
                    ],
                    [
                        'contour_region_id' => $regionId,
                        'lat'               => $trap->lat,
                        'lng'               => $trap->lng,
                        'created_by'        => 1,
                    ]);
            }

            DB::commit();
        } catch (\Exception $e) {
            DB::rollBack();
            $this->error($e->getMessage());
            return CommandAlias::FAILURE;
        }
        return CommandAlias::SUCCESS;
    }
}
```

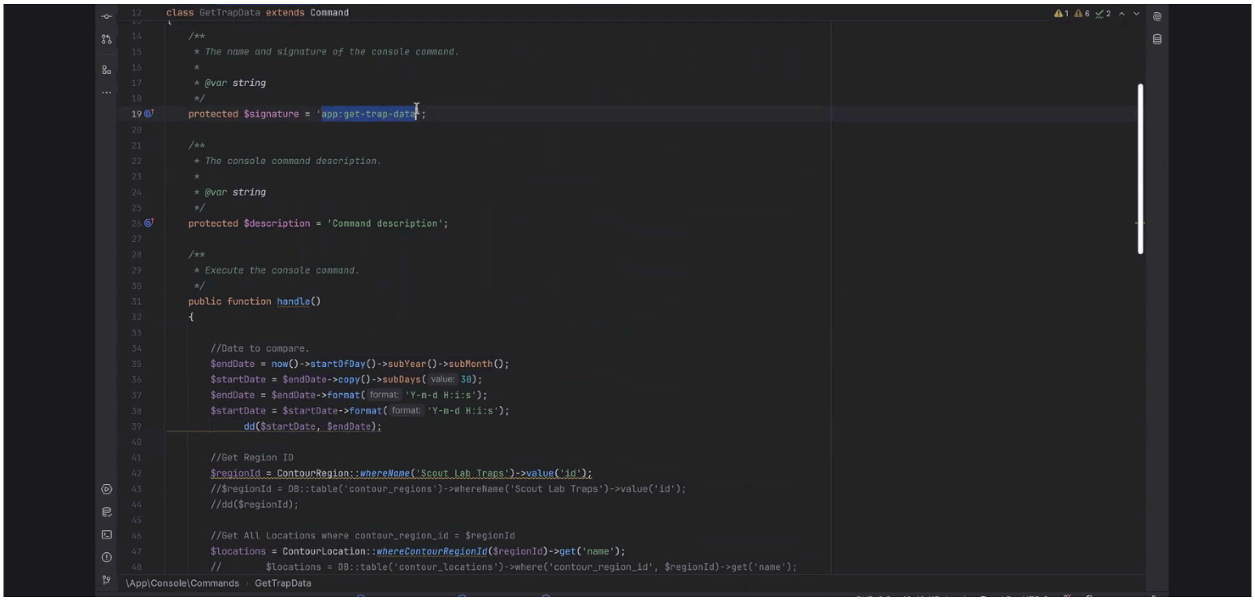
\App\Console\Commands › LoadScoutlabTrapsLocationIntoContourLocation › handle()



```php
        DB::beginTransaction();
        try {

            foreach ($traps as $trap) {
                $this->info("Loading trap: {$trap->smapp_id}");
                ContourLocation::firstOrCreate(
                    [
                        'name'        => $trap->smapp_id,
                        'survey_year' => now()->year,
                    ],
                    [
                        'contour_region_id' => $regionId,
                        'lat'               => $trap->lat,
                        'lng'               => $trap->lng,
                        'created_by'        => 1,
                    ]);
            }

