

Submission Worksheet

CLICK TO GRADE

<https://learn.ethereallab.app/assignment/IT202-008-S2024/it202-api-project-milestone-2-2024/grade/ak2774>

IT202-008-S2024 - [IT202] API Project Milestone 2 2024

Submissions:

1. Submission Selection

1 Submission [active] 4/15/2024 12:05:20 PM
3.
4.

5. Instructions
6.

^ COLLAPSE ^

7.
8.
9.
10. Implement the Milestone 2 features from the project's proposal document: <https://docs.google.com/document/d/1XE96a8DQ52Vp49XACBDTNCq0xYDt3kF29cO88E>
Make sure you add your ucid/date as code comments where code changes are done
All code changes should reach the Milestone2 branch
Create a pull request from Milestone2 to dev and keep it open until you get the output PDF from this assignment.
Gather the evidence of feature completion based on the below tasks.
Once finished, get the output PDF and copy/move it to your repository folder on your local machine.
Run the necessary git add, commit, and push steps to move it to GitHub
Complete the pull request that was opened earlier
Create and merge a pull request from dev to prod
Upload the same output PDF to Canvas

Branch name: Milestone2

Tasks: 29 **Points:** 10.00

● Define Appropriate Tables for Data (1 pt.)

^ COLLAPSE ^

● Task #1 - Points: 1

Text: Screenshots of Table SQL

Checklist

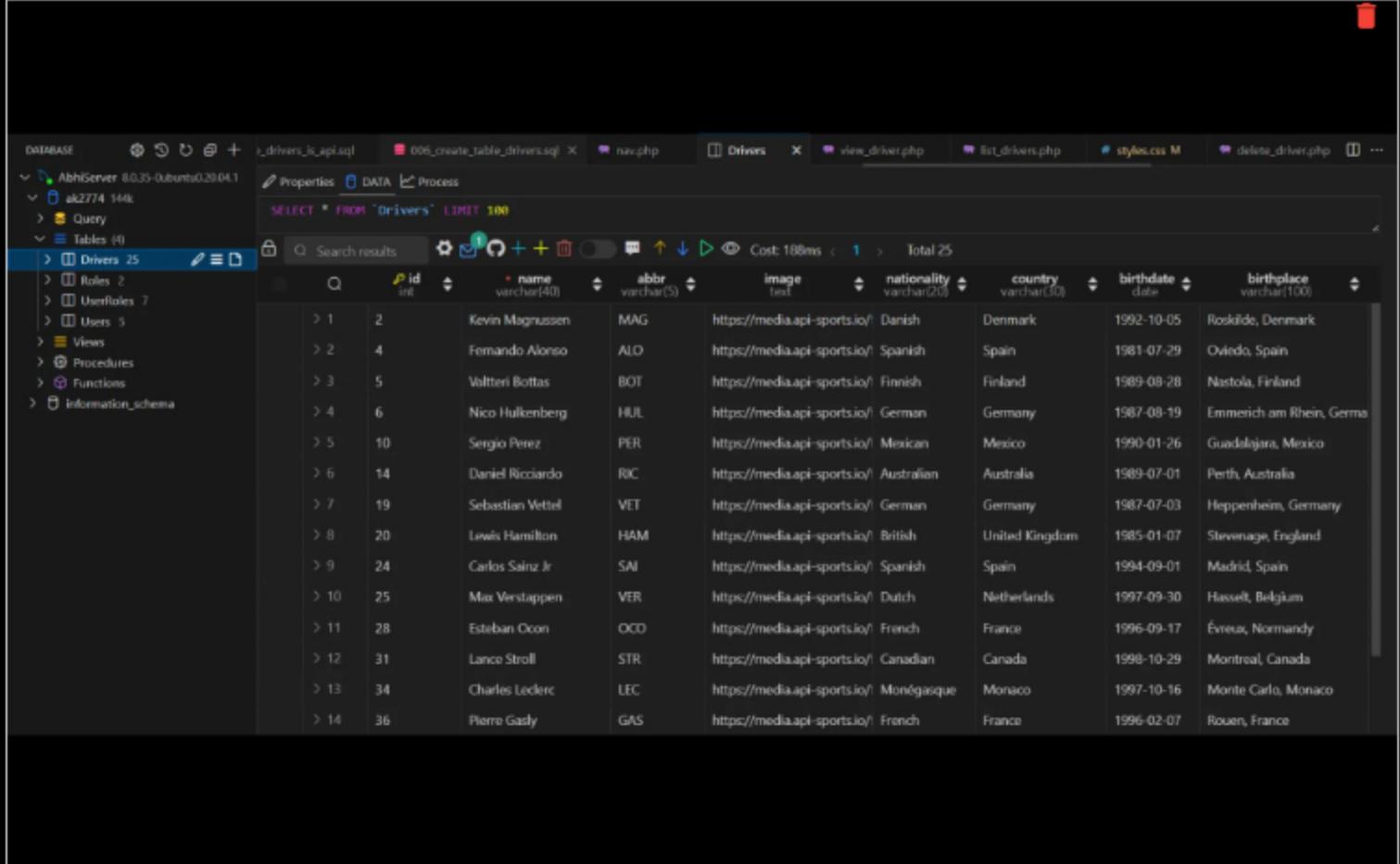
*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Table(s) should have the 3 core columns we'll always be using (<code>id</code> , <code>created</code> , <code>modified</code>) plus additional columns for the incoming API data
<input type="checkbox"/> #2	1	Columns should be logical and thought out (not valid to have a single field of JSON data or similar)
<input type="checkbox"/> #3	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

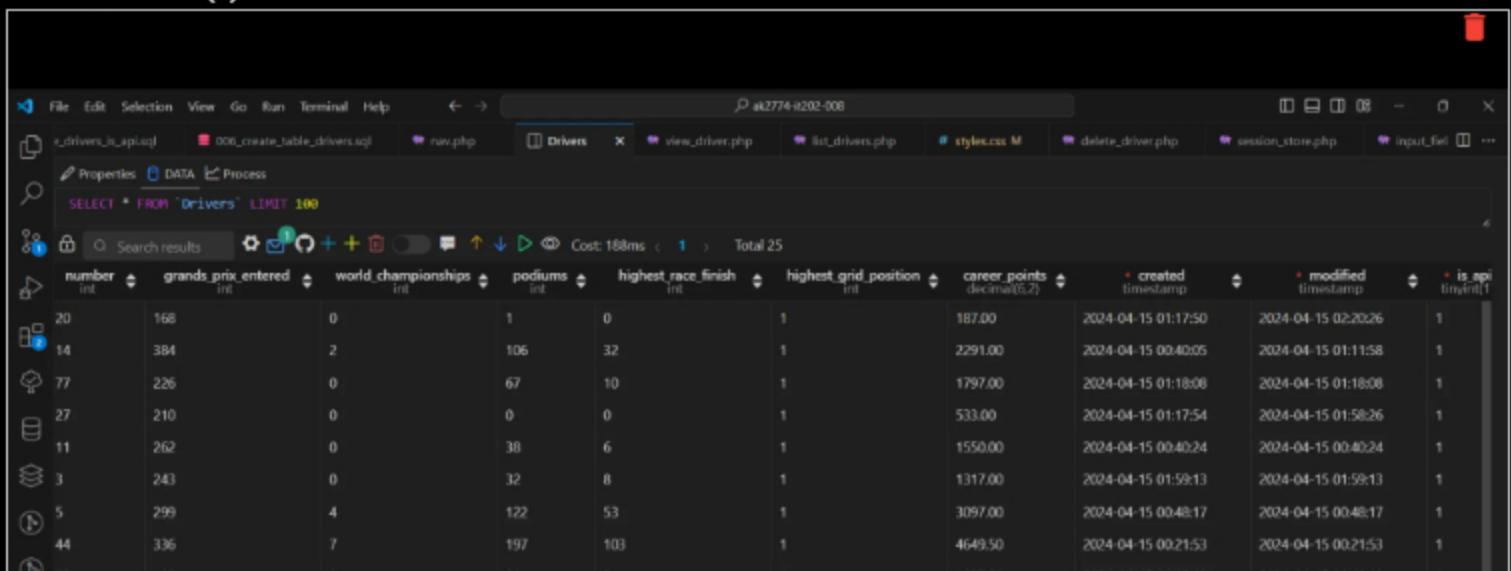
Small Medium Large



The screenshot shows a database management interface with a sidebar containing database and schema navigation. The main area displays a table titled 'Drivers' with 25 rows of data. The columns are: id (int), name (varchar(40)), abbr (varchar(5)), image (text), nationality (varchar(20)), country (varchar(30)), birthdate (date), and birthplace (varchar(100)). The data includes names like Kevin Magnussen, Fernando Alonso, and Lewis Hamilton, along with their respective nationalities, countries of birth, and birthdates.

Screenshot of database table

Checklist Items (0)



The screenshot shows a database management interface with a sidebar containing database and schema navigation. The main area displays a table with 10 rows of data. The columns are: number (int), grands_prix_entered (int), world_championships (int), podiums (int), highest_race_finish (int), highest_grid_position (int), career_points (decimal(6,2)), created (timestamp), modified (timestamp), and is_api (tinyint). The data includes various race statistics and career points for different drivers.

55	188	0	21	3	1	1037.50	2024-04-15 00:40:12	2024-04-15 00:40:12	1
1	189	3	101	57	1	2663.50	2024-04-15 01:08:47	2024-04-15 01:08:47	1
31	137	0	3	1	3	422.00	2024-04-15 01:59:04	2024-04-15 01:59:04	1
18	147	0	3	0	1	277.00	2024-04-15 01:08:52	2024-04-15 01:09:22	1
16	129	0	32	5	1	1133.00	2024-04-15 00:40:09	2024-04-15 00:40:09	1
10	134	0	4	1	2	394.00	2024-04-15 01:58:59	2024-04-15 01:58:59	1
7	(NULL)	7	155	(NULL)	(NULL)	1566.00	2024-04-15 02:00:16	2024-04-15 02:00:16	1
4	108	0	14	0	1	670.00	2024-04-15 01:17:29	2024-04-15 01:58:40	1
23	85	0	2	0	4	228.00	2024-04-15 01:18:00	2024-04-15 01:58:07	1

Screenshot of database table

Checklist Items (0)

Task #2 - Points: 1

Text: Explain the design

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Note the different fields and their purpose
<input type="checkbox"/> #2	1	Note if you needed to normalize the data into separate tables or if one table fit your needs

Response:

First column is id, which is the id number from the API, the second column is the driver name to store their name, third column is the driver abbreviation to store thier 3 letter abbr, fourth column is the image link to store a pic of the driver, fifth column is nationality which stores the nationality of the driver, sixth column is country which stores what country the driver is from, seventh column is birthdate which stores each drivers birthdate, eight column is birthplace which stores where the driver is born, ninth column is number which stores the driver number, tenth column is gp's entered which stores how many races the driver entered, eleventh column is world championships which stores the drivers world championships, twelth column is podiums which stores the drivers podiums, thirteenth column is highest race finish which stores the drivers highest race finish, fourteenth column is highest grid position which stores the drivers highest grid position, fifteenth column is career points which stores the drivers total points, sixteenth column is created column which stores the time when the data was created, seventeenth column is modified which stores when it was last modified, eighteenth column is is api which stores whether the driver is from the api or not.

Task #3 - Points: 1

Text: Add the pull request link for the branch related to this feature

Details:

Note: the link should end with /pull/#. Same pull request shouldn't be used for each feature.

● Data Creation Page (2 pts.)

[^COLLAPSE ^](#)

● Task #1 - Points: 1

Text: Screenshots of the creation page

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Show potentially valid data filled in for the custom creation page
<input type="checkbox"/> #2	1	Show how the API data is fetched for API data (must be server-side)
<input type="checkbox"/> #3	1	Show examples of validation messages
<input type="checkbox"/> #4	1	Show an example of successful creation message
<input type="checkbox"/> #5	1	Design/Style should be considered (i.e., bootstrap, custom css, etc)
<input type="checkbox"/> #6	1	Make sure the heroku dev url is visible in the address bar
<input type="checkbox"/> #7	1	Clearly caption screenshots

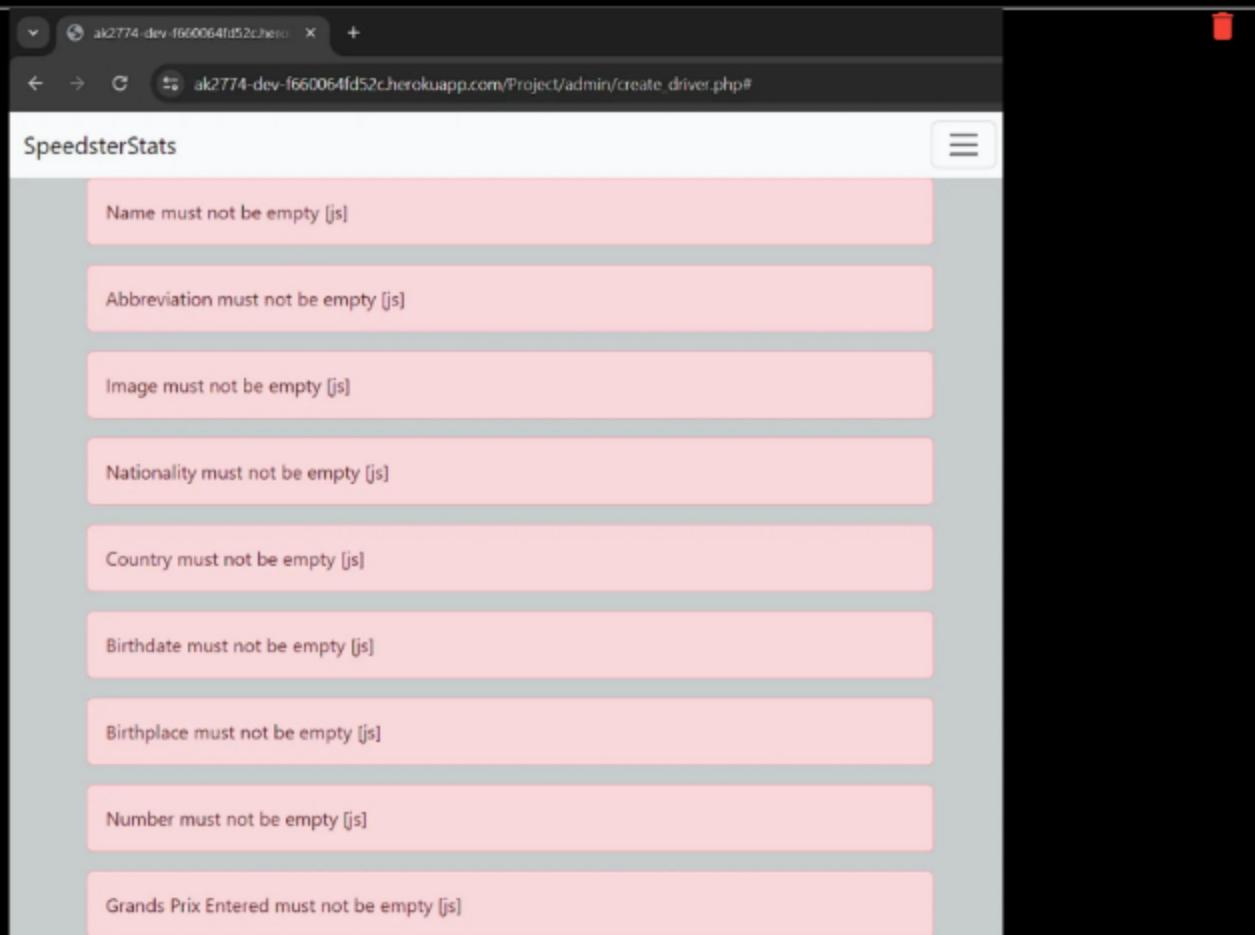
Task Screenshots:

Gallery Style: Large View

Small

Medium

Large



Checklist Items (0)

The screenshot shows a web browser window with the URL ak2774-dev-f660064fd52c.herokuapp.com/Project/admin/create_driver.php#. The page title is "Create or Fetch Driver". There are five validation error messages displayed in pink boxes:

- World Championships must not be empty [js]
- Podiums must not be empty [js]
- Wins must not be empty [js]
- Highest grid position must not be empty [js]
- Career points must not be empty [js]

Create or Fetch Driver

Screenshot of JS validation where all fields are empty part 2

Checklist Items (0)

The screenshot shows a web browser window with the URL ak2774-dev-f660064fd52c.herokuapp.com/Project/admin/create_driver.php#. The page title is "SpeedsterStats". There are nine validation error messages displayed in yellow boxes:

- You must provide a name
- You must provide an abbreviation
- You must provide an image
- You must provide a nationality
- You must provide a country
- You must provide a birthdate
- You must provide a birthplace
- You must provide a number
- You must provide a number of GPs entered

Screenshot of PHP validation where all fields are empty part 1

Checklist Items (0)

The screenshot shows a web browser window with a URL of `ak2774-dev-f66006fd52cherokuapp.com/Project/admin/create_driver.php#`. The page displays a series of validation error messages in yellow and pink boxes, indicating required fields. Below the errors, there is a title and two buttons.

You must provide a number

You must provide a number of GPs entered

You must provide a number of world championships

You must provide a number of podiums

You must provide a number of wins

You must provide a highest grid position

You must provide a number of career points

An error occurred

Create or Fetch Driver

Fetch Create

Driver name

Screenshot of PHP validation where all fields are empty part 2

Checklist Items (0)

The screenshot shows a web browser window with a URL of `ak2774-dev-f66006fd52cherokuapp.com/Project/admin/create_driver.php`. The page displays a title and a search interface. There are validation errors shown in yellow and pink boxes, similar to the first screenshot.

SpeedsterStats Home Profile Admin ▾ Driver Management ▾ Logout

Create or Fetch Driver

Fetch Create

Driver name

Search

Screenshot of fetch page

Checklist Items (0)

Create or Fetch Driver

Fetch Create

Driver Name

Driver Name

Driver Abbreviation

Drive Abbreviation

Image URL

Image URL

Nationality

Nationality

Country

Country

DOB

mm/dd/yyyy

Screenshot of create page part 1

Checklist Items (0)

Birthplace

Birthplace

Driver Number

Driver Number

GPs Entered

GPs Entered

World Championships

WCs

Podiums

Podiums

Wins

Wins

Highest Grid Position

Highest grid position

Career Points

Career Points

Screenshot of create page part 2

Checklist Items (0)

The screenshot shows a web browser window with a form titled "Create". The form contains the following fields:

- Driver Number: A text input field.
- GPs Entered: A text input field.
- World Championships: A text input field.
- Podiums: A text input field.
- Wins: A text input field.
- Highest Grid Position: A text input field.
- Career Points: A text input field.

At the bottom of the form is a blue "Create" button.

Screenshot of Create page part 3

Checklist Items (0)

The screenshot shows a web browser window with a confirmation message and a driver management section.

Confirmation Message:

Inserted record 2

Driver Management Section:

Create or Fetch Driver

Fetch Create

Driver name:

Search

Screenshot of fetch working

Checklist Items (0)

Driver already exists, please enter a different driver

Create or Fetch Driver

Fetch Create

Driver name

Search

Screenshot of fetch where driver already exists

Checklist Items (0)

Driver Name
Abhinav Karthik

Driver Abbr
KAR

Image URL
NULL

Nationality
Indian

Country
India

DOB

11/02/2004



Screenshot of creating driver part 1

Checklist Items (0)

DOB
11/02/2004

Birthplace
Chennai, India

Driver Number
44

GPs Entered
1

World Championships
1

Podiums
1

Wins
1

Highest Grid Position
1

Screenshot of creating driver part 2

Checklist Items (0)

Driver Number
44

GPs Entered
1

World Championships
1

Podiums
1

Wins
1

Highest Grid Position
1

Career Points

1

Create

Screenshot of create driver part 3

Checklist Items (0)

The screenshot shows a web browser window with the URL ak2774-dev-f66006fd52.herokuapp.com/Project/admin/create_driver.php. The page title is "Create or Fetch Driver". At the top, there are navigation links: SpeedsterStats, Home, Profile, Admin (with a dropdown arrow), Driver Management (with a dropdown arrow), and Logout. A success message "Inserted record 107" is displayed in a green box. Below it, there are two buttons: "Fetch" and "Create". A search bar labeled "Driver name" is followed by a "Search" button.

