

PROJET CSI2532 LEADERBOARD

Sommaire



Modèle de calcul du leaderboard



Mise à jour de données




Gestion des recherches et filtres



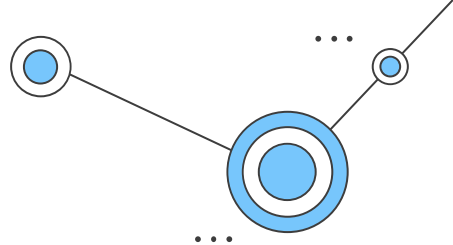


01

Modèle de calcul du leaderboard

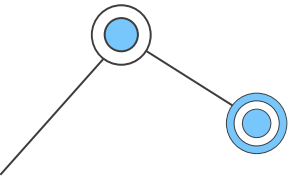


Leaderboard



Un leaderboard est intrinsèquement lié à une compétition et dépend de tous les scores obtenus par les athlètes à chaque évènement. Ainsi afin d'obtenir le leaderboard d'une compétition l'on doit pouvoir l'identifier correctement.

C'est dans ce contexte que nous utilisons l'identifiant de la compétition dans l'écriture de nos requêtes SQL.





MeFit

COMPETITIONS

Un click donne accès au
leaderboard
correspondant en
POSTant l'attribut
competitions.identifier

name	venue	date	time	registrations (per gender)	total events	contacts	organised by
Joyfest	Arena Joshau	2021-02-10 2021-02-14	13:30:00	10	2	galijon086@plateau.ca 8133550456	Plateau Inc.
Montreal Tour	Trudeau Stadium	2021-05-15 2021-05-20	14:00:00	15	3	wegibb@gmail.ca 6134550469	Agartha SARL
Ottawa Fest	uOttawa	2021-04-03 2021-04-06	09:00:00	20	5	laurcastillo@gmail.ca 6144550469	Agartha SARL
Toronto Fest	Martin Square	2021-03-01 2021-03-03	16:00:00	20	2	annakeat@gmail.com 6144557469	Limuse Inc.
California Fit	Calif Square	2021-04-15 2021-04-20	16:00:00	20	5	frankdelf@gmail.com 8144567469	Apple Inc.
Solar Fest	Solary Beach	2021-03-28 2021-03-29	11:00:00	10	2	jamesfr@gmail.com 8733550472	Plateau Inc.



Recherche de tous les évènements liés à la compétition



Les calculs suivants liés au leaderboard sont disponibles dans le fichier leaderboard.php.

Tout d'abord afin d'obtenir le leaderboard d'une compétition l'on doit rechercher les informations de tous les évènements associés.

```
$competitionId = $_GET["id"];  
$sql = 'SELECT * FROM competitions WHERE identifier = $1';  
$result = pg_query_params($dbconn, $sql, [$competitionId]);  
$data = pg_fetch_all($result);  
  
$sql = "SELECT DISTINCT events.name as event, primary_score, primary_score_tb, time_cap, time_cap_tb  
FROM events JOIN competitions ON competition_id = competitions.id WHERE competitions.identifier = $1";  
$events = pg_query_params($dbconn, $sql, [$competitionId]);  
$eventsData = pg_fetch_all($events);  
$eventsNames = pg_fetch_all_columns($events, pg_field_num($events, 'event'));  
$eventsPrimary = pg_fetch_all_columns($events, pg_field_num($events, 'primary_score'));  
$eventsPrimaryTb = pg_fetch_all_columns($events, pg_field_num($events, 'primary_score_tb'));  
$eventsSecondary = pg_fetch_all_columns($events, pg_field_num($events, 'time_cap'));  
$eventsSecondaryTb = pg_fetch_all_columns($events, pg_field_num($events, 'time_cap_tb'));
```

... Classement des athlètes par rapport à un évènement

Une fois les évènements obtenus nous allons rechercher l'ordre sur lequel il faut se baser afin d'organiser les scores de chaque évènement.

Ce cas ne devrait jamais arrivé on devrait toujours avoir une façon de classier des athlètes. On verra plus tard comment on s'assure que cette contrainte est respectée.

```
for ($event = 0; $event < count($eventsData); $event++){  
    switch($eventsPrimary[$event]){  
        case 'Time ASC':  
            $eventPrimary = 'finish_time ASC';  
            break;  
        case 'Time DESC':  
            $eventPrimary = 'finish_time DESC';  
            break;  
        case 'Count ASC':  
            $eventPrimary = 'number_reps ASC';  
            break;  
        case 'Count DESC':  
            $eventPrimary = 'number_reps DESC';  
            break;  
        default:  
            $eventPrimary = "";  
    }  
}
```