            DB::commit();
        } catch (\Exception $e) {
            DB::rollBack();
            $this->error($e->getMessage());
            return CommandAlias::FAILURE;
        }
        return CommandAlias::SUCCESS;
    }
}
```

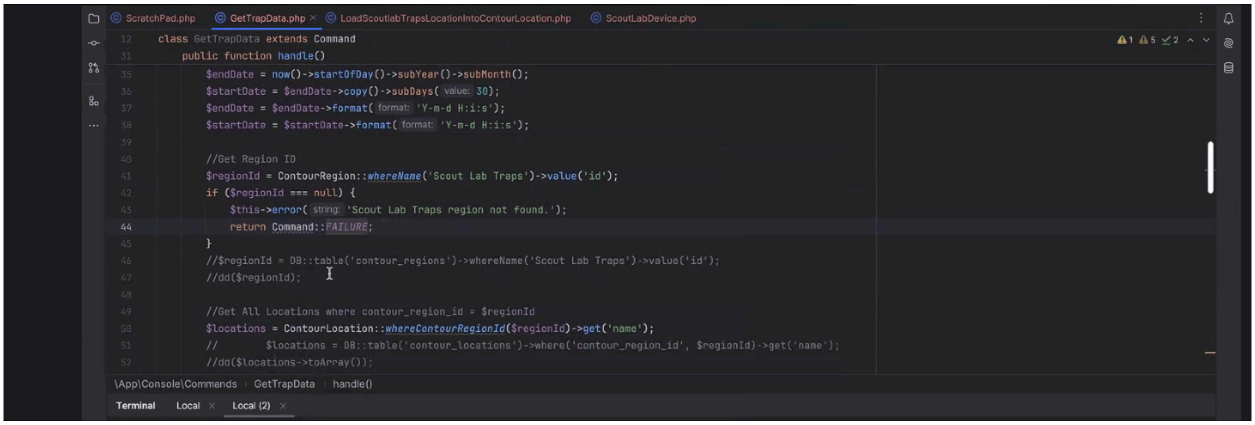
\App\Console\Commands › LoadScoutlabTrapsLocationIntoContourLocation › handle()

Get 30 days:



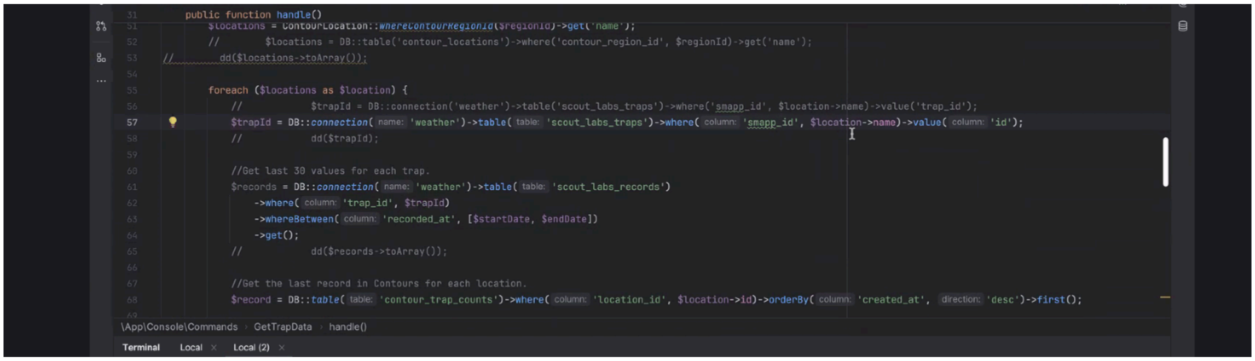
```
class GetTrapData extends Command
    /**
     * The name and signature of the console command.
     *
     * @var string
     */
    protected $signature = 'app:get-trap-data';

    /**
     * The console command description.
     *
     * @var string
     */
    protected $description = 'Command description';

    /**
     * Execute the console command.
     */
    public function handle()
    {

        //Date to compare.
        $endDate = now()->startOfDay()->subYear()->subMonth();
        $startDate = $endDate->copy()->subDays( value: 30);
        $endDate = $endDate->format( format: 'Y-m-d H:i:s');
        $startDate = $startDate->format( format: 'Y-m-d H:i:s');
              dd($startDate, $endDate);

        //Get Region ID
        $regionId = ContourRegion::whereName('Scout Lab Traps')->value('id');
        //$regionId = DB::table('contour_regions')->whereName('Scout Lab Traps')->value('id');
        //dd($regionId);