Screenshot of create driver complete

Checklist Items (0)

Task #2 - Points: 1

Text: Screenshots of creation page code

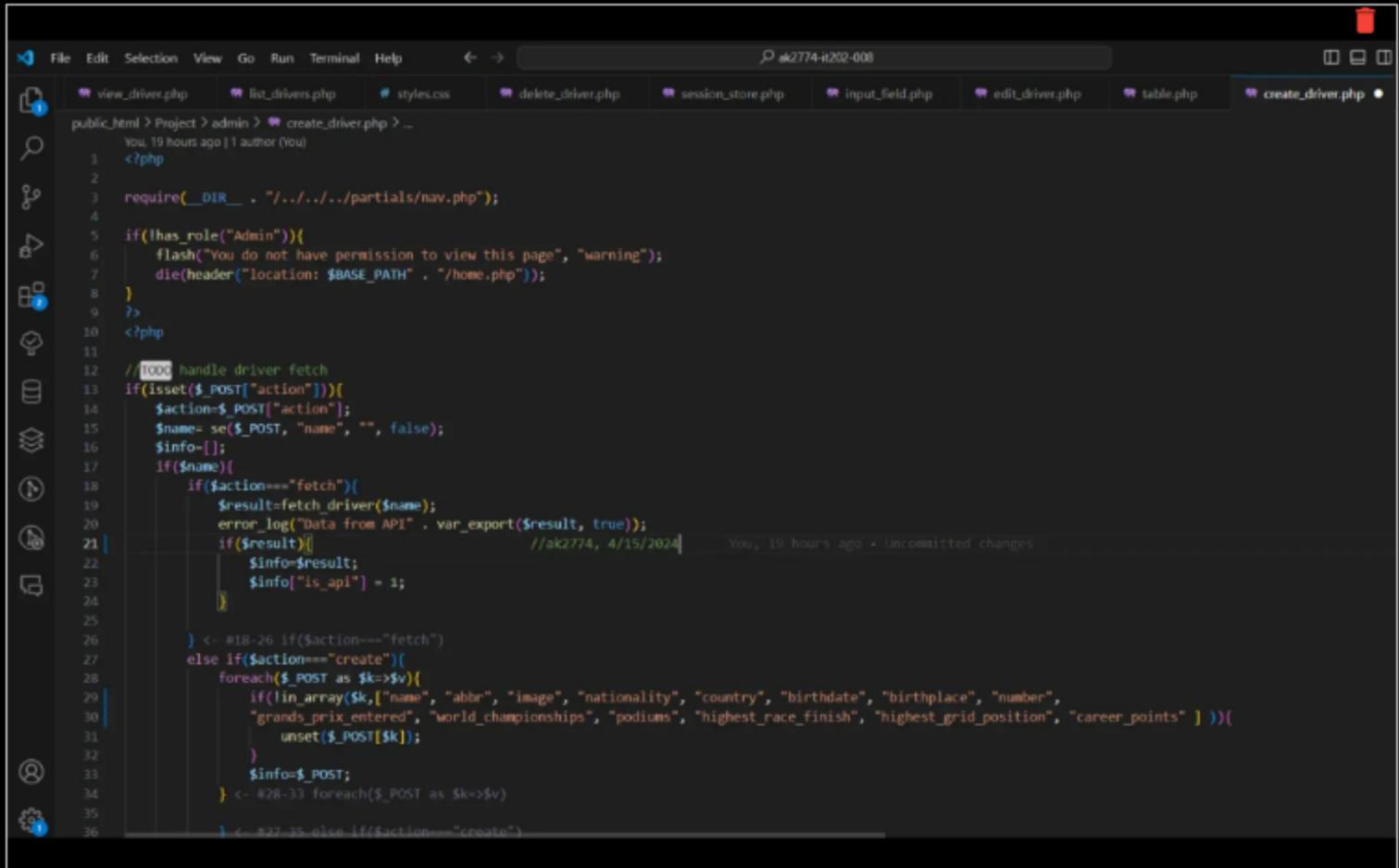
Checklist			*The checkboxes are for your own tracking
#	Points	Details	
<input type="checkbox"/> #1	1	Form should have correct data types for each property being requested	
<input type="checkbox"/> #2	1	Form should have correct validation for each field (HTML, JS, and PHP)	
<input type="checkbox"/> #3	1	Successful creation should have a user-friendly message	
<input type="checkbox"/> #4	1	Any errors should have user-friendly messages	
<input type="checkbox"/> #5	1	Include the form/process for fetching API data	
<input type="checkbox"/> #6	1	Include some indicator between custom data and API data	

<input type="checkbox"/> #6	1	Include some indicator between custom data and API data
<input type="checkbox"/> #7	1	Include any other rules like role guards and login checks
<input type="checkbox"/> #8	1	Include ucid/date comments for each code screenshot
<input type="checkbox"/> #9	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small Medium Large



```

public_html > Project > admin > create_driver.php > ...
You, 19 hours ago | 1 author (you)
<?php
require(__DIR__ . "/../../../../partials/nav.php");

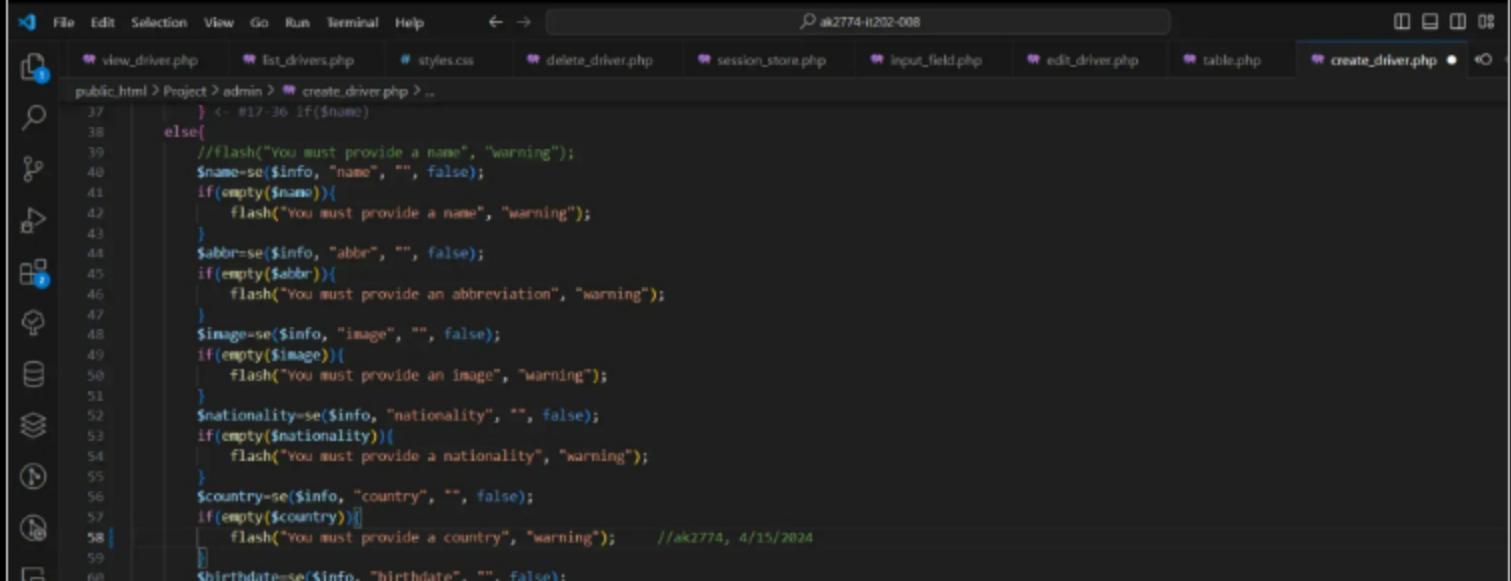
if(!has_role("Admin")){
    flash("You do not have permission to view this page", "warning");
    die(header("location: $BASE_PATH" . "/home.php"));
}

//18 handle driver fetch
if(isset($_POST["action"])){
    $action=$_POST["action"];
    $name= se($_POST, "name", "", false);
    $info=[];
    if($name){
        if($action=="fetch"){
            $result=fetch_driver($name);
            error_log("Data from API" . var_export($result, true));
            if($result){
                $info=$result;
                $info["is_api"] = 1;
            }
        }
        //18-26 if($action=="fetch")
    } else if($action=="create"){
        foreach($_POST as $k=>$v){
            if(!in_array($k, ["name", "abbr", "image", "nationality", "country", "birthdate", "birthplace", "number",
                "grands_prix_entered", "world_championships", "podiums", "highest_race_finish", "highest_grid_position", "career_points"])){
                unset($_POST[$k]);
            }
        }
        $info=$_POST;
        //28-33 foreach($_POST as $k=>$v)
    }
    //27-35 else if($action=="create")
}

```

Screenshot of fetch and create code

Checklist Items (0)



```

public_html > Project > admin > create_driver.php > ...
} <- #17-36 if($name)
else{
    //flash("You must provide a name", "warning");
    $name=se($info, "name", "", false);
    if(empty($name)){
        flash("You must provide a name", "warning");
    }
    $abbr=se($info, "abbr", "", false);
    if(empty($abbr)){
        flash("You must provide an abbreviation", "warning");
    }
    $image=se($info, "image", "", false);
    if(empty($image)){
        flash("You must provide an image", "warning");
    }
    $nationality=se($info, "nationality", "", false);
    if(empty($nationality)){
        flash("You must provide a nationality", "warning");
    }
    $country=se($info, "country", "", false);
    if(empty($country)){
        flash("You must provide a country", "warning");
    }
    $birthdate=se($info, "birthdate", "", false);
}

```

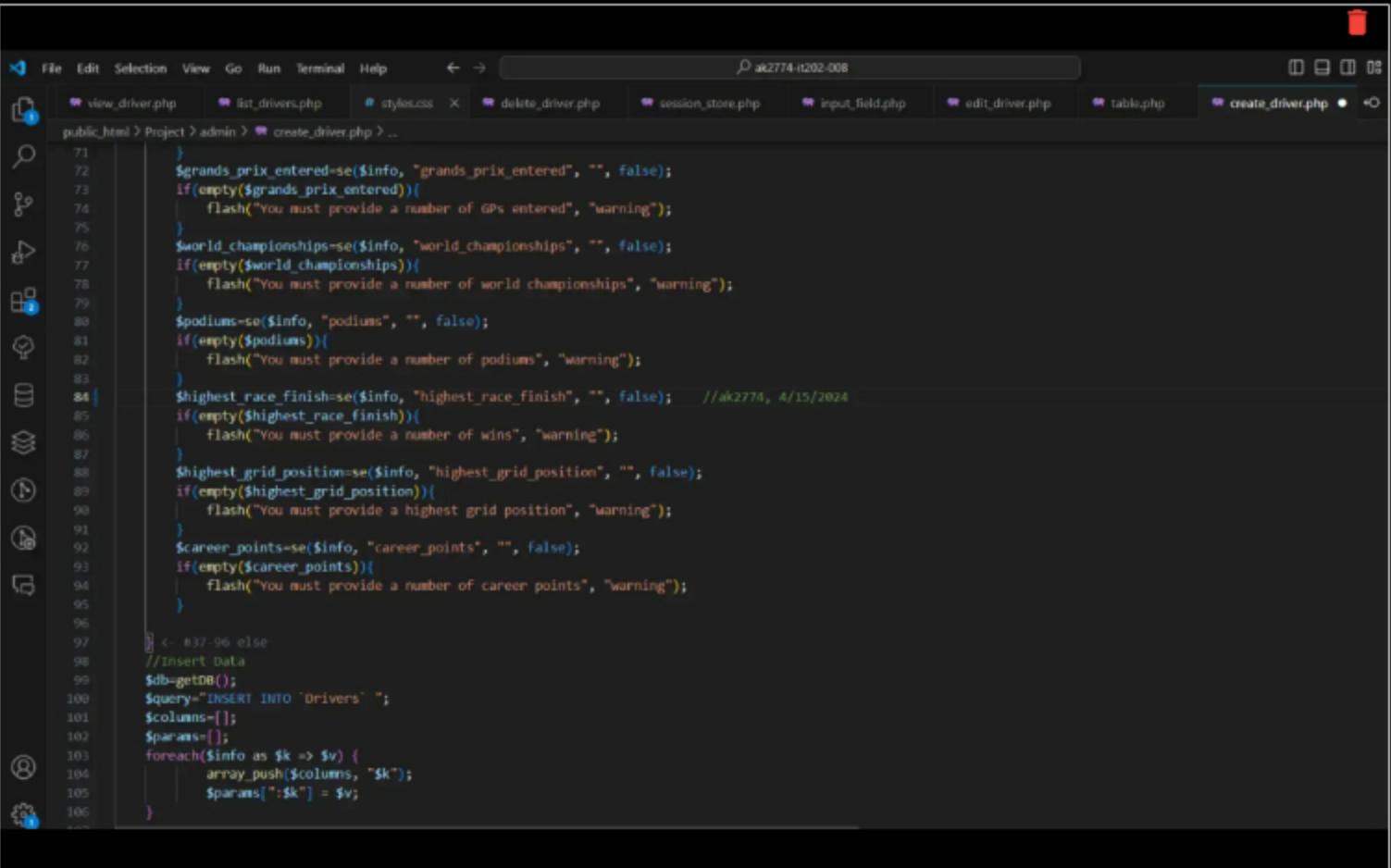
```

61     if(empty($birthdate)){
62         flash("You must provide a birthdate", "warning");
63     }
64     $birthplace=se($info, "birthplace", "", false);
65     if(empty($birthplace)){
66         flash("You must provide a birthplace", "warning");
67     }
68     $number=se($info, "number", "", false);
69     if(empty($number)){
70         flash("You must provide a number", "warning");
71     }

```

Screenshot of php validation part 1

Checklist Items (0)



The screenshot shows a code editor with a dark theme. The file being edited is `create_driver.php`. The code contains validation logic for various fields: birthdate, birthplace, and number. If any of these fields are empty, a flash message is displayed indicating that a value must be provided.

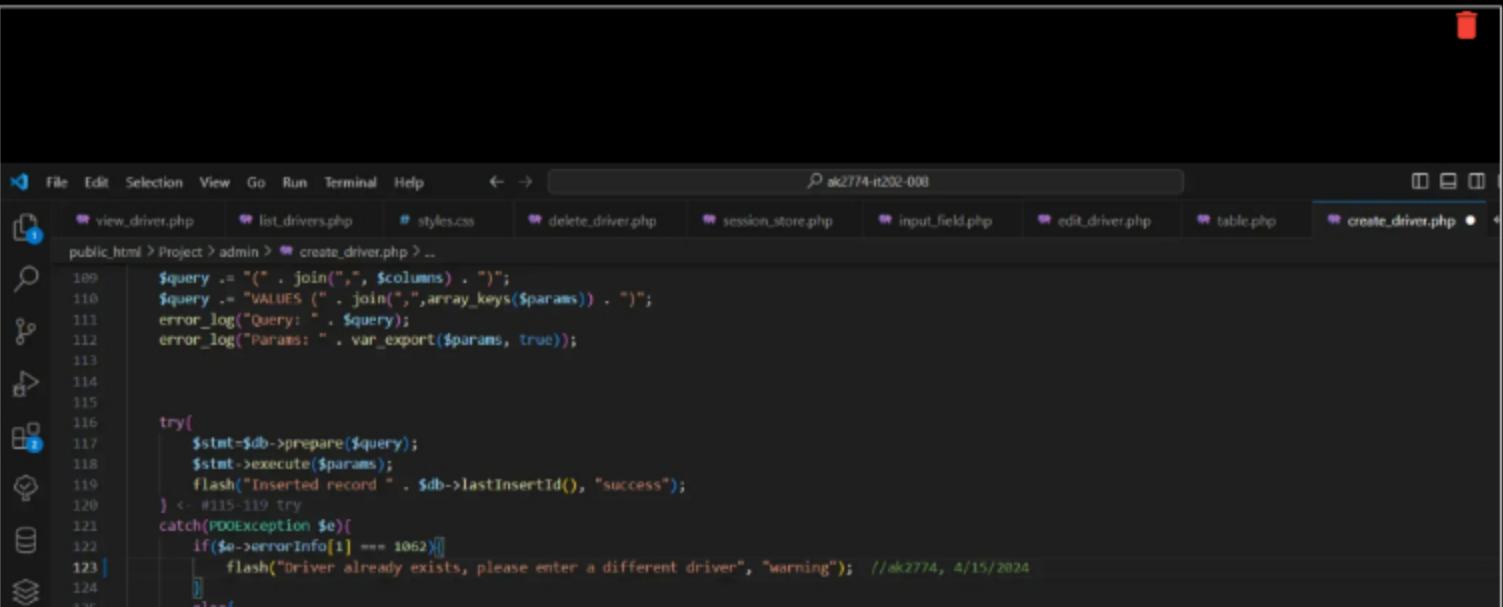
```

71     }
72     $grands_prix_entered=se($info, "grands_prix_entered", "", false);
73     if(empty($grands_prix_entered)){
74         flash("You must provide a number of GPs entered", "warning");
75     }
76     $world_championships=se($info, "world_championships", "", false);
77     if(empty($world_championships)){
78         flash("You must provide a number of world championships", "warning");
79     }
80     $podiums=se($info, "podiums", "", false);
81     if(empty($podiums)){
82         flash("You must provide a number of podiums", "warning");
83     }
84     $highest_race_finish=se($info, "highest_race_finish", "", false); //ak2774, 4/15/2024
85     if(empty($highest_race_finish)){
86         flash("You must provide a number of wins", "warning");
87     }
88     $highest_grid_position=se($info, "highest_grid_position", "", false);
89     if(empty($highest_grid_position)){
90         flash("You must provide a highest grid position", "warning");
91     }
92     $career_points=se($info, "career_points", "", false);
93     if(empty($career_points)){
94         flash("You must provide a number of career points", "warning");
95     }
96
97     } <-- #37-96 else
98     //Insert Data
99     $db->getDB();
100    $query="INSERT INTO `Drivers` ";
101    $columns=[];
102    $params=[];
103    foreach($info as $k => $v) {
104        array_push($columns, "$k");
105        $params[":$k"] = $v;
106    }

```

Screenshot of php validation part 2

Checklist Items (0)



The screenshot continues from the previous code editor session. It shows the insertion logic for the `Drivers` table. The code constructs an SQL query with columns and values, prepares it with the database connection, and executes it. A success message is flashed if the record is inserted successfully.

```

109     $query .= "(" . join(", ", $columns) . ")";
110     $query .= "VALUES (" . join(", ",array_keys($params)) . ")";
111     error_log("Query: " . $query);
112     error_log("Params: " . var_export($params, true));
113
114
115
116     try{
117         $stmt=$db->prepare($query);
118         $stmt->execute($params);
119         flash("Inserted record " . $db->lastInsertId(), "success");
120     } <- #115-119 try
121     catch(PDOException $e){
122         if($e->errorInfo[1] === 1062){
123             flash("Driver already exists, please enter a different driver", "warning"); //ak2774, 4/15/2024
124         }
125     }