        //Get All Locations where contour_region_id = $regionId
        $locations = ContourLocation::whereContourRegionId($regionId)->get('name');
        //        $locations = DB::table('contour_locations')->where('contour_region_id', $regionId)->get('name');
```

\App\Console\Commands › GetTrapData



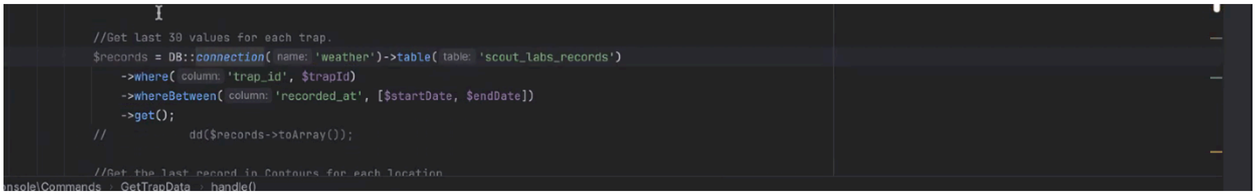
```
class GetTrapData extends Command
    public function handle()
        $endDate = now()->startOfDay()->subYear()->subMonth();
        $startDate = $endDate->copy()->subDays( value: 30);
        $endDate = $endDate->format( format: 'Y-m-d H:i:s');
        $startDate = $startDate->format( format: 'Y-m-d H:i:s');

        //Get Region ID
        $regionId = ContourRegion::whereName('Scout Lab Traps')->value('id');
        if ($regionId === null) {
            $this->error( string: 'Scout Lab Traps region not found.');
            return Command::FAILURE;
        }
        //$regionId = DB::table('contour_regions')->whereName('Scout Lab Traps')->value('id');
        //dd($regionId);

        //Get All Locations where contour_region_id = $regionId
        $locations = ContourLocation::whereContourRegionId($regionId)->get('name');
        //        $locations = DB::table('contour_locations')->where('contour_region_id', $regionId)->get('name');
        //dd($locations->toArray());
```

\App\Console\Commands › GetTrapData › handle()



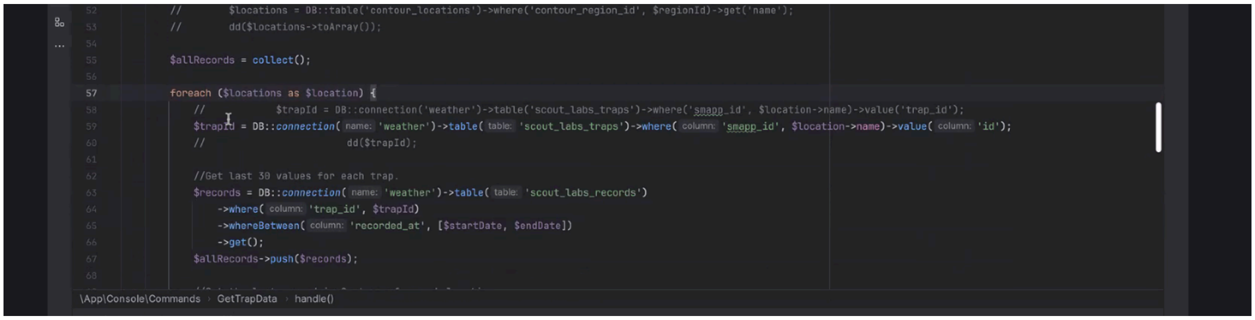
```
    public function handle()
        $locations = ContourLocation::whereContourRegionId($regionId)->get('name');
        //        $locations = DB::table('contour_locations')->where('contour_region_id', $regionId)->get('name');
//        dd($locations->toArray());

        foreach ($locations as $location) {
            //            $trapId = DB::connection('weather')->table('scout_labs_traps')->where('smapp_id', $location->name)->value('trap_id');
            $trapId = DB::connection( name: 'weather')->table( table: 'scout_labs_traps')->where( column: 'smapp_id', $location->name)->value( column: 'id');
            //            dd($trapId);

            //Get last 30 values for each trap.
            $records = DB::connection( name: 'weather')->table( table: 'scout_labs_records')
                ->where( column: 'trap_id', $trapId)
                ->whereBetween( column: 'recorded_at', [$startDate, $endDate])
                ->get();
            //            dd($records->toArray());

            //Get the last record in Contours for each location.
            $record = DB::table( table: 'contour_trap_counts')->where( column: 'location_id', $location->id)->orderBy( column: 'created_at', direction: 'desc')->first();
```

\App\Console\Commands › GetTrapData › handle()



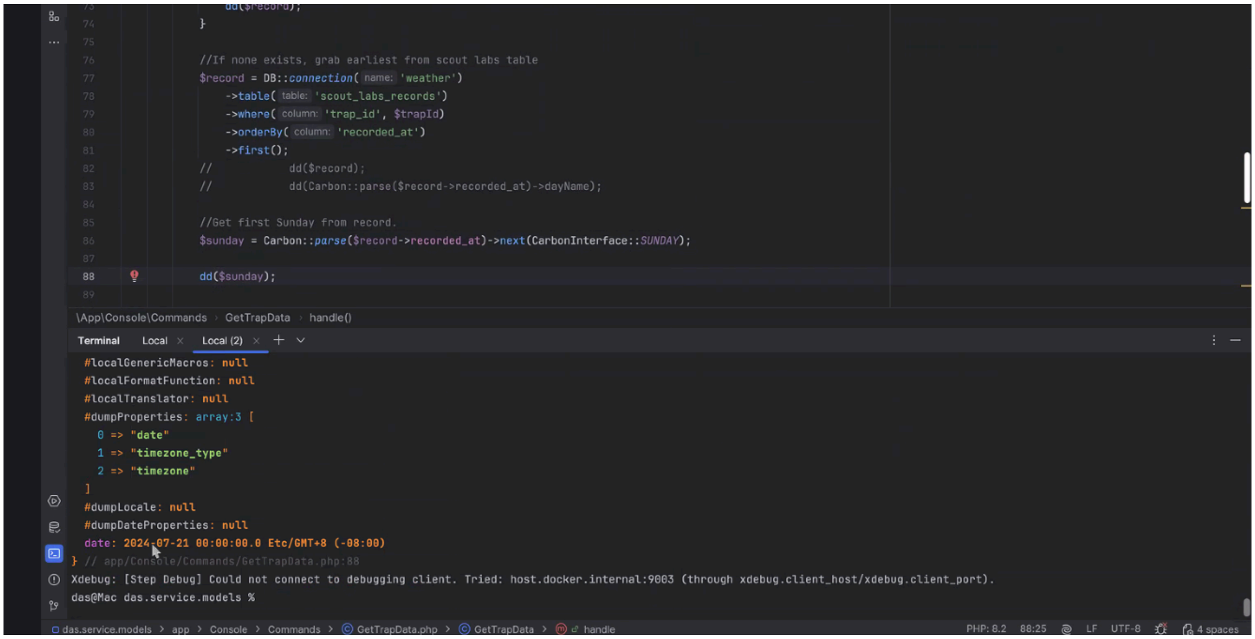
```
//Get last 30 values for each trap.
$records = DB::connection( name: 'weather')->table( table: 'scout_labs_records')
    ->where( column: 'trap_id', $trapId)
    ->whereBetween( column: 'recorded_at', [$startDate, $endDate])
    ->get();
//            dd($records->toArray());

//Get the last record in Contours for each location.
```

nsole\Commands › GetTrapData › handle()



```php
//        $locations = DB::table('contour_locations')->where('contour_region_id', $regionId)->get('name');
//        dd($locations->toArray());

$allRecords = collect();

foreach ($locations as $location) {
    //            $trapId = DB::connection('weather')->table('scout_labs_traps')->where('smapp_id', $location->name)->value('trap_id');
    $trapId = DB::connection('weather')->table('scout_labs_traps')->where('smapp_id', $location->name)->value('id');
    //                        dd($trapId);

    //Get last 30 values for each trap.
    $records = DB::connection('weather')->table('scout_labs_records')
        ->where('trap_id', $trapId)
        ->whereBetween('recorded_at', [$startDate, $endDate])
        ->get();
    $allRecords->push($records);
```

\App\Console\Commands › GetTrapData › handle()



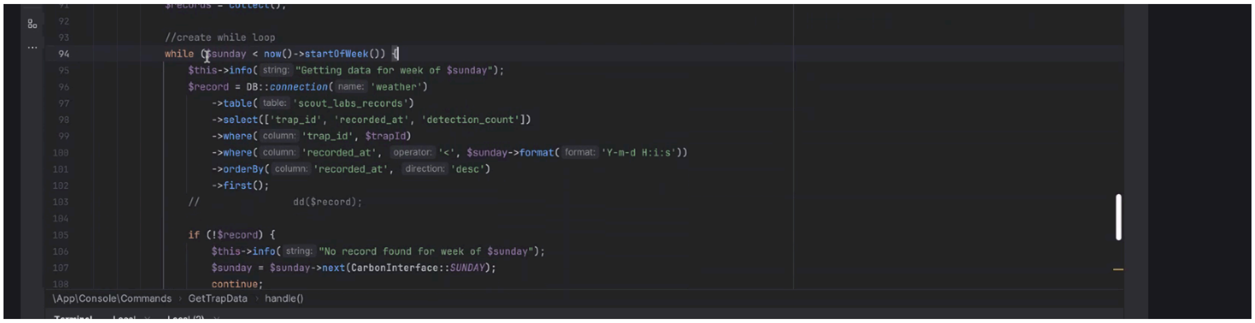
```php
        dd($record);
    }