```

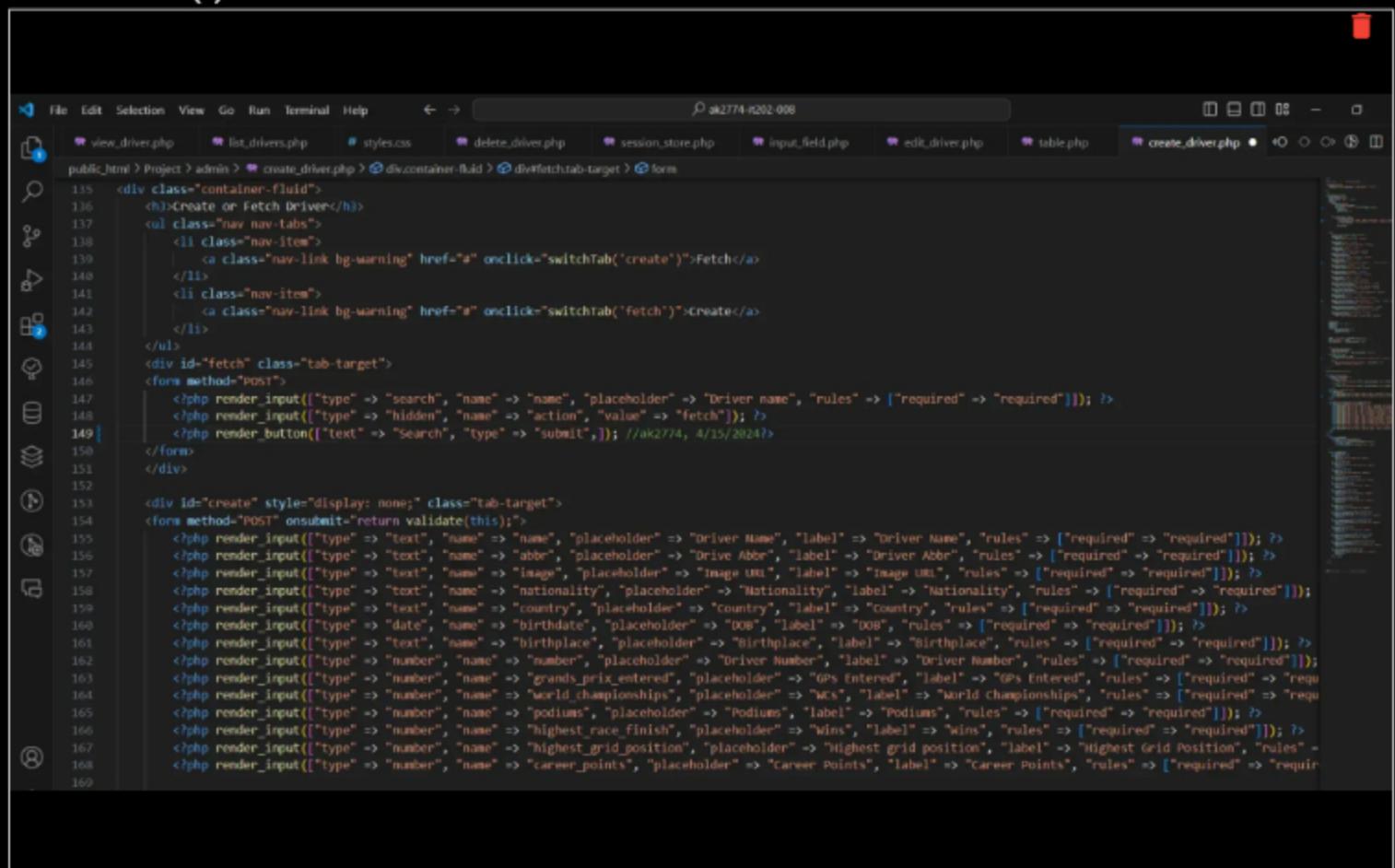
```

129     } <- #120-128 catch(PDOException $e)
130 } <- #13-129 if(isset($_POST["action"]))
131
132 //1000 handle manual create driver
133

```

Screenshot of data insertion

Checklist Items (0)



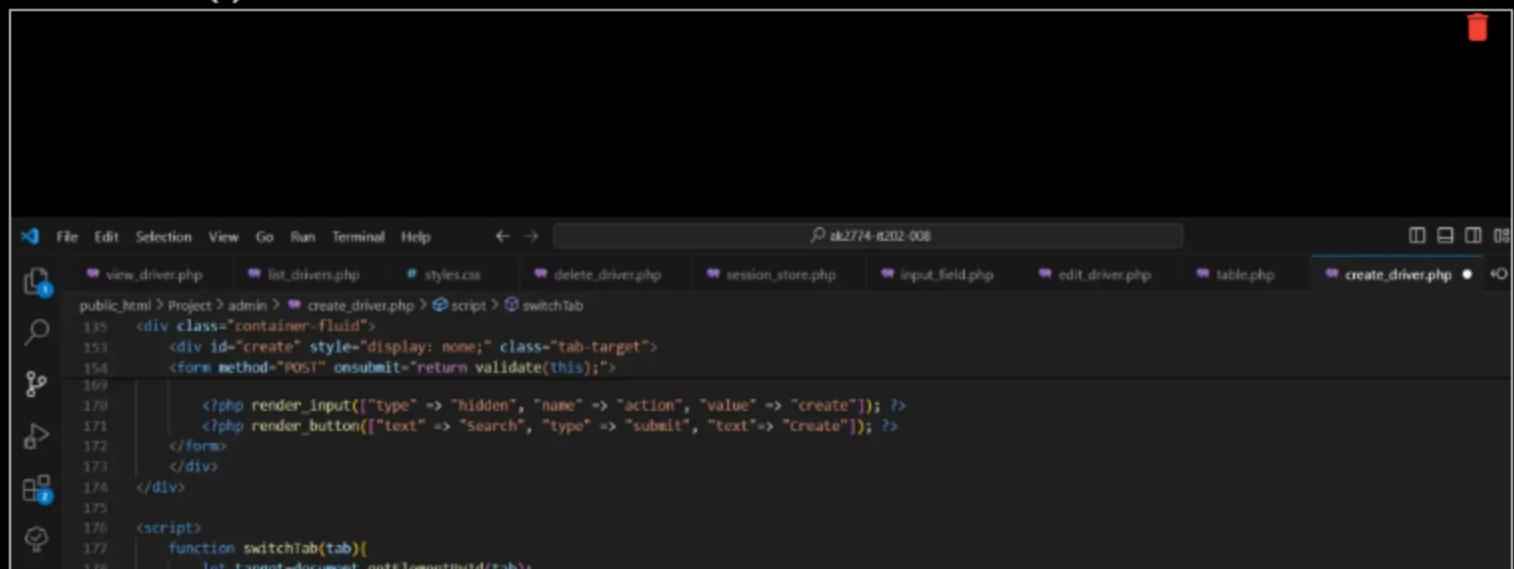
The screenshot shows a web browser window with the URL `ak2774-it202-008`. The page displays a form for creating a driver. At the top, there are two tabs: "Create" (which is active) and "Fetch". The "Create" tab contains the following fields:

- Name: Driver Name
- Abbr: Drive Abbr
- Image: Image URL
- Nationality: Nationality
- Country: Country
- Birthday: DOB
- Birthplace: Birthplace
- Driver Number: Driver Number
- GPs Entered: Grands Prix Entered
- World Championships: World Championships
- Podiums: Podiums
- Highest Race Finish: Wins
- Highest Grid Position: Highest Grid Position
- Career Points: Career Points

The code for this form is visible in the browser's developer tools, showing PHP logic for rendering the inputs and handling the POST request.

Screenshot of css code part 1

Checklist Items (0)



The screenshot shows a web browser window with the URL `ak2774-it202-008`. The page displays a script block containing the following function:

```

function switchTab(tab){
    let target=document.getElementById(tab);

```

This function is part of a larger script block that handles tab switching between "Create" and "Fetch". The "Create" tab is currently active, and the "Fetch" tab is inactive.

```
176
177
178    let target=document.getElementsByClassName("tab-target");
179    if(target){
180        let eles=document.getElementsByTagName("div");
181        for(let ele of eles){
182            ele.style.display=(ele.id==tab)? "none" : "block"; //ak2774, 4/15/2024
183        }
184    } <- #178-183 if(target)
185 } <- #176-184 function switchTab(tab)
```

Screenshot of css code part 2 and switchTab function

Checklist Items (0)

A screenshot of a code editor window titled 'ak2774-t202-008'. The file being edited is 'create_driver.php'. The code contains CSS rules and a JavaScript validation function. The validation function, 'validate', checks various form fields like name, abbr, image, nationality, country, birthdate, birthplace, number, and grands_prix_entered for emptiness. If a field is empty, it uses the 'flash' function to display an error message ('danger') and set the 'isValid' variable to false.

```
File Edit Selection View Go Run Terminal Help ← → ⌘ ak2774-t202-008
view_driver.php list_drivers.php styles.css delete_driver.php session_store.php input_field.php edit_driver.php table.php create_driver.php
public_html > Project > admin > create_driver.php > script > validate
176 <script>
177
178     function validate(form){
179         let isValid=true;
180         let name=form.name.value;
181         if(name==""){
182             flash("Name must not be empty [js]", "danger");
183             isValid=false;
184         }
185         let abbr=form.abbr.value;
186         if(abbr==""){
187             flash("Abbreviation must not be empty [js]", "danger");
188             isValid=false;
189         }
190         let image=form.image.value;
191         if(image==""){
192             flash("Image must not be empty [js]", "danger");
193             isValid=false;
194         }
195         let nationality=form.nationality.value;
196         if(nationality==""){
197             flash("Nationality must not be empty [js]", "danger");
198         }
199         let country=form.country.value;
200         if(country==""){
201             flash("Country must not be empty [js]", "danger");
202             isValid=false;
203         }
204         let birthdate=form.birthdate.value;
205         if(birthdate==""){
206             flash("Birthdate must not be empty [js]", "danger");
207             isValid=false;
208         }
209         let birthplace=form.birthplace.value;
210         if(birthplace==""){
211             flash("Birthplace must not be empty [js]", "danger");
212             isValid=false;
213         }
214         let number=form.number.value;
215         if(number==""){
216             flash("Number must not be empty [js]", "danger");
217             isValid=false;
218         }
219         let grands_prix_entered=form.grands_prix_entered.value;
220         if(grands_prix_entered==""){
221             flash("Grands Prix Entered must not be empty [js]", "danger");
222             isValid=false;
223         }
224         let world_championships=form.world_championships.value;
225         if(world_championships==""){
226             flash("World Championships must not be empty [js]", "danger");
227             isValid=false;
228         }
229     }
230     function switchTab(tab){
```

Screenshot of js validation part 1

Checklist Items (0)

A screenshot of a code editor window titled 'ak2774-t202-008'. The file being edited is 'create_driver.php'. The code continues from the previous screenshot, validating fields like birthplace, number, grands_prix_entered, and world_championships. It uses the 'flash' function to show errors and set the 'isValid' variable accordingly.

```
File Edit Selection View Go Run Terminal Help ← → ⌘ ak2774-t202-008
view_driver.php list_drivers.php styles.css delete_driver.php session_store.php input_field.php edit_driver.php table.php create_driver.php
public_html > Project > admin > create_driver.php > script > validate
176 <script>
177     function validate(form){
178         let birthplace=form.birthplace.value;
179         if(birthplace==""){
180             flash("Birthplace must not be empty [js]", "danger");
181             isValid=false;
182         }
183         let number=form.number.value;
184         if(number==""){
185             flash("Number must not be empty [js]", "danger");
186             isValid=false;
187         }
188         let grands_prix_entered=form.grands_prix_entered.value;
189         if(grands_prix_entered==""){
190             flash("Grands Prix Entered must not be empty [js]", "danger");
191             isValid=false;
192         }
193         let world_championships=form.world_championships.value;
194         if(world_championships==""){
195             flash("World Championships must not be empty [js]", "danger");
196             isValid=false;
197         }
198     }
199     function switchTab(tab){
```

```

234     if(world_championships==""){
235         flash("World Championships must not be empty [js]", "danger"); //ak2774, 4/15/2024
236         isValid=false;
237     }
238     let podiums=form.podiums.value;
239     if(podiums==""){
240         flash("Podiums must not be empty [js]", "danger");
241         isValid=false;
242     }
243     let highest_race_finish=form.highest_race_finish.value;
244     if(highest_race_finish==""){
245         flash("Mins must not be empty [js]", "danger");
246         isValid=false;
247     }
248     let highest_grid_position=form.highest_grid_position.value;
249     if(highest_grid_position==""){
250         flash("Highest grid position must not be empty [js]", "danger");
251         isValid=false;

```

Screenshot of js validation part 2

Checklist Items (0)

```

File Edit Selection View Go Run Terminal Help ← → ⌘ ak2774-i202-008
view_driver.php list_drivers.php # styles.css delete_driver.php session_store.php input_field.php edit_driver.php table.php create_driver.php ⌋
public_html > Project > admin > create_driver.php > script > validate
176 <script>
177     function validate(form){
178         let highest_grid_position=form.highest_grid_position.value;
179         if(highest_grid_position==""){
180             flash("Highest grid position must not be empty [js]", "danger");
181             isValid=false;
182         }
183         let career_points=form.career_points.value;
184         if(career_points==""){
185             flash("Career points must not be empty [js]", "danger"); //ak2774, 4/15/2024
186             isValid=false;
187         }
188         return isValid;
189     } <- #186-259 function validate(form)
260   
```

Screenshot of js validation part 3

Checklist Items (0)

Task #3 - Points: 1

Text: Screenshot of records from DB

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Show at least one record fetched from the API
<input type="checkbox"/> #2	1	Show at least one record created via the creation form

<input type="checkbox"/> #3	1	Note what differs
<input type="checkbox"/> #4	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small

Medium

Large

	id	name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered
> 19	80	Mick Schumacher	MSC	https://media.api-sports.io/	German	Germany	1999-03-22	Vufflens-le-Château, Switzer	47	42
> 20	82	Yuki Tsunoda	TSU	https://media.api-sports.io/	Japanese	Japan	2000-05-11	Sagamihara, Japan	22	70
> 21	83	Guanyu Zhou	ZHO	https://media.api-sports.io/	Chinese	China	1999-05-30	Shanghai, China	24	48
> 22	89	Liam Lawson	(NULL)	https://media.api-sports.io/	New Zealander	New Zealand	2002-02-11	Hastings, New Zealand	40	(NULL)
> 23	92	Logan Sargeant	SAR	https://media.api-sports.io/	American	United States	2000-12-31	Fort Lauderdale, Florida	2	25
> 24	97	Oscar Piastri	PIA	https://media.api-sports.io/	Australian	Australia	2001-04-06	Melbourne, Victoria	81	26
> 25	101	Oliver Bearman	BEA	https://shorturl.at/yfLY	British	United Kingdom	2005-05-08	Chelmsford, Essex, England	38	1
> 26	107	Abhinav Karthik	KAR	NULL	Indian	India	2004-11-02	Chennai, India	44	1

Screenshot of custom driver DB insertion. The difference in the is_api value

Checklist Items (0)

	id	name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered
> 1	2	Kevin Magnussen	MAG	https://media.api-sports.io/	Danish	Denmark	1992-10-05	Roskilde, Denmark	20	168

Screenshot of fetch record working

Checklist Items (0)

id	driver_id	grands_prix_entered	world_championships	podiums	highest_race_finish	highest_grid_position	career_points	created_timestamp	modified_timestamp	is_api
42	0	0	0	6	12.00	2024-04-15 02:00:21	2024-04-15 02:00:50	2024-04-15 02:00:50	1	
70	0	0	0	6	60.00	2024-04-15 01:17:40	2024-04-15 01:57:56	2024-04-15 01:57:56	1	
48	0	0	0	5	12.00	2024-04-15 01:18:12	2024-04-15 01:57:43	2024-04-15 01:57:43	1	
(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	2024-04-15 01:18:18	2024-04-15 01:18:18	2024-04-15 01:18:18	1	
25	0	0	0	6	1.00	2024-04-15 01:17:02	2024-04-15 01:57:09	2024-04-15 01:57:09	1	
26	0	2	0	2	129.00	2024-04-15 01:17:34	2024-04-15 01:56:47	2024-04-15 01:56:47	1	
1	0	0	0	11	6.00	2024-04-12 14:35:37	2024-04-15 02:24:47	2024-04-15 02:24:47	1	
1	1	1	1	1	1.00	2024-04-15 17:58:56	2024-04-15 17:58:56	2024-04-15 17:58:56	0	

is_api value is 0 instead of one for the manually created record

Checklist Items (0)

Task #4 - Points: 1

Text: Explain the process

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Provide a high-level step-by-step of how fetching API data works and gets added to your DB (include how duplicates are handled)
<input type="checkbox"/> #2	1	Provide a high-level step-by-step of how creating custom data works and gets added to your DB (include how duplicates are handled)
<input type="checkbox"/> #3	1	Briefly describe the validations for the applicable fields
<input type="checkbox"/> #4	1	Describe how duplicate data is handled

Response:

When a user interacts with the system to fetch API data, they initiate the process by inputting a driver's name into the search field and submitting the form. On the server side, a PHP script receives this submission and first verifies if the user holds the necessary "Admin" role. If authorized, the script proceeds; otherwise, it redirects the user with a warning flash message. Subsequently, an API request is made to retrieve data associated with the specified driver name. Upon receiving a response from the API, the script evaluates the data's validity. If valid, the information is stored in the \$info array, tagged to indicate its origin from the API. Before insertion into the database, the script prepares an SQL query, ensuring to check for duplicates. If the driver already exists, a warning flash message alerts the user; otherwise, the data is inserted into the database. Throughout this process, the user receives feedback via flash messages, indicating success or failure.

Similarly, when creating custom data, users input details of a new driver via a form and submit it. The server-side PHP script, following the same protocol as with fetching API data, first verifies the user's "Admin" role. Validations are then applied to the submitted form data to ensure essential fields are not left empty. Any failed validations prompt the display of appropriate warning flash messages. Upon successful validation, the script proceeds to prepare an SQL query for database insertion. Duplicate data is checked before insertion, triggering a warning flash message if a match is found. Otherwise, the custom data is added to the database. Throughout this process, users are kept informed of the outcome through flash messages.

Task #5 - Points: 1

Text: Add related links

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Include the heroku prod link for this page
<input type="checkbox"/> #2	1	Add the pull request link for the branch related to this feature Note: the link should end with /pull/#. Same pull request shouldn't be used for each feature

URL #1

https://ak2774-prod-76402b453c05.herokuapp.com/Project/admin/create_driver.php

URL #2

<https://github.com/abhinavkarthikn/ak2774-it202-008/pull/63>

Data List Page (many entities) (2 pts.)