    //If none exists, grab earliest from scout labs table
    $record = DB::connection('weather')
        ->table('scout_labs_records')
        ->where('trap_id', $trapId)
        ->orderBy('recorded_at')
        ->first();
//            dd($record);
//            dd(Carbon::parse($record->recorded_at)->dayName);

    //Get first Sunday from record.
    $sunday = Carbon::parse($record->recorded_at)->next(CarbonInterface::SUNDAY);

    dd($sunday);
```

\App\Console\Commands › GetTrapData › handle()
Terminal
Local
Local (2)

```
  #localGenericMacros: null
  #localFormatFunction: null
  #localTranslator: null
  #dumpProperties: array:3 [
    0 => "date"
    1 => "timezone_type"
    2 => "timezone"
  ]
  #dumpLocale: null
  #dumpDateProperties: null
  date: 2024-07-21 00:00:00.0 Etc/GMT+8 (-08:00)
} // app/Console/Commands/GetTrapData.php:88
Xdebug: [Step Debug] Could not connect to debugging client. Tried: host.docker.internal:9003 (through xdebug.client_host/xdebug.client_port).
das@Mac das.service.models %
```

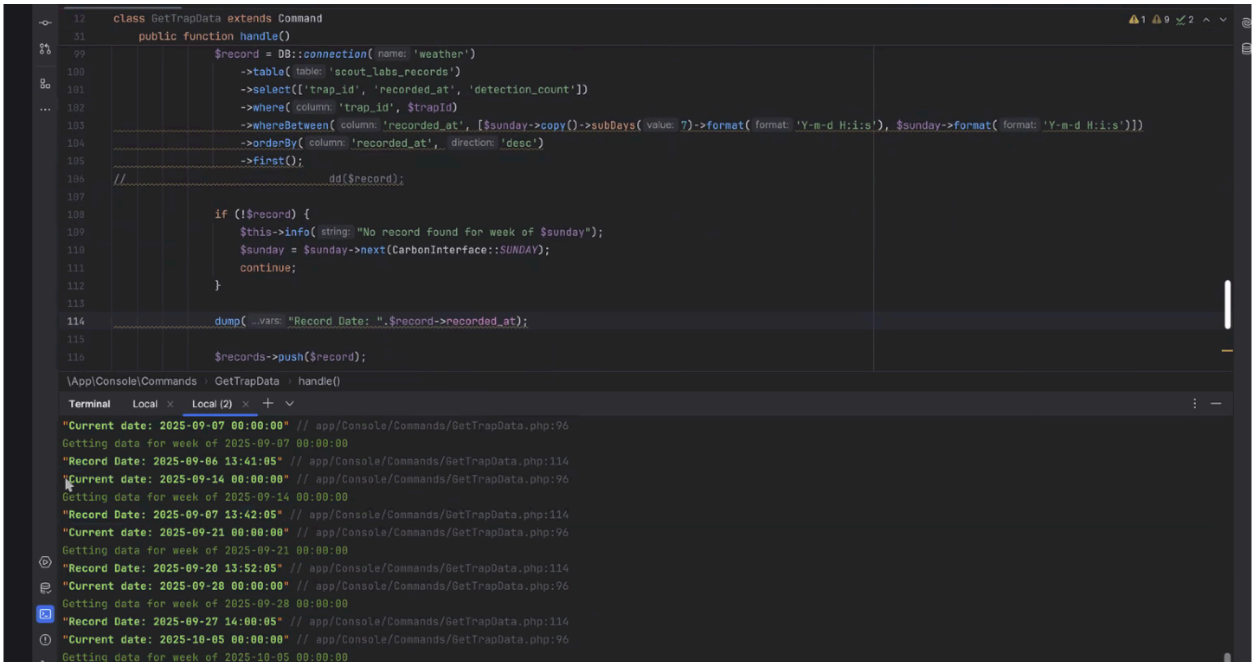
das.service.models › app › Console › Commands › GetTrapData.php › GetTrapData › handle
PHP: 8.2
88:25
LF
UTF-8
4 spaces

Would never do this since it’s very unsafe, but gives the general idea:



```php
$records = collect();

//create while loop
while ($sunday < now()->startOfWeek()) {
    $this->info("Getting data for week of $sunday");
    $record = DB::connection('weather')
        ->table('scout_labs_records')
        ->select(['trap_id', 'recorded_at', 'detection_count'])
        ->where('trap_id', $trapId)
        ->where('recorded_at', '<', $sunday->format('Y-m-d H:i:s'))
        ->orderBy('recorded_at', 'desc')
        ->first();
//                dd($record);

    if (!$record) {
        $this->info("No record found for week of $sunday");
        $sunday = $sunday->next(CarbonInterface::SUNDAY);
        continue;
```

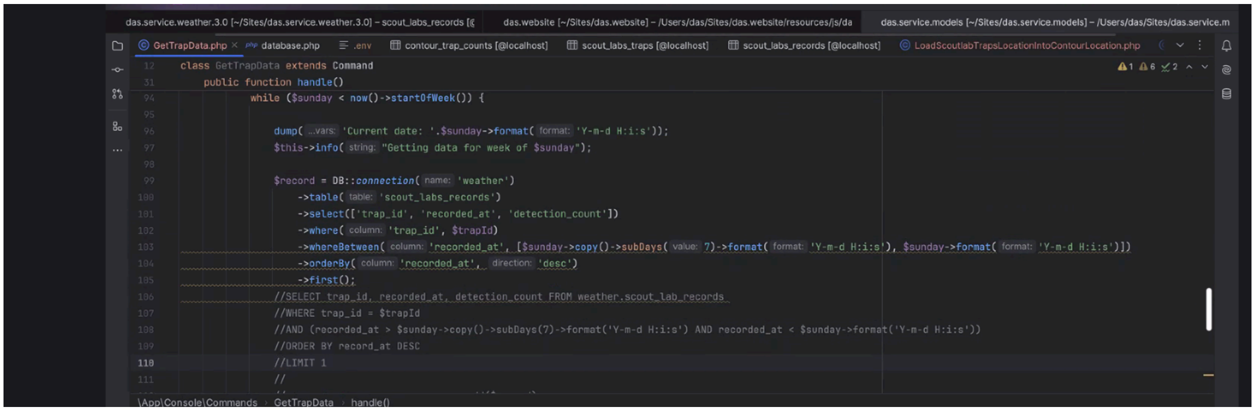
Clamped date added:



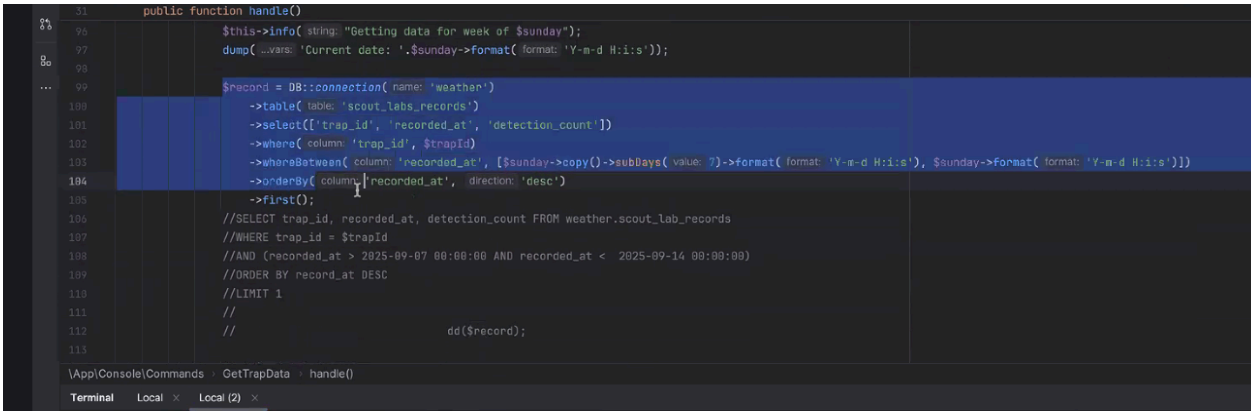
```
class GetTrapData extends Command
    public function handle()
                $record = DB::connection( name: 'weather')
                    ->table( table: 'scout_labs_records')
                    ->select(['trap_id', 'recorded_at', 'detection_count'])
                    ->where( column: 'trap_id', $trapId)
                    ->whereBetween( column: 'recorded_at', [$sunday->copy()->subDays( value: 7)->format( format: 'Y-m-d H:i:s'), $sunday->format( format: 'Y-m-d H:i:s')])
                    ->orderBy( column: 'recorded_at',  direction: 'desc')
                    ->first();
//                              dd($record);