[^COLLAPSE ^](#)

Task #1 - Points: 1

Text: Screenshots of the list page

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Show the page of your entities listed (have a reasonable number shown)

<input type="checkbox"/> #2	1	Show the filter/sort form based on your data and the required limit field
<input type="checkbox"/> #3	1	Demonstrate a few varied filters/sorts
<input type="checkbox"/> #4	1	Demonstrate a filter that doesn't have any records (should show an appropriate message)
<input type="checkbox"/> #5	1	Each list item should have a link of single view (i.e., details), edit, and delete (some of which may only be visible to admin users)
<input type="checkbox"/> #6	1	Each list item should have a summary of the entity (likely won't be the entire entity data)
<input type="checkbox"/> #7	1	Design/Style should be considered (i.e., bootstrap, custom css, etc)
<input type="checkbox"/> #8	1	Make sure the heroku dev url is visible in the address bar
<input type="checkbox"/> #9	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small Medium Large

ak2774-dev-f660064fd52.herokuapp.com/Project/admin/list_drivers.php

SpeedsterStats Home Profile Admin ▾ Driver Management ▾ Logout

List Drivers

Name	Nationality	Country	Driver #	Min GPs	Max GPs
Name	Nationality	Country	Driver #	Min GPs	Max GPs
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min Points	Max Points	Sort	Order	Limit	
Min Points	Max Points	GPs	+	10	

Filter **Clear**

Screenshot of Filter section of the list page

Checklist Items (0)

All Drivers												
name	abbr	image	nationality	country	birthdate	birthplace	number	grands prix entered	world championships	podiums	highest race finish	highest grid
Kevin Magnussen	MAG	https://media.apis-sports.io/formula-1/drivers/kevin-magnussen.jpg	Danish	Denmark	1992-10-05	Roskilde, Denmark	20	168	0	1	1	4

Fernando Alonso	ALO	https://media.api-sports.io/formula-1/drivers/4.png	Spanish	Spain	1981-07-29	Oviedo, Spain	14	384	2	106	32	1
Valtteri Bottas	BOT	https://media.api-sports.io/formula-1/drivers/5.png	Finnish	Finland	1989-08-28	Nastola, Finland	77	226	0	67	10	1
Nico Hulkenberg	HUL	https://media.api-sports.io/formula-1/drivers/6.png	German	Germany	1987-08-19	Emmerich am Rhein, Germany	27	210	0	0	0	1
Sergio Perez	PER	https://media.api-sports.io/formula-1/drivers/10.png	Mexican	Mexico	1990-01-26	Guadalajara, Mexico	11	262	0	38	6	1

Screenshot of table part 1

Checklist Items (0)

	nationality	country	birthdate	birthplace	number	grands prix entered	world championships	podiums	highest race finish	highest grid position	career points	Actions
Alonso	Danish	Denmark	1992-10-05	Roskilde, Denmark	20	168	0	1	1	4	187.00	<button>View</button> <button>Edit</button> <button>Delete</button>
Alonso	Spanish	Spain	1981-07-29	Oviedo, Spain	14	384	2	106	32	1	2291.00	<button>View</button> <button>Edit</button> <button>Delete</button>
Bottas	Finnish	Finland	1989-08-28	Nastola, Finland	77	226	0	67	10	1	1797.00	<button>View</button> <button>Edit</button> <button>Delete</button>
Hulkenberg	German	Germany	1987-08-19	Emmerich am Rhein, Germany	27	210	0	0	0	1	533.00	<button>View</button> <button>Edit</button> <button>Delete</button>
Perez	Mexican	Mexico	1990-01-26	Guadalajara, Mexico	11	262	0	38	6	1	1550.00	<button>View</button> <button>Edit</button> <button>Delete</button>

Screenshot of table part 2

Checklist Items (0)

List Drivers												
Name	Nationality	Country	Driver #	Min GPs	Max GPs							
<input type="text"/>												
<input type="text"/>												

Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min Points	Max Points	Sort	Order	Limit	
Min Points	Max Points	GPs	+	10	

[Filter](#) [Clear](#)

All Drivers

name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered	world_championships	podiums	highest_race_finish	highest_grid_position	career
Esteban Ocon	OCO	https://media.api-sports.io/formula-1/drivers/28.png	French	France	1996-09-17	Évreux, Normandy	31	137	0	3	1	3	422.00
Lance Stroll	STR	https://media.api-sports.io/formula-1/drivers/31.png	Canadian	Canada	1998-10-29	Montreal, Canada	18	147	0	3	0	1	277.00

Screenshot of max and min GP's filter working

Checklist Items (0)

Screenshot of the driver management interface showing the 'Driver Management' tab selected. The page displays a list of drivers with various filters and sorting options.

Name	Nationality	Country	Driver #	Min GPs	Max GPs
lewis	Nationality	Country	Driver #	Min GPs	Max GPs
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min Points	Max Points	Sort	Order	Limit	
Min Points	Max Points	GPs	+	10	

[Filter](#) [Clear](#)

All Drivers

name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered	world_championships	podiums	highest_race_finish	highest_grid_position	career
Lewis Hamilton	HAM	https://media.api-sports.io/formula-1/drivers/20.png	British	United Kingdom	1985-01-07	Stevenage, England	44	336	7	197	103	1	4649.00

Screenshot of name filter working

Checklist Items (0)

Screenshot of the driver management interface showing the 'Driver Management' tab selected. The page displays a list of drivers with various filters and sorting options.

Name	Nationality	Country	Driver #	Min GPs	Max GPs
lewis	Nationality	Country	Driver #	Min GPs	Max GPs
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min Points	Max Points	Sort	Order	Limit	
Min Points	Max Points	GPs	+	10	

[Filter](#) [Clear](#)

List Drivers

Name	Nationality	Country	Driver #	Min GPs	Max GPs
<input type="text"/>	Danish	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Min Points	Max Points	Sort	Order	Limit	
<input type="text"/>	<input type="text"/>	<input type="button" value="GPs"/>	<input type="button" value="+"/>	<input type="text" value="10"/>	

All Drivers

name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered	world_championships	podiums	highest_race_finish	highest_grid_position	car
Kevin Magnussen	MAG	https://mediaapisports.io/formula-1/drivers/2.png	Danish	Denmark	1992-10-05	Roskilde, Denmark	20	168	0	1	1	4	187

Screenshot of nationality filter working

Checklist Items (0)

The screenshot shows the 'List Drivers' page of the SpeedsterStats application. A nationality filter has been applied, showing only drivers from Denmark. The table displays one row for Kevin Magnussen.

name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered	world_championships	podiums	highest_race_finish	highest_grid_position	car
Kevin Magnussen	MAG	https://mediaapisports.io/formula-1/drivers/2.png	Danish	Denmark	1992-10-05	Roskilde, Denmark	20	168	0	1	1	4	187

All Drivers

name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered	world_championships	podiums	highest_race_finish	highest_grid_position	car
Kevin Magnussen	MAG	https://mediaapisports.io/formula-1/drivers/2.png	Danish	Denmark	1992-10-05	Roskilde, Denmark	20	168	0	1	1	4	187

Screenshot of country filter working

Checklist Items (0)

The screenshot shows the 'List Drivers' page of the SpeedsterStats application. A country filter has been applied, showing only drivers from Denmark. The table displays one row for Kevin Magnussen.

name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered	world_championships	podiums	highest_race_finish	highest_grid_position	car
Kevin Magnussen	MAG	https://mediaapisports.io/formula-1/drivers/2.png	Danish	Denmark	1992-10-05	Roskilde, Denmark	20	168	0	1	1	4	187

List Drivers

Name	Nationality	Country	Driver #	Min GPs	Max GPs
Name	Nationality	Country	55	Min GPs	Max GPs
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min Points	Max Points	Sort	Order	Limit	
Min Points	Max Points	GPs	+	10	

FilterClear

All Drivers

name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered	world_championships	podiums	highest_race_finish	highest_grid_position	career_p...
Carlos Sainz Jr	SAI	https://media.api-sports.io/formula-1/drivers/24.png	Spanish	Spain	1994-09-01	Madrid, Spain	55	188	0	21	3	1	1037.50

Screenshot of driver # filter working

Checklist Items (0)

Screenshot of the SpeedsterStats application showing the 'List Drivers' search interface. The 'Driver #' input field contains the value '4'. Other fields like 'Min WCs', 'Max WCs', 'Min Points', 'Max Points', 'Sort', 'Order', and 'Limit' are also visible.

All Drivers

name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered	world_championships	podiums	highest_race_finish	highest_grid_position	car...
Sebastian Vettel	VET	https://media.api-sports.io/formula-1/drivers/19.png	German	Germany	1987-07-03	Heppenheim, Germany	5	299	4	122	53	1	309

Screenshot of max and min WCs filter working

Checklist Items (0)

Screenshot of the SpeedsterStats application showing the 'List Drivers' search interface. The 'Min WCs' input field contains the value '4' and the 'Max WCs' input field contains the value '5'. Other fields like 'Min Points', 'Max Points', 'Sort', 'Order', and 'Limit' are also visible.

List Drivers

Name	Nationality	Country	Driver #	Min GPs	Max GPs
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Min Points	Max Points	Sort	Order	Limit	
<input type="text"/>	<input type="text"/>	<input type="button" value="GPs"/>	<input type="button" value="+"/>	<input type="button" value="10"/>	

All Drivers

name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered	world_championships	podiums	highest_race_finish	highest_grid_position	career_p
Max Verstappen	VER	https://media.api-sports.io/formula-1/drivers/25.png	Dutch	Netherlands	1997-09-30	Hasselt, Belgium	1	189	3	101	57	1	2

Screenshot of min and max filters working

Checklist Items (0)

SpeedsterStats Home Profile Admin Driver Management Logout

List Drivers

Name	Nationality	Country	Driver #	Min GPs	Max GPs
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Min Points	Max Points	Sort	Order	Limit	
<input type="text"/>	<input type="text"/>	<input type="button" value="GPs"/>	<input type="button" value="+"/>	<input type="button" value="10"/>	

All Drivers

name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered	world_championships	podiums	highest_race_finish	highest_grid_position	career_p
Valterri Bottas	BOT	https://media.api-sports.io/formula-1/drivers/5.png	Finnish	Finland	1989-06-28	Nastola, Finland	77	226	0	67	10	1	1797.00

Screenshot of min and max podiums working

Checklist Items (0)

SpeedsterStats Home Profile Admin Driver Management Logout

SpeedsterStats

[Home](#) [Profile](#) [Admin](#) ▾

[Driver Management](#) ▾

[Logout](#)

List Drivers

Name	Nationality	Country	Driver #	Min GPs	Max GPs
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Min Points	Max Points	Sort	Order	Limit	
<input type="text"/> 500	<input type="text"/> 600	<input type="button" value="GPs"/>	<input type="button" value="+"/>	<input type="text"/> 10	

[Filter](#) [Clear](#)

All Drivers

name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered	world_championships	podiums	highest_race_finish	highest_grid_position	caree
Nico Hulkenberg	HUL	https://media.apis.sports.io/formula-1/drivers/6.png	German	Germany	1987-08-19	Emmerich am Rhein, Germany	27	210	0	0	0	1	533.0

Screenshot of min and max points working

Checklist Items (0)

Name	Nationality	Country	Driver #	Min GPs	Max GPs
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Min Points	Max Points	Sort	Order	Limit	
<input type="text"/> Min Points	<input type="text"/> Max Points	<input type="button" value="GPs"/>	<input type="button" value="-"/>	<input type="text"/> 10	

[Filter](#) [Clear](#)

All Drivers

name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered	world_championships	podiums	highest_race_finish	highest_grid_position	
Fernando Alonso	ALO	https://media.apis.sports.io/formula-1/drivers/4.png	Spanish	Spain	1981-07-29	Oviedo, Spain	14	384	2	106	32	1	533.0
Lewis Hamilton	HAM	https://media.apis.sports.io/formula-1/drivers/20.png	British	United Kingdom	1985-01-07	Stevenage, England	44	336	7	197	103	1	533.0

Screenshot of sort GPs working

Checklist Items (0)

List Drivers

Name	Nationality	Country	Driver #	Min GPs	Max GPs
Name	Nationality	Country	Driver #	Min GPs	Max GPs
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min Points	Max Points	Sort	Order	Limit	
Min Points	Max Points	WCs	-	10	

Filter **Clear**

All Drivers

name	abbr	image	nationality	country	birthdate	birthplace	number	grands.prix_entered	world.championships	podiums	highest_race_finish	highest_grid_position
Michael Schumacher	MSC	https://media.api-sports.io/formula-1/drivers/46.png	German	Germany	1969-03-01		7		7	155		
Lewis Hamilton	HAM	https://media.api-sports.io/formula-1/drivers/20.png	British	United Kingdom	1985-01-07	Stevenage, England	44	336	7	197	103	1

Screenshot of WCs sort working

Checklist Items (0)

Remove

List Drivers

Name	Nationality	Country	Driver #	Min GPs	Max GPs
Name	Nationality	Country	Driver #	Min GPs	Max GPs
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min Points	Max Points	Sort	Order	Limit	
Min Points	Max Points	Wins	-	10	

Filter **Clear**

All Drivers

name	abbr	image	nationality	country	birthdate	birthplace	number	grands.prix_entered	world.championships	podiums	highest_race_finish	highest_grid_position
Lewis Hamilton	HAM	https://media.api-sports.io/formula-1/drivers/20.png	British	United Kingdom	1985-01-07	Stevenage, England	44	336	7	197	103	1
Max Verstappen	VER	https://media.api-sports.io/formula-1/drivers/25.png	Dutch	Netherlands	1997-09-30	Hasselt, Belgium	1	189	3	101	57	1

Screenshot of wins sort working

Checklist Items (0)

List Drivers

Name	Nationality	Country	Driver #	Min GPs	Max GPs
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Min Points	Max Points	Sort	Order	Limit	
<input type="text"/>	<input type="text"/>	<input type="button" value="Podiums"/>	<input type="button" value="-"/>	<input type="button" value="10"/>	

All Drivers

name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered	world_championships	podiums	highest_race_finish	highest_grid_pos
Lewis Hamilton	HAM	https://mediaapisports.io/formula-1/drivers/20.png	British	United Kingdom	1985-01-07	Stevenage, England	44	336	7	197	103	1
Michael Schumacher	MSC	https://mediaapisports.io/formula-1/drivers/46.png	German	Germany	1969-03-01		7		7	155		

Screenshot of podiums sort working

Checklist Items (0)

List Drivers

Name	Nationality	Country	Driver #	Min GPs	Max GPs
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Min Points	Max Points	Sort	Order	Limit	
<input type="text"/>	<input type="text"/>	<input type="button" value="Points"/>	<input type="button" value="-"/>	<input type="button" value="10"/>	

All Drivers

name	abbr	image	nationality	country	birthdate	birthplace	number	grands_prix_entered	world_championships	podiums	highest_race_finish	highest_grid_position	career_points	Actions
Lewis Hamilton	HAM	https://mediaapisports.io/formula-1/drivers/20.png	British	United Kingdom	1985-01-07	Stevenage, England	44	336	7	197	103	1	4640.50	<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="Delete"/>
Sebastian Vettel	VET	https://mediaapisports.io/formula-1/drivers/19.png	German	Germany	1987-07-03	Heppenheim, Germany	5	299	4	122	53	1	3087.00	<input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="Delete"/>

Screenshot of points sort working

Checklist Items (0)

List Drivers

Name	Nationality	Country	Driver #	Min GPs	Max GPs
Name	Nationality	Country	Driver #	Min GPs	Max GPs
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min Points	Max Points	Sort	Order	Limit	
Min Points	Max Points	GPs	-	2	

Filter **Clear**

All Drivers

name	abbr	image	nationality	country	birthdate	birthplace	number	grands.prix_entered	world.championships	podiums	highest_race_finish	highest_grid_position	car
Fernando Alonso	ALO	https://media.api-sports.io/formula-1/drivers/4.png	Spanish	Spain	1981-07-29	Oviedo, Spain	14	384	2	106	32	1	229
Lewis Hamilton	HAM	https://media.api-sports.io/formula-1/drivers/20.png	British	United Kingdom	1985-01-07	Stevenage, England	44	336	7	197	103	1	464

Screenshot of limit filter working

Checklist Items (0)

Task #2 - Points: 1

Text: Screenshots of the list page code

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Show the filter/sort form generation
<input type="checkbox"/> #2	1	Show the DB query and how the filter/sort is handled (including the restriction on the limit field)
<input type="checkbox"/> #3	1	Show how the output is generated and displayed
<input type="checkbox"/> #4	1	Show any restrictions like role guard or login checks
<input type="checkbox"/> #5	1	Include ucid/date comments for each code screenshot
<input type="checkbox"/> #6	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small Medium Large

A screenshot of a code editor window titled "File Edit Selection View Go Run Terminal Help". The tab bar shows "view_driver.php", "list_drivers.php", "styles.css", "delete_driver.php", "session_store.php", "input_field.php", "edit_driver.php", "table.php", and "create_driver.php". The main pane displays the following PHP code:

```
public_html > Project > admin > list_drivers.php > ...
...
1 <?php
2
3 require(__DIR__ . "/../../../../partials/nav.php");
4
5 if (!has_role("Admin")) {
6     flash("You do not have permission to view this page", "warning"); //ak2774, 4/15/2024
7     die(header("location: $BASE_PATH" . "/home.php"));
8 }
```

Screenshot of admin check code

Checklist Items (0)

A screenshot of a code editor window showing a large block of PHP code for building a search form. The code uses a foreach loop to generate form fields based on an array of input types and their properties.

```
//build search form
$form = [
    ["type" => "text", "name" => "name", "placeholder" => "Name", "label" => "Name", "include_margin" => false],
    ["type" => "text", "name" => "nationality", "placeholder" => "Nationality", "label" => "Nationality", "include_margin" => false],
    ["type" => "text", "name" => "country", "placeholder" => "Country", "label" => "Country", "include_margin" => false],
    ["type" => "number", "name" => "number", "placeholder" => "Driver #", "label" => "Driver #", "include_margin" => false],
    ["type" => "number", "name" => "grands_prix_entered_min", "placeholder" => "Min GPs", "label" => "Min GPs", "include_margin" => false],
    ["type" => "number", "name" => "grands_prix_entered_max", "placeholder" => "Max GPs", "label" => "Max GPs", "include_margin" => false],
    ["type" => "number", "name" => "world_championships_min", "placeholder" => "Min WCs", "label" => "Min WCs", "include_margin" => false],
    ["type" => "number", "name" => "world_championships_max", "placeholder" => "Max WCs", "label" => "Max WCs", "include_margin" => false],
    ["type" => "number", "name" => "highest_race_finish_min", "placeholder" => "Min Wins", "label" => "Min Wins", "include_margin" => false],
    ["type" => "number", "name" => "highest_race_finish_max", "placeholder" => "Max Wins", "label" => "Max Wins", "include_margin" => false],
    ["type" => "number", "name" => "podiums_min", "placeholder" => "Min Podiums", "label" => "Min Podiums", "include_margin" => false], //ak2774, 4/15/2024
    ["type" => "number", "name" => "podiums_max", "placeholder" => "Max Podiums", "label" => "Max Podiums", "include_margin" => false],
    ["type" => "number", "name" => "career_points_min", "placeholder" => "Min Points", "label" => "Min Points", "include_margin" => false],
    ["type" => "number", "name" => "career_points_max", "placeholder" => "Max Points", "label" => "Max Points", "include_margin" => false],
    ["type" => "select", "name" => "sort", "label" => "Sort", "options" => ["grands_prix_entered" => "GPs", "world_championships" => "WCs",
        "highest_race_finish" => "Wins", "podiums" => "Podiums", "career_points" => "Points"], "include_margin" => false],
    ["type" => "select", "name" => "order", "label" => "Order", "options" => ["asc" => "+", "desc" => "-"], "include_margin" => false],
    ["type" => "number", "name" => "limit", "label" => "Limit", "value" => "10", "include_margin" => false]
];
$ -> #11-38 $form =
```

Screenshot of form code

Checklist Items (0)

```

41
42
43
44
45 $query = "SELECT id, name, abbr, image, nationality, country, birthdate, birthplace, number, grands_prix_entered, world_championships, podiums,
46 highest_race_finish, highest_grid_position, career_points FROM `Drivers` WHERE 1=1";
47 $params = [];
48 $session_key = $_SERVER["SCRIPT_NAME"];
49 $is_clear=isset($_GET["clear"]);
50 if($is_clear){
51     session_delete($session_key);
52     unset($_GET["clear"]);
53     die(header("location: " . $session_key));
54 } <- #48-52 if($is_clear)
55 else{
56     $session_data = session_load($session_key);      //ak2774, 4/15/2024
57 }
58
59 if (count($_GET) == 0 && isset($session_data) && count($session_data) > 0) {
60     if ($session_data) (
61         $_GET = $session_data;
62     )
63 } <- #57-61 if (count($_GET) == 0 && isset($session_data) && count($sessi...
64

```

Screenshot of db query

Checklist Items (0)

```

File Edit Selection View Go Run Terminal Help ← → J D ak2774-h202-008
public_html > Project > admin > list_drivers.php > ...
66 if (count($_GET) > 0) {
67     session_save($session_key, $_GET);
68     $keys = array_keys($_GET);
69
70     foreach ($form as $k => $v) {
71         if (in_array($v["name"], $keys)) {
72             $form[$k]["value"] = $_GET[$v["name"]];
73         }
74     } <- #68-72 foreach ($form as $k => $v)
75     $name = se($_GET, "name", "", false);
76     if (!empty($name)) {
77         $query .= " AND name LIKE :name";
78         $params[":name"] = "%$name%";
79     }
80     $nationality = se($_GET, "nationality", "", false);
81     if (!empty($nationality)) {
82         $query .= " AND nationality LIKE :nationality"; //ak2774, 4/15/2024
83         $params[":nationality"] = "%$nationality%";
84     }
85     $country = se($_GET, "country", "", false);
86     if (!empty($country)) {
87         $query .= " AND country LIKE :country";
88         $params[":country"] = "%$country%";
89     }
90     $number = se($_GET, "number", "-1", false);
91     if (!empty($number) && $number > -1) {
92         $query .= " AND number = :number";
93         $params[":number"] = $number;
94     }