                if (!$record) {
                    $this->info( string: "No record found for week of $sunday");
                    $sunday = $sunday->next(CarbonInterface::SUNDAY);
                    continue;
                }

                dump( ...vars: "Record Date: ".$record->recorded_at);

                $records->push($record);
```

```
"Current date: 2025-09-07 00:00:00" // app/Console/Commands/GetTrapData.php:96
Getting data for week of 2025-09-07 00:00:00
"Record Date: 2025-09-06 13:41:05" // app/Console/Commands/GetTrapData.php:114
"Current date: 2025-09-14 00:00:00" // app/Console/Commands/GetTrapData.php:96
Getting data for week of 2025-09-14 00:00:00
"Record Date: 2025-09-07 13:42:05" // app/Console/Commands/GetTrapData.php:114
"Current date: 2025-09-21 00:00:00" // app/Console/Commands/GetTrapData.php:96
Getting data for week of 2025-09-21 00:00:00
"Record Date: 2025-09-20 13:52:05" // app/Console/Commands/GetTrapData.php:114
"Current date: 2025-09-28 00:00:00" // app/Console/Commands/GetTrapData.php:96
Getting data for week of 2025-09-28 00:00:00
"Record Date: 2025-09-27 14:00:05" // app/Console/Commands/GetTrapData.php:114
"Current date: 2025-10-05 00:00:00" // app/Console/Commands/GetTrapData.php:96
Getting data for week of 2025-10-05 00:00:00
```

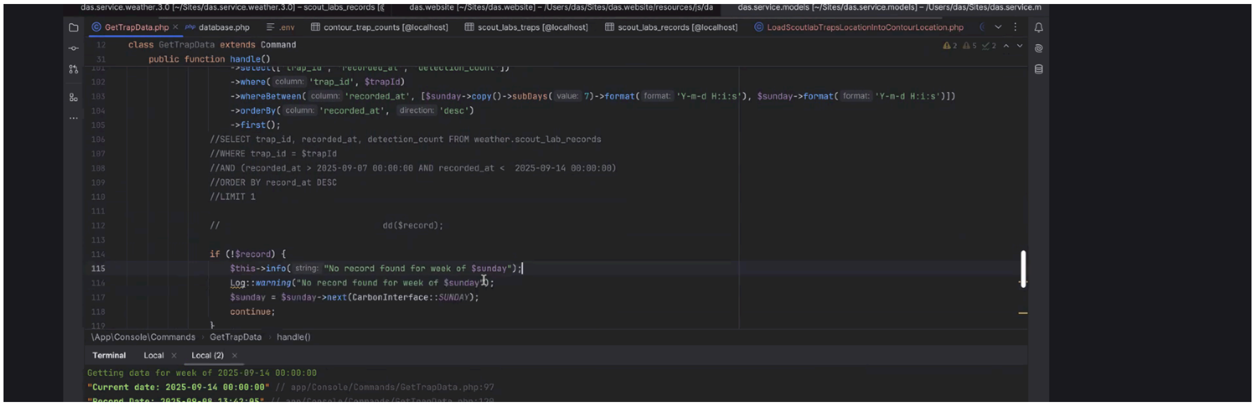
Annotated:


das.service.weather.3.0 [~/Sites/das.service.weather.3.0] – scout_labs_records [@
das.website [~/Sites/das.website] – /Users/das/Sites/das.website/resources/js/da
das.service.models [~/Sites/das.service.models] – /Users/das/Sites/das.service.m
GetTrapData.php ×
database.php
.env
contour_trap_counts [@localhost]
scout_labs_traps [@localhost]
scout_labs_records [@localhost]
LoadScoutlabTrapsLocationIntoContourLocation.php
⚠1 ⚠6 ✓2
\App\Console\Commands › GetTrapData › handle()