```

Screenshot of validation part 1

Checklist Items (0)

File Edit Selection View Go Run Terminal Help ↶ → ⌂ ak2774-it202-008

public_html > Project > admin > list_drivers.php ..

```

95 //gp's range
96 $grands_prix_entered_min = se($_GET, "grands_prix_entered_min", "-1", false);
97 if (!empty($grands_prix_entered_min) && $grands_prix_entered_min > -1) {
98     $query .= " AND grands_prix_entered >= :grands_prix_entered_min";
99     $params[":grands_prix_entered_min"] = $grands_prix_entered_min;
100 }
101 $grands_prix_entered_max = se($_GET, "grands_prix_entered_max", "-1", false);
102 if (!empty($grands_prix_entered_max) && $grands_prix_entered_max > -1) {
103     $query .= " AND grands_prix_entered <= :grands_prix_entered_max";
104     $params[":grands_prix_entered_max"] = $grands_prix_entered_max;
105 }
106 if($grands_prix_entered_min > $grands_prix_entered_max && !empty($grands_prix_entered_min) && !empty($grands_prix_entered_max)){
107     flash("Min GP's must be less than Max GP's", "warning");
108 }

110 //wc's range
111 $world_championships_min = se($_GET, "world_championships_min", "-1", false);
112 if (!empty($world_championships_min) && $world_championships_min > -1) {
113     $query .= " AND world_championships >= :world_championships_min";
114     $params[":world_championships_min"] = $world_championships_min;
115 }
116 $world_championships_max = se($_GET, "world_championships_max", "-1", false); //ak2774, 4/15/2024
117 if (!empty($world_championships_max) && $world_championships_max > -1) {
118     $query .= " AND world_championships <= :world_championships_max";
119     $params[":world_championships_max"] = $world_championships_max;
120 }
121 if($world_championships_min > $world_championships_max && !empty($world_championships_min) && !empty($world_championships_max)){
122     flash("Min WCs must be less than Max WCs", "warning");
123 }

125 //wins range
126 $highest_race_finish_min = se($_GET, "highest_race_finish_min", "-1", false);
127 if(!empty($highest_race_finish_min) && $highest_race_finish_min > -1){
128     $query .= " AND highest_race_finish >= :highest_race_finish_min";
129     $params[":highest_race_finish_min"] = $highest_race_finish_min;
130 }

```

Screenshot of validation part 2

Checklist Items (0)

File Edit Selection View Go Run Terminal Help ↶ → ⌂ ak2774-it202-008

public_html > Project > admin > list_drivers.php ..

```

131 $highest_race_finish_max = se($_GET, "highest_race_finish_max", "-1", false);
132 if(!empty($highest_race_finish_max) && $highest_race_finish_max > -1){
133     $query .= " AND highest_race_finish <= :highest_race_finish_max";
134     $params[":highest_race_finish_max"] = $highest_race_finish_max;
135 }
136 if($highest_race_finish_min > $highest_race_finish_max && !empty($highest_race_finish_min) && !empty($highest_race_finish_max)){
137     flash("Min Wins must be less than Max Wins", "warning");
138 }

139 //podiums range
140 $podiums_min = se($_GET, "podiums_min", "-1", false);
141 if (!empty($podiums_min) && $podiums_min > -1) {
142     $query .= " AND podiums >= :podiums_min";
143     $params[":podiums_min"] = $podiums_min;
144 }
145 $podiums_max = se($_GET, "podiums_max", "-1", false);
146 if (!empty($podiums_max) && $podiums_max > -1) {
147     $query .= " AND podiums <= :podiums_max";
148     $params[":podiums_max"] = $podiums_max;
149 }
150 if($podiums_min > $podiums_max && !empty($podiums_min) && !empty($podiums_max)){ //ak2774, 4/15/2024
151     flash("Min Podiums must be less than Max Podiums", "warning");
152 }
153 }

155 //points range
156 $career_points_min = se($_GET, "career_points_min", "-1", false);
157 if (!empty($career_points_min) && $career_points_min > -1) {
158     $query .= " AND career_points >= :career_points_min";
159     $params[":career_points_min"] = $career_points_min;
160 }
161 $career_points_max = se($_GET, "career_points_max", "-1", false);
162 if (!empty($career_points_max) && $career_points_max > -1) {
163     $query .= " AND career_points <= :career_points_max";
164     $params[":career_points_max"] = $career_points_max;
165 }

```

Screenshot of validation part 3

Checklist Items (0)

A screenshot of a code editor window titled "ak2774-it202-008". The file being edited is "list_drivers.php". The code handles user input for sorting and ordering, sets a limit of 10 results, and logs an error if the limit is invalid. It then prepares a PDO statement to fetch all drivers from the database.

```
166     if($career_points_min > $career_points_max && !empty($career_points_min) && !empty($career_points_max)){
167         flash("Min Points must be less than Max Points", "warning");
168     }
169
170     //sort and order
171     $sort = se($_GET, "sort", "grands_prix_entered", false);
172     if (!in_array($sort, ["grands_prix_entered", "world_championships", "highest_race_finish", "podiums", "career_points"])){
173         $sort = "grands_prix_entered";
174     }
175     $order = se($_GET, "order", "desc", false); //ak2774, 4/15/2024
176     if (!in_array($order, ["asc", "desc"])){
177         $order = "desc";
178     }
179
180     $query .= " ORDER BY $sort $order";
181
182     try {
183         $limit = (int)se($_GET, "limit", "10", false);
184     } catch (Exception $e) {
185         $limit = 10;
186     }
187     if ($limit < 1 || $limit > 100) {
188         $limit = 10;
189     }
190
191     $query .= " LIMIT $limit";
192 } <- #64-190 if (count($_GET) > 0)
193
194
```

Screenshot of validation part 4

Checklist Items (0)

A screenshot of a code editor window titled "ak2774-it202-008". The file being edited is "list_drivers.php". The code handles user input for sorting and ordering, sets a limit of 10 results, and logs an error if the limit is invalid. It then prepares a PDO statement to fetch all drivers from the database.

```
195
196
197     $db = getDB();
198     $stmt = $db->prepare($query);
199     $results = [];
200     try {
201         $stmt->execute($params);
202         $r = $stmt->fetchAll(); //ak2774, 4/15/2024
203         if ($r) {
204             $results = $r;
205         }
206     } catch (PDOException $e) {
207         error_log("Error fetching drivers " . var_export($e, true));
208         flash("Unhandled error occurred ", "danger");
209     }
210
211     $table = ["data" => $results, "title" => "All Drivers", "ignored_columns" => ["id"],
212     "edit_url" => get_url("admin/edit_driver.php"),
213     "delete_url" => get_url("admin/delete_driver.php"),
214     "view_url" => get_url("admin/view_driver.php")
215 ];
216 ?>
```

Screenshot of DB insertion

Checklist Items (0)

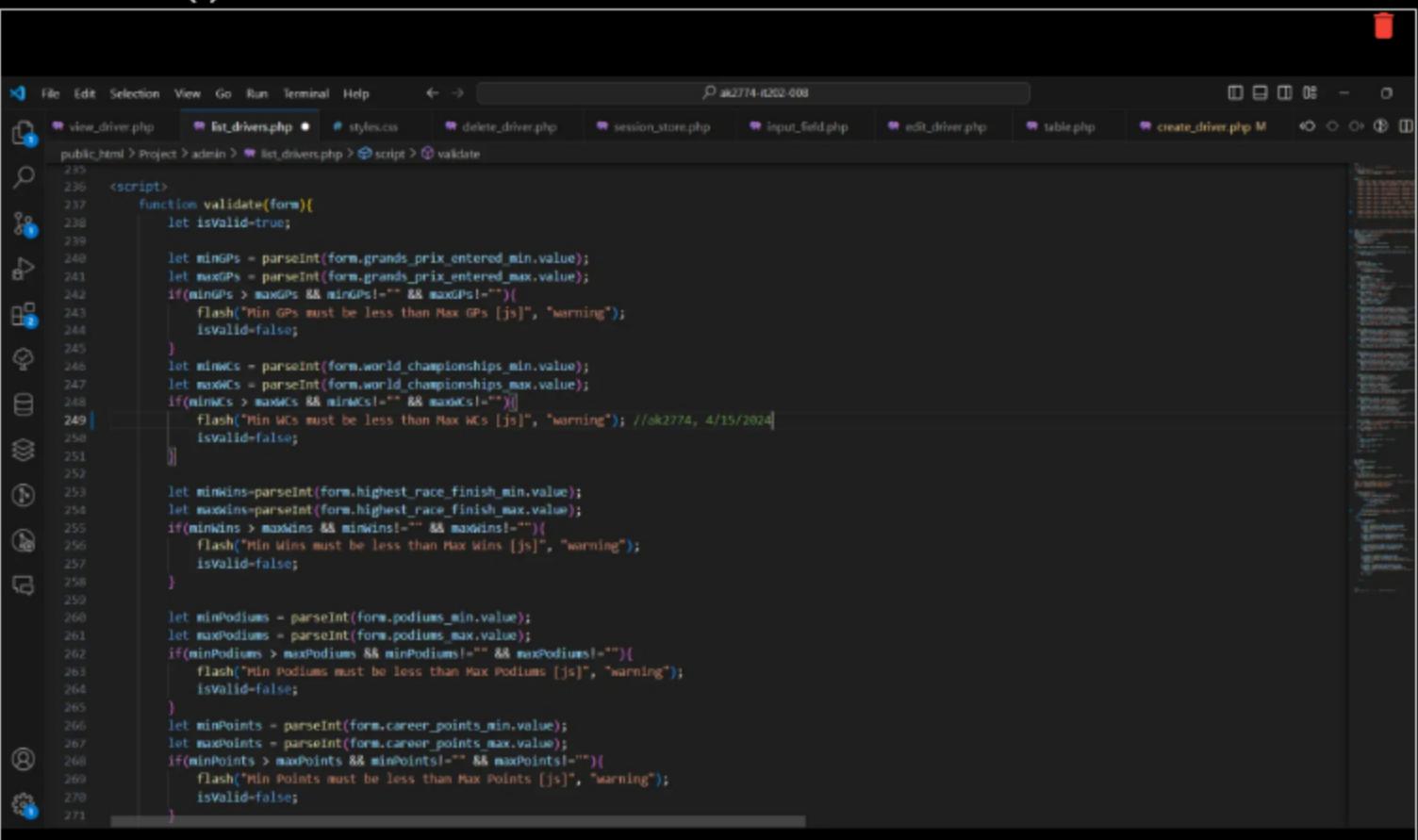
```

<div class="container-fluid">
    <h3>List Drivers</h3>
    <form method="GET" onsubmit="return validate(this);">
        <div class="row mb-3" style="align-items: flex-end;">
            <?php foreach ($form as $k => $v) : ?>
                <div class="col col-2">
                    <?php render_input($v); ?>
                </div>
            <?php endforeach; ?>
        </div>
        <?php render_button(["text" => "Search", "type" => "submit", "text" => "Filter"]); //ak2774, 4/15/2024?>
        <a href="?clear" class="btn btn-secondary">Clear</a>
    </form>
    <?php render_table($table); ?>
</div>

```

Screenshot of css code

Checklist Items (0)



```

public_html > Project > admin > list_drivers.php > script > validate
235 <script>
236     function validate(form){
237         let isValid=true;
238
239         let minGPs = parseInt(form.grands_prix_entered_min.value);
240         let maxGPs = parseInt(form.grands_prix_entered_max.value);
241         if(minGPs > maxGPs || minGPs!="" && maxGPs!=""){
242             flash("Min GPs must be less than Max GPs [js]", "warning");
243             isValid=false;
244         }
245
246         let minWCs = parseInt(form.world_championships_min.value);
247         let maxWCs = parseInt(form.world_championships_max.value);
248         if(minWCs > maxWCs || minWCs!="" && maxWCs!=""){
249             flash("Min WCs must be less than Max WCs [js]", "warning"); //ak2774, 4/15/2024
250             isValid=false;
251         }
252
253         let minWins=parseInt(form.highest_race_finish_min.value);
254         let maxWins=parseInt(form.highest_race_finish_max.value);
255         if(minWins > maxWins || minWins!="" && maxWins!=""){
256             flash("Min Wins must be less than Max Wins [js]", "warning");
257             isValid=false;
258         }
259
260         let minPodiums = parseInt(form.podiums_min.value);
261         let maxPodiums = parseInt(form.podiums_max.value);
262         if(minPodiums > maxPodiums || minPodiums!="" && maxPodiums!=""){
263             flash("Min Podiums must be less than Max Podiums [js]", "warning");
264             isValid=false;
265         }
266         let minPoints = parseInt(form.career_points_min.value);
267         let maxPoints = parseInt(form.career_points_max.value);
268         if(minPoints > maxPoints || minPoints!="" && maxPoints!=""){
269             flash("Min Points must be less than Max Points [js]", "warning");
270             isValid=false;
271         }
    
```

Screenshot of js validation part 1

Checklist Items (0)

The screenshot shows a code editor with a dark theme. The file being edited is `validate.js`. The code contains a function `validate` that checks various input fields for numerical values and compares them. It uses `parseInt` to parse strings into integers. It includes conditional statements to check if one value is greater than another and if either value is empty. If conditions are violated, it calls a `flash` function to display a warning message. The code also includes a `require_once` statement at the bottom to include a partial file.

```
File Edit Selection View Go Run Terminal Help ← → P ok2774-t202-008
view_driver.php list_drivers.php styles.css delete_driver.php session_store.php input_field.php edit_driver.php table.php create_driver.php M eO O C
public_html > Project > admin > list_drivers.php > script > validate
236 <script>
237     function validate(form){
238         let minWins=parseInt(form.highest_race_finish_min.value);
239         let maxWins=parseInt(form.highest_race_finish_max.value);
240         if(minWins > maxWins && minWins!="" && maxWins!=""){
241             flash("Min Wins must be less than Max Wins [js]", "warning");
242             isValid=false;
243         }
244
245         let minPodiums = parseInt(form.podiums_min.value);
246         let maxPodiums = parseInt(form.podiums_max.value);
247         if(minPodiums > maxPodiums && minPodiums!="" && maxPodiums!=""){
248             flash("Min Podiums must be less than Max Podiums [js]", "warning"); //ok2774, 4/15/2024
249             isValid=false;
250         }
251
252         let minPoints = parseInt(form.career_points_min.value);
253         let maxPoints = parseInt(form.career_points_max.value);
254         if(minPoints > maxPoints && minPoints!="" && maxPoints!=""){
255             flash("Min Points must be less than Max Points [js]", "warning");
256             isValid=false;
257         }
258     return isValid;
259 } <- #235-271 function validate(form)
260
261 </script>
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282 <?php
283 require_once(__DIR__ . "/../../../../partials/flash.php");
284 ?>
```

Screenshot of js validation part 2

Checklist Items (0)

Task #3 - Points: 1

Text: Explain how the page works

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Provide the high-level steps of how the filter/sorting works and how data is displayed
<input type="checkbox"/> #2	1	Summarize what you're showing on the screen
<input type="checkbox"/> #3	1	Mention your design/style choice
<input type="checkbox"/> #4	1	Mention which users can interact with the view, edit, and delete links

Response:

The PHP script generates a dynamic form for filtering and searching drivers based on various criteria such as name, nationality, country, driver number, career statistics like the number of Grands Prix entered, World Championships won, podiums achieved, highest race finish, and career points. Users can input specific values or ranges for these criteria to filter the displayed results. Additionally, users can select the sorting parameter and order (ascending or descending) for the displayed data.

The filtered results are displayed in a table format, presenting essential details of each driver, including their name, abbreviation, image, nationality, country, birthdate, birthplace, driver number, and career statistics. The table allows users to interact with each entry through view, edit, and delete links, providing comprehensive control over the displayed data.

The design choice emphasizes simplicity and clarity, with a clean layout for both the filter form and the displayed table. Input fields and buttons are arranged in a user-friendly manner to facilitate easy navigation and interaction.

The styling maintains consistency with the overall aesthetic of the application, ensuring a seamless user experience.

Users with the appropriate permissions, specifically those with the "Admin" role, can interact with the filter form, view, edit, and delete links to manage the displayed driver data. This ensures that authorized users can perform necessary actions like viewing detailed driver information, editing existing entries, or removing outdated records directly from the interface.

Task #4 - Points: 1

Text: Add related links

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Include the heroku prod link for this page
<input type="checkbox"/> #2	1	Add the pull request link for the branch related to this feature Note: the link should end with /pull/#. Same pull request shouldn't be used for each feature

URL #1

https://ak2774-prod-76402b453c05.herokuapp.com/Project/admin/list_drivers.php

URL #2

<https://github.com/abhinavkarthikn/ak2774-it202-008/pull/65>

View Details Page (single entity) (1 pt.)

[^COLLAPSE ^](#)

Task #1 - Points: 1

Text: Screenshots of the details page

Checklist

*The checkboxes are for your own tracking

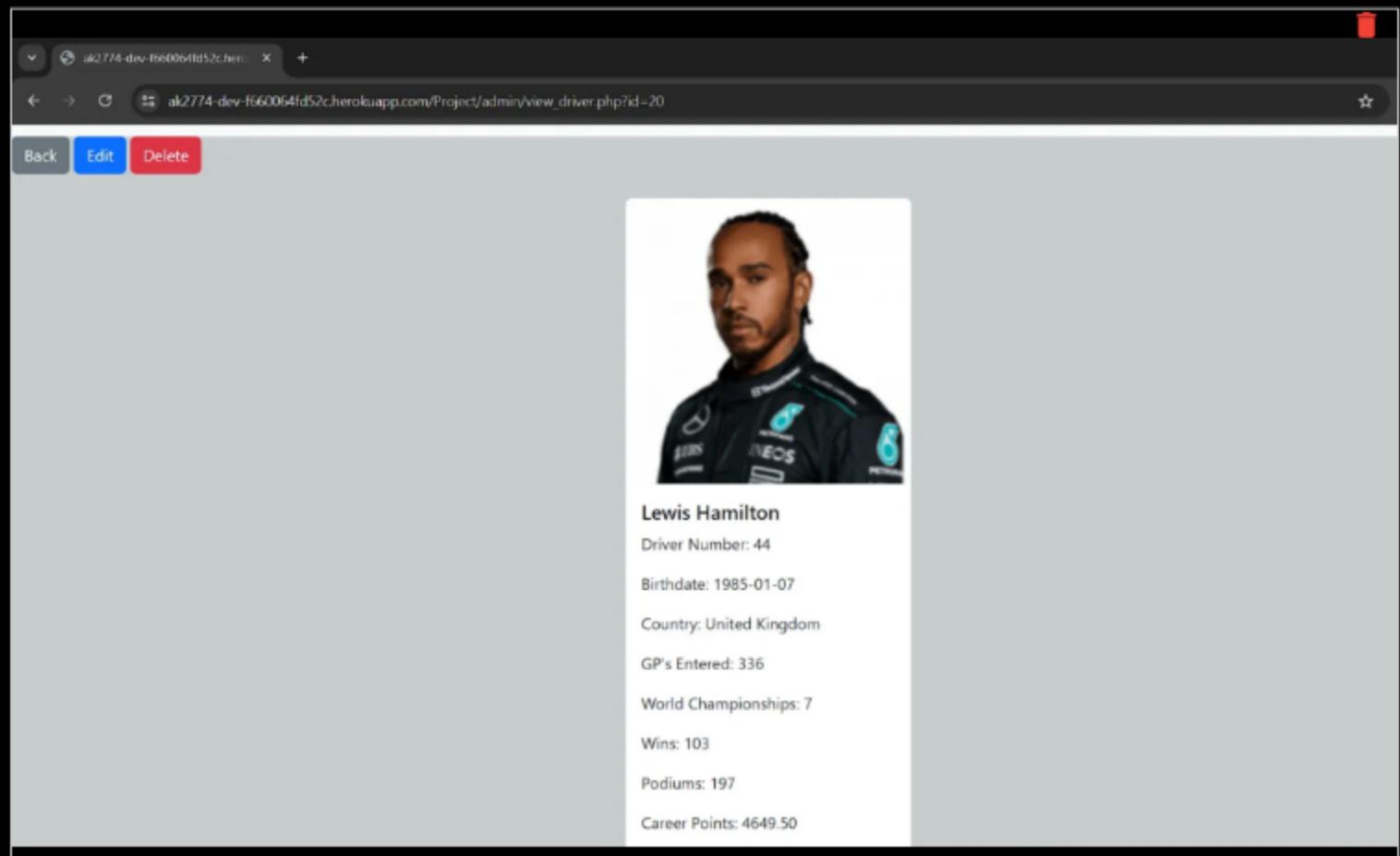
#	Points	Details
<input type="checkbox"/> #1	1	Entity should be fetch by id (via the url)
<input type="checkbox"/> #2	1	A missing id should redirect back to the list page with an applicable message
<input type="checkbox"/> #3	1	Design/Style should be considered (i.e., bootstrap, custom css, etc)
<input type="checkbox"/> #4	1	Data shown should be more detailed/inclusive than the summary view
<input type="checkbox"/> #5	1	There should be a link to edit the entity (this may be an admin-only thing, but it should be

		present for the respective role)
<input type="checkbox"/> #6	1	There should be a link to delete the entity (this may be an admin-only thing, but it should be present for the respective role)
<input type="checkbox"/> #7	1	Make sure the heroku dev url is visible in the address bar
<input type="checkbox"/> #8	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

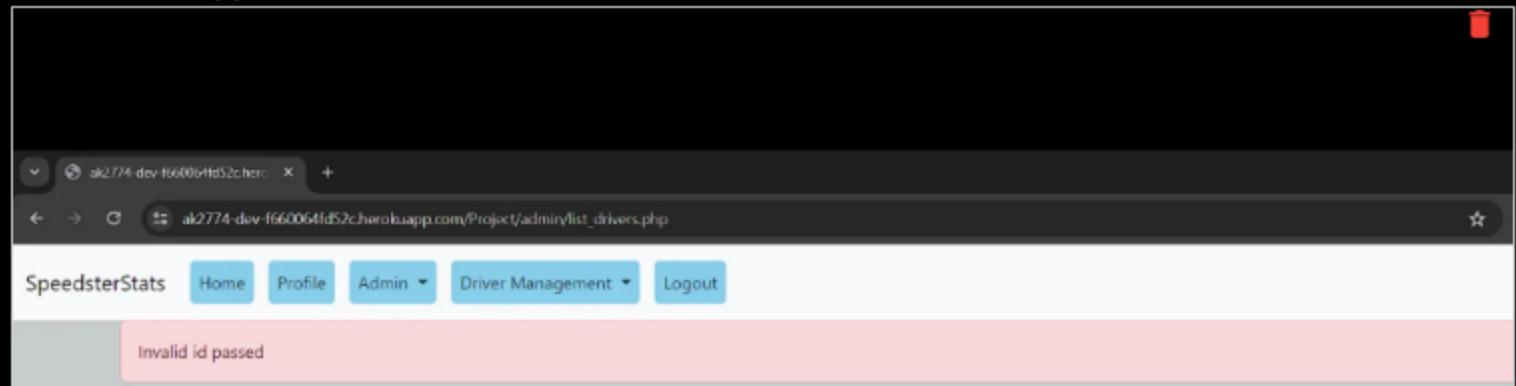
Small Medium Large



Lewis Hamilton
Driver Number: 44
Birthdate: 1985-01-07
Country: United Kingdom
GP's Entered: 336
World Championships: 7
Wins: 103
Podiums: 197
Career Points: 4649.50

Screenshot of view single entity page with links to edit and delete

Checklist Items (0)



Invalid id passed

Name	Nationality	Country	Driver #	Min GPs	Max GPs
Name	Nationality	Country	Driver #	Min GPs	Max GPs
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums
Min WCs	Max WCs	Min Wins	Max Wins	Min Podiums	Max Podiums

List Drivers

Min Points	Max Points	Sort	Order	Limit
<input type="text"/>	<input type="text"/>	GPs	+	10
<input type="button" value="Filter"/>		<input type="button" value="Clear"/>		

All Drivers

Screenshot of missing id

Checklist Items (0)

Task #2 - Points: 1

Text: Screenshots of the details page code

Checklist

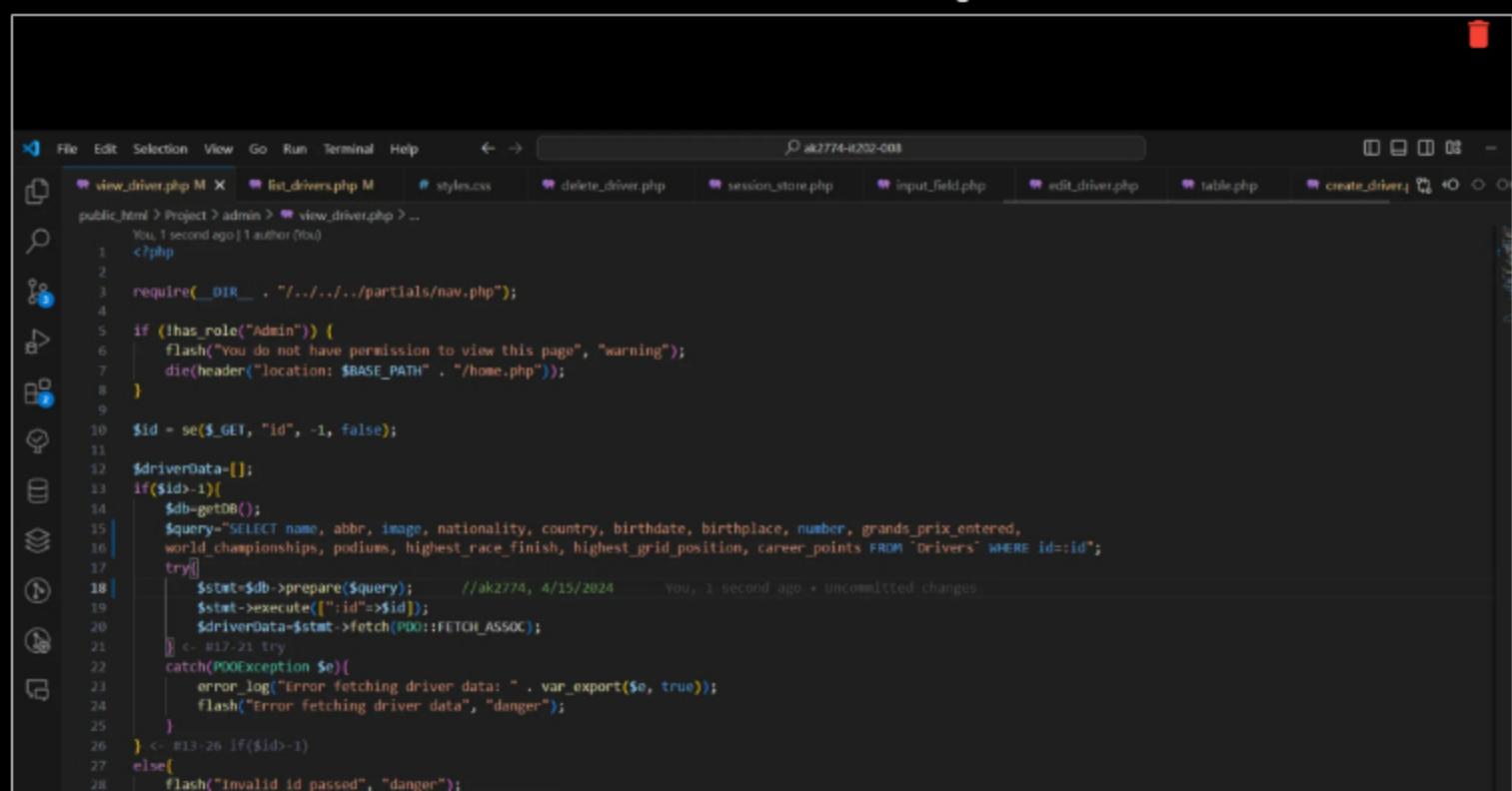
*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Show how id is fetched
<input type="checkbox"/> #2	1	Show the DB query to get the record
<input type="checkbox"/> #3	1	Show the code related to presenting the data and showing the links
<input type="checkbox"/> #4	1	Include ucid/date comments for each code screenshot
<input type="checkbox"/> #5	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small Medium Large



```

public_html > Project > admin > view_driver.php < ...
You, 1 second ago | 1 author (You)
<?php
require(__DIR__ . "/../../../../partials/nav.php");
if (!has_role("Admin")) {
    flash("you do not have permission to view this page", "warning");
    die(header("location: $BASE_PATH" . "/home.php"));
}
$id = se($_GET, "id", -1, false);
$driverData = [];
if ($id > -1) {
    $db = getDB();
    $query = "SELECT name, abbr, image, nationality, country, birthdate, birthplace, number, grands_prix_entered,
    world_championships, podiums, highest_race_finish, highest_grid_position, career_points FROM `Drivers` WHERE id=:id";
    try {
        $stmt = $db->prepare($query);
        $stmt->execute([":id" => $id]);
        $driverData = $stmt->fetch(PDO::FETCH_ASSOC);
    } <- #17-21 try
    catch(PDOException $e){
        error_log("Error fetching driver data: " . var_export($e, true));
        flash("Error fetching driver data", "danger");
    }
} <- #13-26 if($id>-1)
else{
    flash("Invalid id passed", "danger");
}

```

```
29 | die(header("location: " . get_url("admin/list_drivers.php")));  
30 | }
```

Screenshot of admin required and id fetch

Checklist Items (0)

```
...  
31 ?>  
32 <div>  
33   <a href=<?php echo get_url("admin/list_drivers.php"); ?>" class="btn btn-secondary">Back</a>  
34   <a href=<?php echo get_url("admin/edit_driver.php?id=" . $id); ?>" class="btn btn-primary">Edit</a>  
35   <a href=<?php echo get_url("admin/delete_driver.php?id=" . $id); ?>" class="btn btn-danger">Delete</a>  
36 </div>  
37 <div class="container mt-4 d-flex justify-content-center">  
38  
39   <div class="card" style="width: 18rem;">  
40     <?php if(!empty($driverData["image"])):>  
41       <img src=<?php echo $driverData["image"];?>" class="card-img-top" alt="Driver Image">  
42     <?php endif;?>  
43  
44     <div class="card-body">  
45       <h5 class="card-title"><?php echo($driverData["name"]);?></h5>  
46       <p class="card-text">Driver Number: <?php safer_echo($driverData["number"]); //ak2774, 4/15/2024?></p> You, 1 minute ago  
47       <p class="card-text">Birthdate: <?php safer_echo($driverData["birthdate"]);?></p>  
48       <p class="card-text">Country: <?php safer_echo($driverData["country"]);?></p>  
49       <p class="card-text">GP's Entered: <?php safer_echo($driverData["grands_prix_entered"]);?></p>  
50       <p class="card-text">World Championships: <?php safer_echo($driverData["world_championships"]);?></p>  
51       <p class="card-text">Wins: <?php safer_echo($driverData["highest_race_finish"]);?></p>  
52       <p class="card-text">Podiums: <?php safer_echo($driverData["podiums"]);?></p>  
53       <p class="card-text">Career Points: <?php safer_echo($driverData["career_points"]);?></p>  
54     </div>  
55   </div>  
56 </div>  
57  
58 <?php require(__DIR__ . "/../../../../partials/flash.php");?>  
59
```

Screenshot of css

Checklist Items (0)

Task #3 - Points: 1

Text: Explain how the page works

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Provide the high-level steps for handling the DB lookup and presenting the data

Response:

The script first checks if the user has admin privileges; if not, they're redirected with a warning. Then, it retrieves the driver's ID from the URL parameters and fetches their details from the database. The data is then presented in a card layout, including the driver's name, image, birthdate, nationality, career statistics, and other relevant information. Buttons for navigation, editing, and deleting the driver entry are provided. If any errors occur during the process, appropriate messages are displayed to the user.

Task #4 - Points: 1

Text: Add related links

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Include the heroku prod link for this page
<input type="checkbox"/> #2	1	Add the pull request link for the branch related to this feature Note: the link should end with /pull/#. Same pull request shouldn't be used for each feature

URL #1

https://ak2774-prod-76402b453c05.herokuapp.com/Project/admin/view_driver.php?id=20

URL #2

<https://github.com/abhinavkarthikn/ak2774-it202-008/pull/66>

Edit Data Page (2 pts.)

[^COLLAPSE ^](#)

Task #1 - Points: 1

Text: Screenshots of the edit page

Checklist

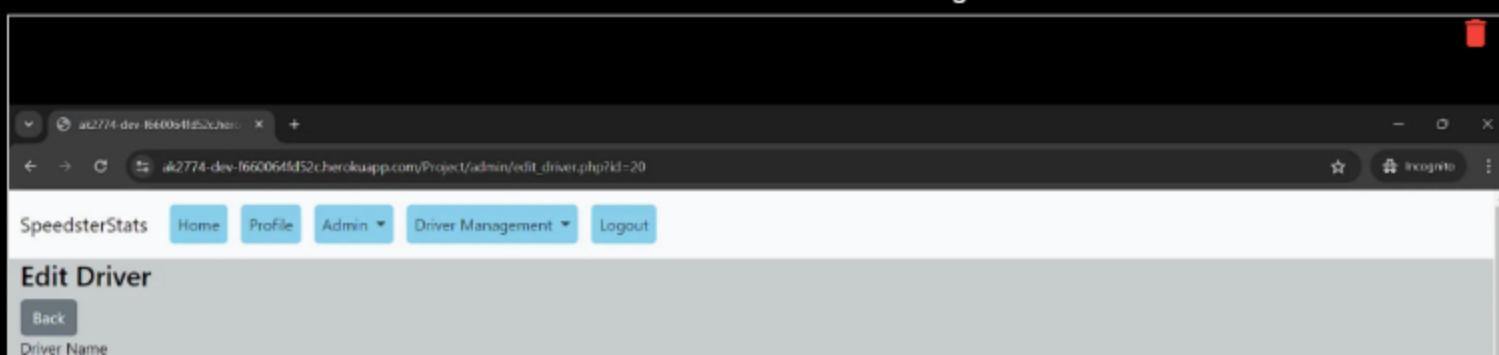
*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Show before and after screenshots of data you'll edit
<input type="checkbox"/> #2	1	Show examples of validation messages
<input type="checkbox"/> #3	1	Show an example of successful edit messages
<input type="checkbox"/> #4	1	Design/Style should be considered (i.e., bootstrap, custom css, etc)
<input type="checkbox"/> #5	1	Make sure the heroku dev url is visible in the address bar
<input type="checkbox"/> #6	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small Medium Large



Lewis Hamilton

Driver Abbr

HAM

Image URL

<https://media.api-sports.io/formula-1/drivers/20.png>

Nationality

British

Country

United Kingdom

DOB

01/07/1985

Screenshot of edit page part 1

Checklist Items (0)

Stevenage, England

Driver Number

44

GPs Entered

336

World Championships

7

Podiums

197

Wins

103

Highest Grid Position

1

Career Points

4649.50

Update

Screenshot of edit page part 2

Checklist Items (0)

SpeedsterStats

Name must not be empty

Abbr must not be empty

Image must not be empty

Nationality must not be empty

Country must not be empty

Birthday must not be empty

Birthplace must not be empty

Number must not be empty

GPs Entered must not be empty

Screenshot of PHP validation part 1

Checklist Items (0)

A screenshot of a web browser window titled "ak2774-dev-f660064fd52c.herokuapp.com". The URL in the address bar is "ak2774-dev-f660064fd52c.herokuapp.com/Project/admin/edit_driver.php?id=20". The page content displays several validation error messages in yellow boxes:

- GPs Entered must not be empty
- World Championships must not be empty
- Podiums must not be empty
- Wins must not be empty
- Highest Grid Position must not be empty
- Career Points must not be empty

The page has a header "Edit Driver" and a "Back" button.

Screenshot of PHP validation part 2

Checklist Items (0)

A screenshot of a web browser window titled "ak2774-dev-f660064fd52c.herokuapp.com". The URL in the address bar is "ak2774-dev-f660064fd52c.herokuapp.com/Project/admin/edit_driver.php?id=2". The page content displays a single validation error message in a yellow box:

- Name must not be empty [js]

The page has a header "SpeedsterStats" and a menu icon.

Abbr must not be empty [js]

Image must not be empty [js]

Nationality must not be empty [js]

Country must not be empty [js]

Birthday must not be empty [js]

Birthplace must not be empty [js]

Number must not be empty [js]

GPs Entered must not be empty [js]

Screenshot of js validation part 1

Checklist Items (0)

The screenshot shows a web browser window with the URL ak2774-dev-f660064fd52c.herokuapp.com/Project/admin/edit_driver.php?id=2. The page title is "Edit Driver". There are six validation error messages displayed in yellow boxes:

- GPs Entered must not be empty [js]
- World Championships must not be empty [js]
- Podiums must not be empty [js]
- Highest race finish must not be empty [js]
- Highest Grid Position must not be empty [js]
- Career Points must not be empty [js]

Below the errors, there is a "Back" button and a "Driver Name" input field.

Screenshot of JS validation part 2

Checklist Items (0)

The screenshot shows a web browser window with the URL ak2774-dev-f660064fd52c.herokuapp.com/Project/admin/edit_driver.php?id=2. The page title is "SpeedsterStats". There are no validation errors displayed on this screen.