```
class GetTrapData extends Command
    public function handle()
            while ($sunday < now()->startOfWeek()) {

                dump( ...vars: 'Current date: '.$sunday->format( format: 'Y-m-d H:i:s'));
                $this->info( string: "Getting data for week of $sunday");

                $record = DB::connection( name: 'weather')
                    ->table( table: 'scout_labs_records')
                    ->select(['trap_id', 'recorded_at', 'detection_count'])
                    ->where( column: 'trap_id', $trapId)
                    ->whereBetween( column: 'recorded_at', [$sunday->copy()->subDays( value: 7)->format( format: 'Y-m-d H:i:s'), $sunday->format( format: 'Y-m-d H:i:s')])
                    ->orderBy( column: 'recorded_at',  direction: 'desc')
                    ->first();
                //SELECT trap_id, recorded_at, detection_count FROM weather.scout_lab_records
                //WHERE trap_id = $trapId
                //AND (recorded_at > $sunday->copy()->subDays(7)->format('Y-m-d H:i:s') AND recorded_at < $sunday->format('Y-m-d H:i:s'))
                //ORDER BY record_at DESC
                //LIMIT 1
                //
```



```
    public function handle()
                $this->info( string: "Getting data for week of $sunday");
                dump( ...vars: 'Current date: '.$sunday->format( format: 'Y-m-d H:i:s'));

                $record = DB::connection( name: 'weather')
                    ->table( table: 'scout_labs_records')
                    ->select(['trap_id', 'recorded_at', 'detection_count'])
                    ->where( column: 'trap_id', $trapId)
                    ->whereBetween( column: 'recorded_at', [$sunday->copy()->subDays( value: 7)->format( format: 'Y-m-d H:i:s'), $sunday->format( format: 'Y-m-d H:i:s')])
                    ->orderBy( column: 'recorded_at',  direction: 'desc')
                    ->first();
                //SELECT trap_id, recorded_at, detection_count FROM weather.scout_lab_records
                //WHERE trap_id = $trapId
                //AND (recorded_at > 2025-09-07 00:00:00 AND recorded_at <  2025-09-14 00:00:00)
                //ORDER BY record_at DESC
                //LIMIT 1
                //
                //                              dd($record);
```


das.service.weather.3.0 [~/Sites/das.service.weather.3.0] – scout_labs_records [@
das.website [~/Sites/das.website] – /Users/das/Sites/das.website/resources/js/da
das.service.models [~/Sites/das.service.models] – /Users/das/Sites/das.service.m
GetTrapData.php
database.php
.env
contour_trap_counts [@localhost]
scout_labs_traps [@localhost]
scout_labs_records [@localhost]
LoadScoutlabTrapsLocationIntoContourLocation.php
⚠2 ⚠5 ✓2

```
class GetTrapData extends Command
    public function handle()
                    ->select(['trap_id', 'recorded_at', 'detection_count'])
                    ->where( column: 'trap_id', $trapId)
                    ->whereBetween( column: 'recorded_at', [$sunday->copy()->subDays( value: 7)->format( format: 'Y-m-d H:i:s'), $sunday->format( format: 'Y-m-d H:i:s')])
                    ->orderBy( column: 'recorded_at',  direction: 'desc')
                    ->first();
                //SELECT trap_id, recorded_at, detection_count FROM weather.scout_lab_records
                //WHERE trap_id = $trapId
                //AND (recorded_at > 2025-09-07 00:00:00 AND recorded_at <  2025-09-14 00:00:00)
                //ORDER BY record_at DESC
                //LIMIT 1

                //                                  dd($record);

                if (!$record) {
                    $this->info( string: "No record found for week of $sunday");
                    Log::warning("No record found for week of $sunday");
                    $sunday = $sunday->next(CarbonInterface::SUNDAY);
                    continue;
                }
```

```
Getting data for week of 2025-09-14 00:00:00
"Current date: 2025-09-14 00:00:00" // app/Console/Commands/GetTrapData.php:97
"Record Date: 2025-09-08 13:42:05" // app/Console/Commands/GetTrapData.php:120
```