Edit Driver

[Back](#)

Driver Name

Kevin Magnussen

Driver Abbr

MAG

Image URL

https://media.api-sports.io/formula-1/drivers/2.png

Nationality

Danis

Comments

Screenshot of successful edit message

Checklist Items (0)

Task #2 - Points: 1

Text: Screenshots of edit page code

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Form should have correct data types for each property being requested
<input type="checkbox"/> #2	1	Form should have correct validation for each field (HTML, JS, and PHP)
<input type="checkbox"/> #3	1	Successful edit should have a user-friendly message
<input type="checkbox"/> #4	1	Any errors should have user-friendly messages
<input type="checkbox"/> #5	1	Include any other rules like role guards and login checks
<input type="checkbox"/> #6	1	Include ucid/date comments for each code screenshot
<input type="checkbox"/> #7	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

[Small](#)[Medium](#)[Large](#)

File Edit Selection View Go Run Terminal Help ← → ↻ dk2774-t202-008

public_html > Project > admin > edit_driver.php ..

```

1  <?php
2
3  require(__DIR__ . "/../../../../partials/nav.php");
4
5  if(!has_role("Admin")){
6
7      header("Location: " . URL . "error/?e=403");
8      exit();
9  }
10
11  $driver = get_driver($_GET["id"]);
12
13  if(!$driver){
14      header("Location: " . URL . "error/?e=404");
15      exit();
16  }
17
18  if($_POST) {
19      $driver["name"] = $_POST["name"];
20      $driver["abbr"] = $_POST["abbr"];
21      $driver["image"] = $_POST["image"];
22      $driver["nationality"] = $_POST["nationality"];
23
24      update_driver($driver);
25
26      header("Location: " . URL . "edit_driver?msg=1");
27      exit();
28  }
29
30
31  include("partials/header.php");
32
33  echo "

# Edit Driver


34
35  
36
37      

38          Driver Name
39          
40


41
42      

43          Driver Abbr
44          
45


46
47      

48          Image URL
49          
50


51
52      

53          Nationality
54          
55


56
57      

58          <input type='submit' value='Save' style='width: 100%; height: 100%; background-color: #007bff; color: white; border: none; font-size: 1.2em; padding: 0.5em; border-radius: 0.3em; margin-top: 10px;"/>
59


60  
61
62  include("partials/footer.php");
63
64  
65  

```

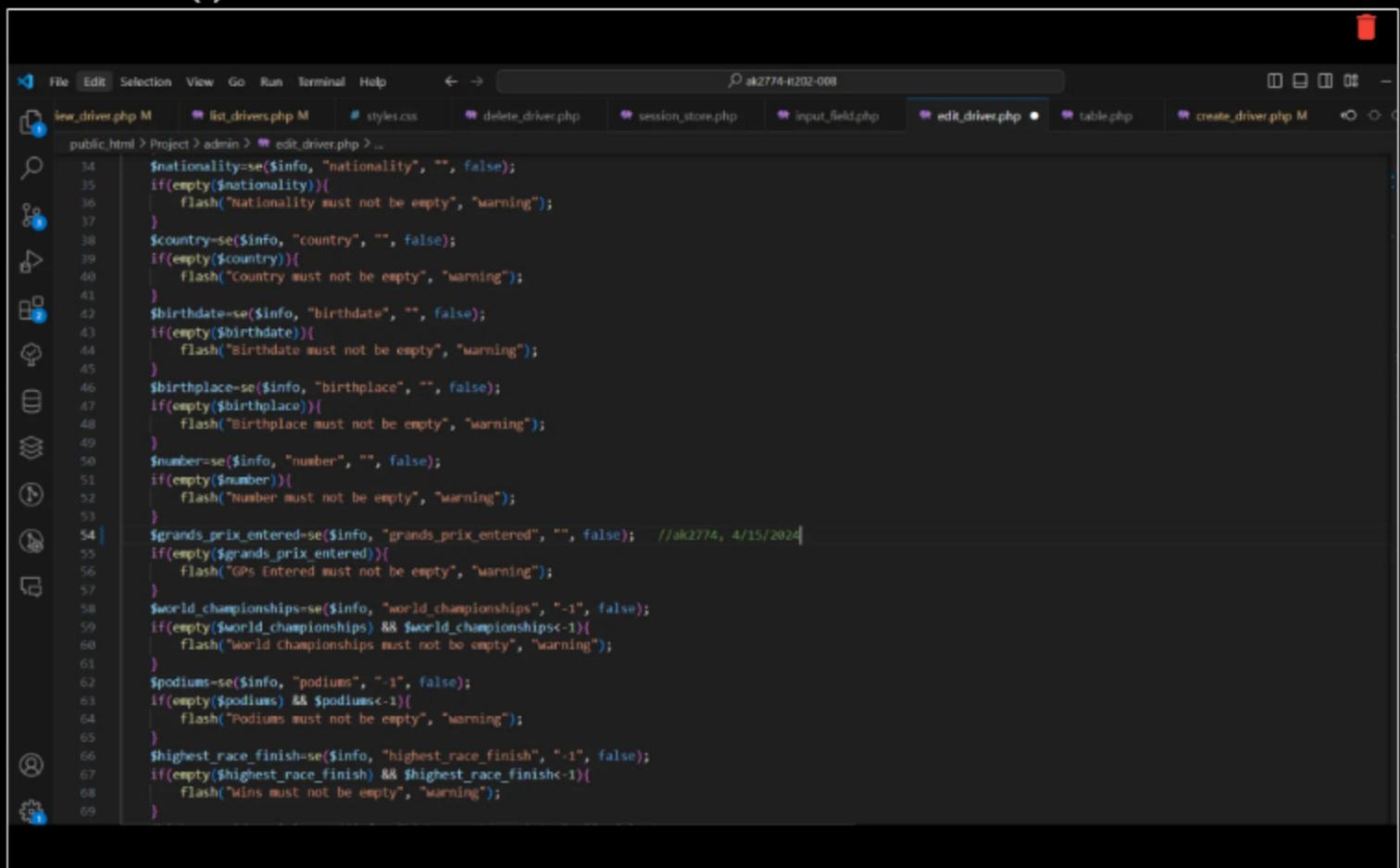
```

6     flash("You do not have permission to view this page", "warning");
7     die(header("location: $BASE_PATH" . "/home.php"));
8 }
9 ?>
10 <?php
11 $id = se($_GET, "id", -1, false);
12 //TODO handle driver fetch
13 if(isset($_POST["name"])){
14     foreach($_POST as $k=>$v){
15         if(in_array($k, ["name", "abbr", "image", "nationality", "country", "birthdate", "birthplace", "number", "grands_prix_entered",
16             "world_championships", "podiums", "highest_race_finish", "highest_grid_position", "career_points"] ) ){
17             unset($_POST[$k]);
18         }
19         $info+=$info;
20     } <- #14-19 foreach($_POST as $k=>$v)
21
22     $name=se($info, "name", "", false); //ak2774, 4/15/2024]
23     if(empty($name)){
24         flash("Name must not be empty", "warning");
25     }
26     $abbr=se($info, "abbr", "", false);
27     if(empty($abbr)){
28         flash("abbr must not be empty", "warning");
29     }
30     $image=se($info, "image", "", false);
31     if(empty($image)){
32         flash("Image must not be empty", "warning");
33     }
34     $nationality=se($info, "nationality", "", false);

```

Screenshot of admin validation, database info and part 1 of php validation

Checklist Items (0)



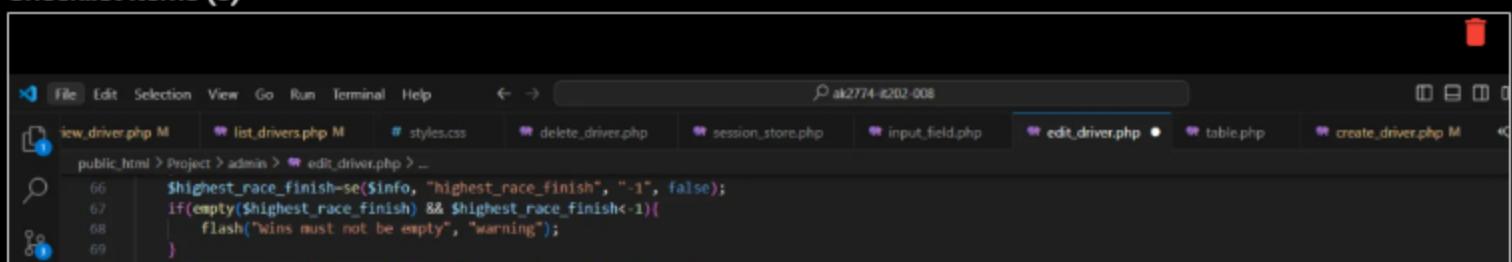
```

File Edit Selection View Go Run Terminal Help ← → ak2774-k202-008
new_driver.php M list_drivers.php M styles.css delete_driver.php session_store.php input_field.php edit_driver.php table.php create_driver.php M
public_html > Project > admin > edit_driver.php > ...
34     $nationality=se($info, "nationality", "", false);
35     if(empty($nationality)){
36         flash("Nationality must not be empty", "warning");
37     }
38     $country=se($info, "country", "", false);
39     if(empty($country)){
40         flash("Country must not be empty", "warning");
41     }
42     $birthdate=se($info, "birthdate", "", false);
43     if(empty($birthdate)){
44         flash("Birthdate must not be empty", "warning");
45     }
46     $birthplace=se($info, "birthplace", "", false);
47     if(empty($birthplace)){
48         flash("Birthplace must not be empty", "warning");
49     }
50     $number=se($info, "number", "", false);
51     if(empty($number)){
52         flash("Number must not be empty", "warning");
53     }
54     $grands_prix_entered=se($info, "grands_prix_entered", "", false); //ak2774, 4/15/2024]
55     if(empty($grands_prix_entered)){
56         flash("GP's Entered must not be empty", "warning");
57     }
58     $world_championships=se($info, "world_championships", "-1", false);
59     if(empty($world_championships) && $world_championships<-1){
60         flash("World Championships must not be empty", "warning");
61     }
62     $podiums=se($info, "podiums", "-1", false);
63     if(empty($podiums) && $podiums<-1){
64         flash("Podiums must not be empty", "warning");
65     }
66     $highest_race_finish=se($info, "highest_race_finish", "-1", false);
67     if(empty($highest_race_finish) && $highest_race_finish<-1){
68         flash("Wins must not be empty", "warning");
69     }

```

Screenshot of php validation part 2

Checklist Items (0)



```

File Edit Selection View Go Run Terminal Help ← → ak2774-k202-008
new_driver.php M list_drivers.php M styles.css delete_driver.php session_store.php input_field.php edit_driver.php table.php create_driver.php M
public_html > Project > admin > edit_driver.php > ...
66     $highest_race_finish=se($info, "highest_race_finish", "-1", false);
67     if(empty($highest_race_finish) && $highest_race_finish<-1){
68         flash("Wins must not be empty", "warning");
69     }

```

```

70 $highest_grid_position=se($info, "highest_grid_position", "", false);
71 if(empty($highest_grid_position)){
72     flash("Highest Grid Position must not be empty", "warning");
73 }
74 $career_points=se($info, "career_points", "-1", false);
75 if(empty($career_points) && $career_points<-1){
76     flash("Career Points must not be empty", "warning"); //ak2774, 4/15/2024
77 }

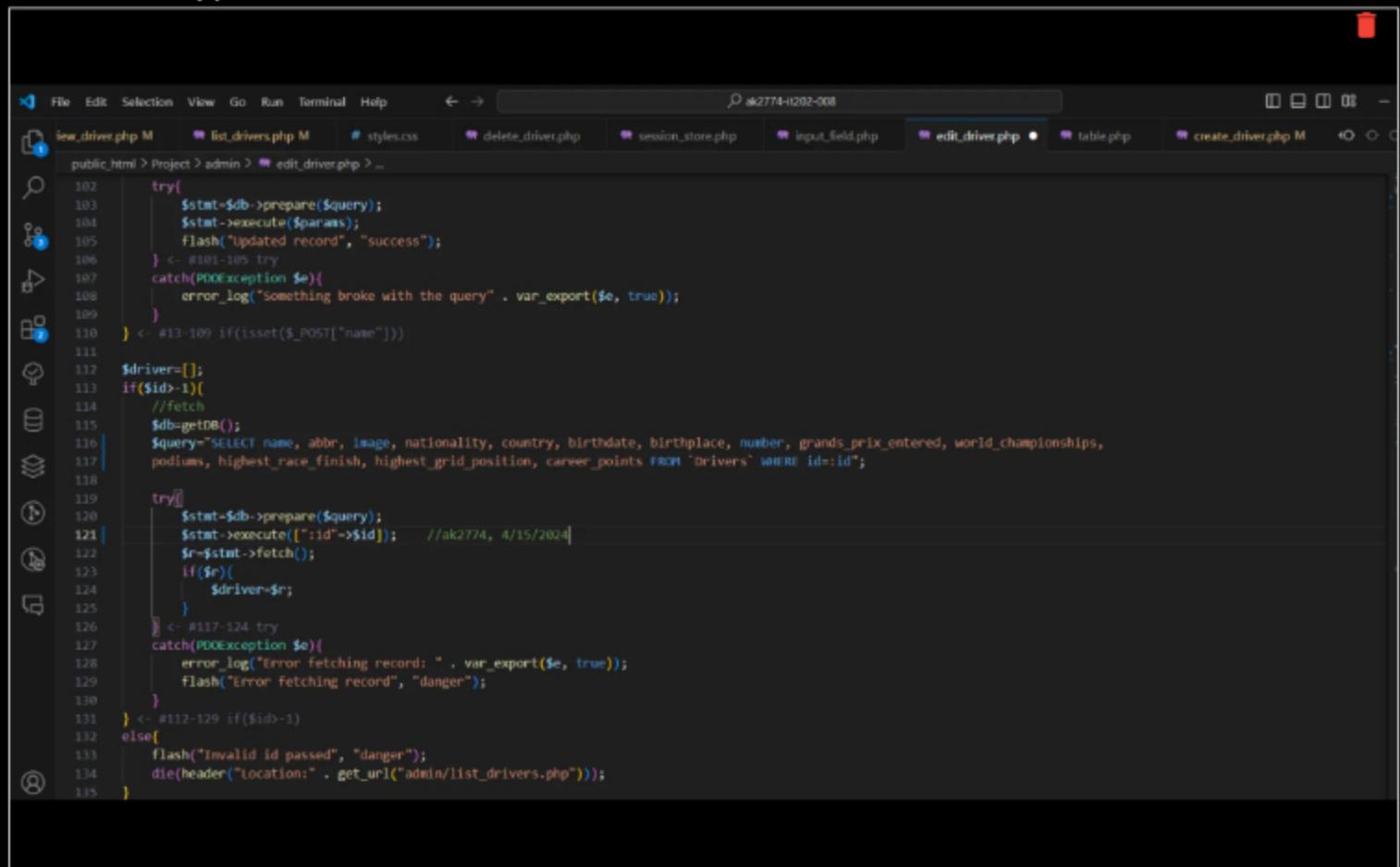
78 //Insert Data
79 $db=getDB();
80 $query="UPDATE `Drivers` SET ";
81
82 $params=[];
83 foreach($info as $k => $v) {
84
85     if($params){
86         $query .= ",";
87     }
88     //be sure $k is trusted as this is a source of sql injection
89     $query .= "$k=:{$k}";
90     $params[":{$k}"] = $v;
91 } <- #83-91 foreach($info as $k => $v)

92
93
94
95 $query .= " WHERE id=:id";
96 $params[":id"]=$id;
97 error_log("Query: " . $query);
98 error_log("Params: " . var_export($params, true));
99
100

```

Screenshot of php validation part 3 and part 1 of db insertion

Checklist Items (0)



```

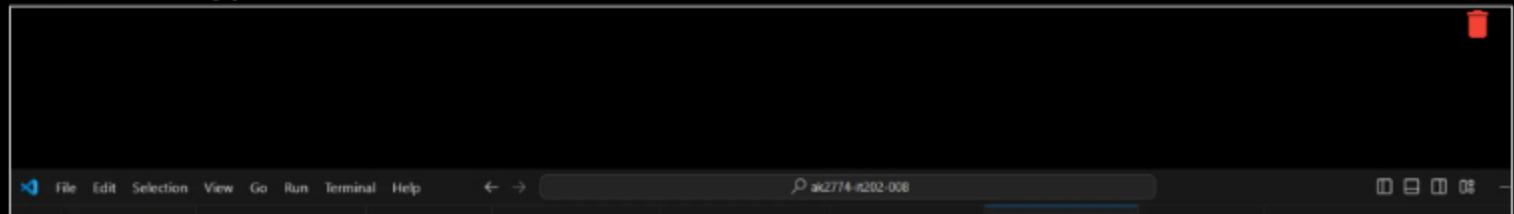
File Edit Selection View Go Run Terminal Help ← → ak2774-H202-008
new_driver.php M list_drivers.php M styles.css delete_driver.php session_store.php input_field.php edit_driver.php table.php create_driver.php M
public_html > Project > admin > edit_driver.php > ...
try{
    $stmt=$db->prepare($query);
    $stmt->execute($params);
    flash("Updated record", "success");
} <- #101-105 try
catch(PDOException $e){
    error_log("Something broke with the query" . var_export($e, true));
}
} <- #13-109 if(isset($_POST["name"]))

$driver=[];
if($id>-1){
//fetch
$db=getDB();
$query="SELECT name, abbr, image, nationality, country, birthdate, birthplace, number, grands_prix_entered, world_championships,
podiums, highest_race_finish, highest_grid_position, career_points FROM `Drivers` WHERE id=:id";
try{
    $stmt=$db->prepare($query);
    $stmt->execute([":id"=>$id]); //ak2774, 4/15/2024
    $r=$stmt->fetch();
    if($r){
        $driver=$r;
    }
} <- #117-124 try
catch(PDOException $e){
    error_log("Error fetching record: " . var_export($e, true));
    flash("Error fetching record", "danger");
}
} <- #112-129 if($id>-1)
else{
    flash("Invalid id passed", "danger");
    die(header("Location:" . get_url("admin/list_drivers.php")));
}

```

Screenshot of db insertion part 2

Checklist Items (0)



public_html > Project > admin > edit_driver.php ...

```

136 if($driver){
137     $form = [
138         ["type" => "text", "name" => "name", "placeholder" => "Driver Name", "label" => "Driver Name", "rules" => ["required" => "required"]],
139         ["type" => "text", "name" => "abbr", "placeholder" => "Drive Abbr", "label" => "Driver Abbr", "rules" => ["required" => "required"]],
140         ["type" => "text", "name" => "image", "placeholder" => "Image URL", "label" => "Image URL", "rules" => ["required" => "required"]],
141         ["type" => "text", "name" => "nationality", "placeholder" => "Nationality", "label" => "Nationality", "rules" => ["required" => "required"]],
142         ["type" => "text", "name" => "country", "placeholder" => "Country", "label" => "Country", "rules" => ["required" => "required"]],
143         ["type" => "date", "name" => "birthdate", "placeholder" => "DOB", "label" => "008", "rules" => ["required" => "required"]],
144         ["type" => "text", "name" => "birthplace", "placeholder" => "Birthplace", "label" => "Birthplace", "rules" => ["required" => "required"]],
145         ["type" => "number", "name" => "number", "placeholder" => "Driver Number", "label" => "Driver Number", "rules" => ["required" => "required"]],
146         ["type" => "number", "name" => "grands_prix_entered", "placeholder" => "GPs Entered", "label" => "GPs Entered", "rules" => ["required" => "required"]],
147         ["type" => "number", "name" => "world_championships", "placeholder" => "WCS", "label" => "World Championships", "rules" => ["required" => "required"]],
148         ["type" => "number", "name" => "podiums", "placeholder" => "Podiums", "label" => "Podiums", "rules" => ["required" => "required"]],
149         ["type" => "number", "name" => "highest_race_finish", "placeholder" => "Wins", "label" => "Wins", "rules" => ["required" => "required"]],
150         ["type" => "number", "name" => "highest_grid_position", "placeholder" => "Highest grid position", "label" => "Highest Grid Position", "rules" => ["required" => "required"]],
151         ["type" => "number", "name" => "career_points", "placeholder" => "Career Points", "label" => "Career Points", "rules" => ["required" => "required"]]
152     ];
153     foreach($form as $k=>$v){
154         if(in_array($v["name"], $keys)[]){
155             $form[$k]["value"]=$driver[$v["name"]]; //ak2774, 4/15/2024
156         }
157     } <- #151-155 foreach($form as $k=>$v)
158 } <- #154-156 if($driver)
159
160 //1000 handle manual create driver
161

```

Screenshot of form

Checklist Items (0)

public_html > Project > admin > edit_driver.php > script > validate

```

162 ?>
163 <div class="container-fluid">
164     <h3>Edit Driver</h3>
165     <div>
166         <a href="php echo get_url("admin/list_drivers.php"); ?" class="btn btn-secondary">Back</a>
167     </div>
168     <form method="POST" onsubmit="return validate(this);">
169         <?php foreach($form as $k=>$v){>
170             render_input($v);
171         } ?>
172         <?php render_button(["text" => "Search", "type" => "submit", "text"=> "Update"]); ?>
173     </form>
174 </div>
175
176 <script>
177     function validate(form){
178         let isValid=true;
179
180         let name=form.name.value;
181         if(name==""){
182             flash("Name must not be empty [js]", "warning"); //ak2774, 4/15/2024
183             isValid=false;
184         }
185         let abbr=form.abbr.value;
186         if(abbr==""){
187             flash("Abbr must not be empty [js]", "warning");
188             isValid=false;
189         }
190         let image=form.image.value;
191         if(image==""){
192             flash("Image must not be empty [js]", "warning");
193             isValid=false;
194         }
195     }
196 </script>

```

Screenshot css and part 1 of js validation

Checklist Items (0)

public_html > Project > admin > edit_driver.php > validate

```

162 ?>
163 <div class="container-fluid">
164     <h3>Edit Driver</h3>
165     <div>
166         <a href="php echo get_url("admin/list_drivers.php"); ?" class="btn btn-secondary">Back</a>
167     </div>
168     <form method="POST" onsubmit="return validate(this);">
169         <?php foreach($form as $k=>$v){>
170             render_input($v);
171         } ?>
172         <?php render_button(["text" => "Search", "type" => "submit", "text"=> "Update"]); ?>
173     </form>
174 </div>
175
176 <script>
177     function validate(form){
178         let isValid=true;
179
180         let name=form.name.value;
181         if(name==""){
182             flash("Name must not be empty [js]", "warning"); //ak2774, 4/15/2024
183             isValid=false;
184         }
185         let abbr=form.abbr.value;
186         if(abbr==""){
187             flash("Abbr must not be empty [js]", "warning");
188             isValid=false;
189         }
190         let image=form.image.value;
191         if(image==""){
192             flash("Image must not be empty [js]", "warning");
193             isValid=false;
194         }
195     }
196 </script>

```

```

public_html > Project > admin > edit_driver.php > script > validate
176     <script>
177         function validate(form){
195             let nationality=form.nationality.value;
196             if(nationality==""){
197                 flash("Nationality must not be empty [js]", "warning");
198                 isValid=false;
199             }
200             let country=form.country.value;
201             if(country==""){
202                 flash("Country must not be empty [js]", "warning");
203                 isValid=false;
204             }
205             let birthdate=form.birthdate.value;
206             if(birthdate==""){
207                 flash("Birthdate must not be empty [js]", "warning");
208                 isValid=false;
209             }
210             let birthplace=form.birthplace.value;
211             if(birthplace==""){
212                 flash("Birthplace must not be empty [js]", "warning"); //ak2774, 4/15/2024
213                 isValid=false;
214             }
215             let number=form.number.value;
216             if(number==""){
217                 flash("Number must not be empty [js]", "warning");
218                 isValid=false;
219             }
220             let grands_prix_entered=form.grands_prix_entered.value;
221             if(grands_prix_entered==""){
222                 flash("GPs Entered must not be empty [js]", "warning");
223                 isValid=false;
224             }
}

```

Screenshot of js validation part 2

Checklist Items (0)

```

File Edit Selection View Go Run Terminal Help ← → ak2774-it202-008
new_driver.php M list_drivers.php M styles.css delete_driver.php session_store.php input_field.php edit_driver.php ● table.php create_driver.php
public_html > Project > admin > edit_driver.php > script > validate
176     <script>
177         function validate(form){
229             }
230             let podiums=form.podiums.value;
231             if(podiums==""){
232                 flash("Podiums must not be empty [js]", "warning");
233                 isValid=false;
234             }
235             let highest_race_finish=form.highest_race_finish.value;
236             if(highest_race_finish==""){
237                 flash("Highest race finish must not be empty [js]", "warning");
238                 isValid=false;
239             }
240             let highest_grid_position=form.highest_grid_position.value;
241             if(highest_grid_position==""){
242                 flash("Highest Grid Position must not be empty [js]", "warning"); //ak2774, 4/15/2024
243                 isValid=false;
244             }
245             let career_points=form.career_points.value;
246             if(career_points==""){
247                 flash("Career Points must not be empty [js]", "warning");
248                 isValid=false;
249             }
250             return isValid;
251         } <- #175-250 function validate(form)
252
253
254     </script>
255
256     <?php
257     require_once(__DIR__ . "/../../../../partials/flash.php");
258     ?>

```

Screenshot of js validation part 3

Checklist Items (0)

Task #3 - Points: 1

Text: Screenshot of records from DB

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Show a before and after screenshot of the record
<input type="checkbox"/> #2	1	Note what differs
<input type="checkbox"/> #3	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small

Medium

Large

	id	name	abbr	image	nationality	country	birthdate	birthplace
> 1	2	Kevin Magnussen	MAG	https://media.api-sports.io/	Danish	Denmark	1992-10-05	Roskilde, Denmark
> 2	4	Fernando Alonso	ALO	https://media.api-sports.io/	Spanish	Spain	1981-07-29	Oviedo, Spain
> 3	5	Valtteri Bottas	BOT	https://media.api-sports.io/	Finish	Finland	1989-08-28	Nestola, Finland
> 4	6	Nico Hulkenberg	HUL	https://media.api-sports.io/	German	Germany	1987-08-19	Emmerich am Rhein, Germs
> 5	10	Sergio Perez	PER	https://media.api-sports.io/	Mexican	Mexico	1990-01-26	Guadalajara, Mexico
> 6	14	Daniel Ricciardo	RIC	https://media.api-sports.io/	Australian	Australia	1989-07-01	Perth, Australia
> 7	19	Sebastian Vettel	VET	https://media.api-sports.io/	German	Germany	1987-07-03	Heppenheim, Germany
> 8	20	Lewis Hamilton	HAM	https://media.api-sports.io/	British	United Kingdom	1985-01-07	Stevenage, England
> 9	24	Carlos Sainz Jr	SAI	https://media.api-sports.io/	Spanish	Spain	1994-09-01	Madrid, Spain

Screenshot of the db after update

Checklist Items (0)

The screenshot shows a database table with the following columns: id (int), name (varchar(40)), abbr (varchar(5)), image (text), nationality (varchar(20)), country (varchar(30)), birthdate (date), and birthplace (varchar(100)). The data consists of six rows:

	id	name	abbr	image	nationality	country	birthdate	birthplace
> 1	2	Kevin Magnussen	MAG	https://media.api-sports.io/	Danish	Denmark	1992-10-05	Roskilde, Denmark
> 2	4	Fernando Alonso	ALO	https://media.api-sports.io/	Spanish	Spain	1981-07-29	Oviedo, Spain
> 3	5	Valterri Bottas	BOT	https://media.api-sports.io/	Finnish	Finland	1989-08-28	Nastola, Finland
> 4	6	Nico Hulkenberg	HUL	https://media.api-sports.io/	German	Germany	1987-08-19	Emmerich am Rhein, Germany
> 5	10	Sergio Perez	PER	https://media.api-sports.io/	Mexican	Mexico	1990-01-26	Guadalajara, Mexico
> 6	14	Daniel Ricciardo	RIC	https://media.api-sports.io/	Australian	Australia	1989-07-01	Perth, Australia

Screenshot of db before update

Checklist Items (0)

Task #4 - Points: 1

Text: Explain the process

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Provide a high-level step-by-step of how the fetching of the record, populating the form, and the update works and gets changed in your DB
<input type="checkbox"/> #2	1	Briefly describe the validations for the applicable fields

Response:

The script begins by confirming the user's admin privileges; if not authenticated, it redirects them and displays a warning message. Once authenticated, it extracts the driver's ID from the URL parameters and attempts to fetch the corresponding record from the database. If the ID is valid and a record is found, the form fields are populated with the retrieved data, linking each field with its respective database column. Upon form submission, the script validates the input data, ensuring critical fields such as name, abbreviation, image URL, nationality, country, birthdate, birthplace, number, GPs entered, world championships, podiums, highest race finish, highest grid position, and career points are not empty. If any field fails validation, a warning message alerts the user. If all inputs pass validation, an SQL UPDATE query is constructed dynamically, incorporating the submitted form data to update the driver record in the database. Prepared statements are employed to execute the query, mitigating SQL injection risks. Validation rules encompass checking for required fields, validating numeric data types for numerical inputs, and potentially enforcing additional constraints like image URL format correctness or specific numeric ranges. These validation mechanisms ensure data integrity and consistency in the database, safeguarding against erroneous or malicious inputs.

Task #5 - Points: 1

Text: Add related links

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Include the heroku prod link for this page
<input type="checkbox"/> #2	1	Add the pull request link for the branch related to this feature Note: the link should end with /pull/#. Same pull request shouldn't be used for each feature

URL #1

https://ak2774-prod-76402b453c05.herokuapp.com/Project/admin/edit_driver.php?id=2

URL #2

<https://github.com/abhinavkarthikn/ak2774-it202-008/pull/64>

● Delete Handling (1 pt.)

[^COLLAPSE ^](#)

Task #1 - Points: 1

Text: Screenshots related to delete

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Show the success message of a delete
<input type="checkbox"/> #2	1	Show any error messages of a failed delete (like id not being passed)
<input type="checkbox"/> #3	1	Show the code related to the delete processing

Task Screenshots:

Gallery Style: Large View

Small

Medium

Large

The screenshot shows a web application interface for managing drivers. At the top, there's a navigation bar with links for Home, Profile, Admin, Driver Management, and Logout. Below the navigation, a success message 'Deleted record with id 2' is shown in a green box. The main content area is titled 'List Drivers' and contains a table with columns for Name, Nationality, Country, Driver #, Min GPs, and Max GPs. There are input fields for filtering these columns. At the bottom, there are 'Filter' and 'Clear' buttons.

Screenshot of delete message

Checklist Items (0)

```
# styles.css      delete_driver.php ●  session_store.php  input_field.php  edit_driver.php M  table.php  create_driver.php 🗑  
public_html > Project > admin > delete_driver.php > ...  
You, 2 days ago | 1 author (You)  
1  <?php  
2  
3  session_start();  
4  require_once(__DIR__ . "/../../../../lib/functions.php");  
5  if(!has_role("Admin")){  
6      flash("You do not have permission to view this page", "warning");  
7      die(header("location: $BASE_PATH" . "/home.php"));  
8  }  
9  
10 $id=se($_GET, "id", -1, false);  
11 if($id<1){  
12     flash("Invalid id passed to delete", "danger");  
13     die(header("Location: " . get_url("admin/list_drivers.php")));  
14 }  
15  
16 $db=getDB();  
17 $query="DELETE FROM `Drivers` WHERE id=:id"; //ak2774, 4/15/2024 You, 2 days ago * Uncommitted changes  
18 try{  
19     $stmt=$db->prepare($query);  
20     $stmt->execute([":id"=>$id]);  
21     flash("Deleted record with id $id", "success");  
22 } <- #18-22 try  
23 catch(PDOException $e){  
24     error_log("Error deleting driver $id" . var_export($e, true));  
25     flash("Error deleting record", "danger");  
26 }  
27 } <- #23-27 catch(PDOException $e)  
28 die(header("Location: " . get_url("admin/list_drivers.php")));
```

Screenshot of delete code

Checklist Items (0)

The screenshot shows a web browser window with the URL ak2774-dev-f660064fd52.cherokapp.com/Project/admin/list_drivers.php. The page title is "List Drivers". At the top, there is a navigation bar with links for "Home", "Profile", "Admin", "Driver Management", and "Logout". Below the navigation bar, a red alert message box displays the text "Invalid id passed to delete". The main content area is titled "List Drivers" and contains a table with columns: Name, Nationality, Country, Driver #, Min GPs, and Max GPs. There are input fields for filtering these columns. At the bottom left, there are "Filter" and "Clear" buttons.

Screenshot of error message showing failed delete

Checklist Items (0)

Task #2 - Points: 1

Text: Screenshots of the data

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Show a before and after screenshot of the DB data
<input type="checkbox"/> #2	1	Note what changed (i.e., record removed or soft delete value changed)
<input type="checkbox"/> #3	1	Clearly caption screenshots

Task Screenshots:

Gallery Style: Large View

Small

Medium

Large

The screenshot shows a MySQL database interface with the following details:

- Database:** abh5server 0.35-Ubuntu0.20.04.1
- Table:** Drivers
- Query:** SELECT * FROM 'Drivers' LIMIT 100
- Results:** Total 26 rows
- Columns:** id, name, abbr, image, nationality, country, birthdate, birthplace
- Data:** The table lists 26 drivers, including Kevin Magnussen, Fernando Alonso, Valtteri Bottas, Nico Hulkenberg, Sergio Perez, Daniel Ricciardo, etc.

Screenshot of db before deletion

Checklist Items (0)

The screenshot shows a MySQL database interface with the following details:

- Database:** AlibiServer 8.0.35 (Ubuntu 20.04.1)
 - al2774 144k
 - Tables (4): Drivers (25), Roles (2), UserRoles (7), Users (5)
 - Views (0)
 - Procedures (0)
 - Functions (0)
 - Information_schema (0)
- Table:** Drivers (25) - SELECT * FROM "Drivers" LIMIT 100

	ID	Name	Abr	Image	Nationality	Country	Birthdate	Birthplace
> 1	4	Fernando Alonso	ALO	https://media.api-sports.io/	Spanish	Spain	1981-07-29	Oviedo, Spain
> 2	5	Valterri Bottas	BOT	https://media.api-sports.io/	Finnish	Finland	1989-08-28	Nastola, Finland
> 3	6	Nico Hulkenberg	HUL	https://media.api-sports.io/	German	Germany	1987-08-19	Emmerich am Rhein, Germany
> 4	10	Sergio Perez	PER	https://media.api-sports.io/	Mexican	Mexico	1990-01-26	Guadalajara, Mexico
> 5	14	Daniel Ricciardo	RIC	https://media.api-sports.io/	Australian	Australia	1989-07-01	Perth, Australia
> 6	19	Sebastian Vettel	VET	https://media.api-sports.io/	German	Germany	1987-07-03	Heppenheim, Germany

Screenshot of db after deletion

Checklist Items (0)

Task #3 - Points: 1

Text: Explain the delete logic

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Is it a soft or hard delete
<input type="checkbox"/> #2	1	Are there any necessary roles or restrictions? (can only delete their data, can only be done by admin, etc)
<input type="checkbox"/> #3	1	Provide the high-level steps for handling the DB lookup and handling the delete

Response:

This PHP script facilitates a hard delete operation, removing a record from the "Drivers" table within the database. It begins by verifying the user's administrative privileges. If the user lacks admin rights, a warning message is displayed, and they are redirected to the home page. Following this, the script extracts the driver's ID from the URL parameters, ensuring it's a valid identifier. If an invalid or nonexistent ID is detected, an error message is displayed, and the user is redirected to the list of drivers. Subsequently, the script establishes a connection to the database and constructs a SQL DELETE query aimed at removing the driver record associated with the specified ID. Using a prepared statement, the query is executed, with the ID passed as a parameter. Upon successful execution, a success message is flashed, confirming the deletion of the record. However, if an error occurs during the deletion process, it is logged for debugging purposes, an error message is flashed, and the redirection process is handled accordingly. In terms of roles or restrictions, the script only allows users with administrative privileges to execute the delete operation.

Task #4 - Points: 1

Text: Add the pull request link for the branch related to this feature

Details:

Note: the link should end with /pull/#. Same pull request shouldn't be used for each feature

URL #1

<https://github.com/abhinavkarthikn/ak2774-it202-008/pull/65>

Misc (1 pt.)

COLLAPSE

Task #1 - Points: 1

Text: Screenshot of your project board from GitHub (tasks should be in the proper column)

Task Screenshots:

Gallery Style: Large View

Small

Medium

Large

The screenshot shows a GitHub project board with three columns: Todo, In Progress, and Done. The Todo column has one item: "This item hasn't been started". The In Progress column has one item: "This is actively being worked on". The Done column contains five items, each with a checkmark and a red circular icon with a white 'X':

- ak2774-it202-008 #55 MS2- Define the appropriate table or tables for your API
- ak2774-it202-008 #56 MS2-Data creation page
- ak2774-it202-008 #57 MS2- Data list page
- ak2774-it202-008 #58 *** MS2- View data details page
- ak2774-it202-008 #59 MS2- Edit data page

Screenshot of project board part 1

The screenshot shows a GitHub project board with three columns: Todo, In Progress, and Done.

- Todo:** 0 items. Description: This item hasn't been started.
- In Progress:** 0 items. Description: This is actively being worked on.
- Done:** 16 items. Description: This has been completed.
 - ak2774-it202-008 #60 MS2- Delete handling
 - ak2774-it202-008 #61 MS2- API handling
 - ak2774-it202-008 #39 MS1- User will be able to logout
 - ak2774-it202-008 #41 MS1 - User will be able to register a new account
 - ak2774-it202-008 #42 MS1- User will be able to edit their profile

Screenshot of project board part 2

Task #2 - Points: 1

Text: Provide a direct link to the project board on GitHub

URL #1

<https://github.com/users/abhinavkarthikn/projects/1>

Task #3 - Points: 1

Text: Talk about any issues or learnings during this assignment

Response:

I had a lot of issues with the API data and it being connected to the db so my db wouldn't have any records unless I used create. I also had issues with formatting the data from the database because I didn't use all the data from the database so I had to transform my data to only take what I needed and removed the rest of the data. I also had issues with the validation part because my validation would not work for a while until it was formatted properly and the if statements were in the correct order

[COLLAPSE](#)

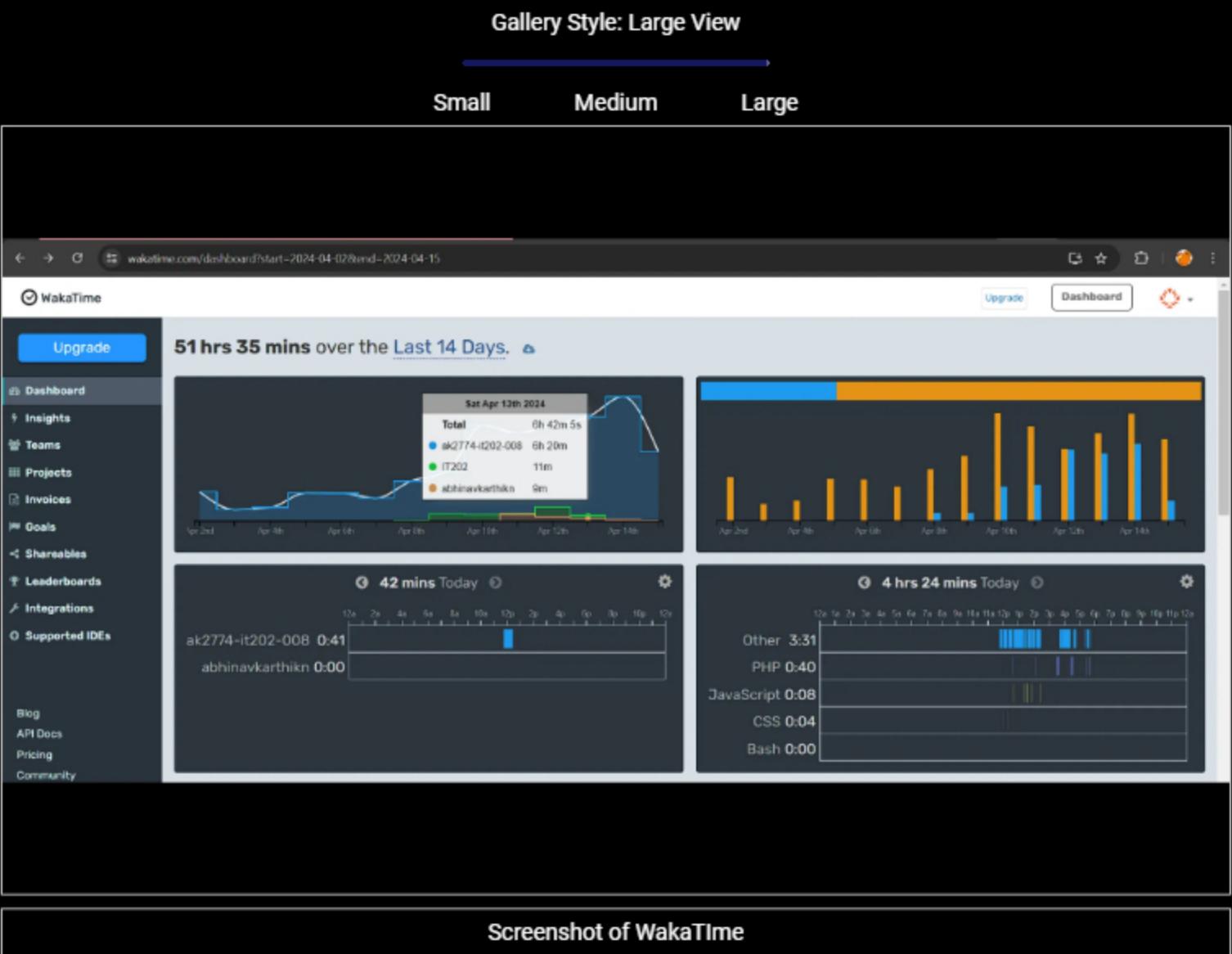
Task #4 - Points: 1

Text: WakaTime Screenshot

1 Details:

Grab a snippet showing the approximate time involved that clearly shows your repository. The duration isn't considered for grading, but there should be some time involved

Task Screenshots:



End of Assignment