...Classement des athletes par rapport à un évènement

```
65     switch($eventsPrimaryTb[$event]){
66     case 'Time ASC':
67         $eventPrimaryTb = ',finish_time ASC';
68         break;
69
70     case 'Time DESC':
71         $eventPrimaryTb = ',finish_time DESC';
72         break;
73
74     case 'Count ASC':
75         $eventPrimaryTb = ',number_reps ASC';
76         break;
77
78     case 'Count DESC':
79         $eventPrimaryTb = ',number_reps DESC';
80         break;
81
82     default:
83         $eventPrimaryTb = "";
84         break;
85 }
```

```
87     switch($eventsSecondary[$event]){
88     case 'Time ASC':
89         $eventSecondary = ',finish_time ASC';
90         break;
91
92     case 'Time DESC':
93         $eventSecondary = ',finish_time DESC';
94         break;
95
96     case 'Count ASC':
97         $eventSecondary = ',number_reps ASC';
98         break;
99
100    case 'Count DESC':
101        $eventSecondary = ',number_reps DESC';
102        break;
103
104    default:
105        $eventSecondary = "";
106        break;
107 }
```

```
107 }
108
109     switch($eventsSecondaryTb[$event]){
110     case 'Time ASC':
111         $eventSecondaryTb = ',best_set_time ASC';
112         break;
113
114     case 'Time DESC':
115         $eventSecondaryTb = ',best_set_time DESC';
116         break;
117
118     case 'Count ASC':
119         $eventSecondaryTb = ',number_reps ASC';
120         break;
121
122     case 'Count DESC':
123         $eventSecondaryTb = ',number_reps DESC';
124         break;
125
126     default:
127         $eventSecondaryTb = "";
128         break;
129 }
```




Classement des athlètes par rapport à un évènement



Ainsi après les 4 switch() l'ordre de classification sera :

```
ORDER BY '.$eventPrimary.$eventPrimaryTb.$eventSecondary.$eventSecondaryTb.') as rank '
```

Nous nous assurons ainsi de respecter l'ordre déterminé lors de la création de l'évènement.

Nous allons écrire maintenant notre requête SQL afin de classer les scores en fonction de l'ordre préétabli:

```
$sql = 'SELECT athlete.identifier as identifier ,athlete.name as athlete,events.name as event, number_reps,finish_time,best_set_time,EXTRACT (year from age(date_of_birth)) as age, gender,
RANK()OVER( ORDER BY '.$eventPrimary.$eventPrimaryTb.$eventSecondary.$eventSecondaryTb.') as rank '
"FROM scores
JOIN athlete ON athlete.id = scores.athlete_id
JOIN events ON events.id = scores.event_id
JOIN register ON register.athlete_id = scores.athlete_id
JOIN competitions ON competitions.id = register.competition_id
WHERE competitions.id = $1 AND events.name = $2";

$result = pg_query_params($dbconn, $sql, [$competition['id'],$eventsNames[$event]]);
array_push($scoresArray, $result);
```


Après chaque tour i de boucle nous allons ajouter dans le tableau \$scoresArray les classifications de chaque athlète par rapport à l'évènement i .



Classement par rapport à l'ensemble de la compétition



Nous allons à présent créer un tableau \$leaderboard qui va regrouper le classement final et quelques informations liées à l'athlète, Pour ce faire l'on va se servir du tableau \$scoresArray. Nous ferons une boucle sur ce tableau. À chaque tour de boucle nous vérifions si l'athlète est dans le \$leaderboard si oui on va mettre à jour ses points en ajoutant son rang obtenu à l'évènement i-1 à son rang obtenu à l'évènement i. Sinon on l'ajoute directement dans le \$leaderboard.



```
$leaderboard = array();
foreach ($scoresArray as $eventScores){
    while($row = pg_fetch_assoc($eventScores)){
        if (count($leaderboard) == 0){
            array_push($leaderboard, array($row['identifiant'],$row['athlete'],$row['rank'],$row['age'],$row['gender']));
        }else{
            $num = count($leaderboard);
            for($i = 0 ; $i < $num ; $i++){
                if($row['identifiant'] == $leaderboard[$i][0]){
                    $leaderboard[$i][2]+=$row['rank'];
                    break;
                }elseif($i == count($leaderboard)-1){
                    array_push($leaderboard, array($row['identifiant'],$row['athlete'],$row['rank'],$row['age'],$row['gender']));
                }
            }
        }
    }
}
```



Classement par rapport à l'ensemble de la compétition



La boucle précédente n'est pas suffisante il faut trier à présent le \$leaderboard en fonction de 'rank' qui représente maintenant les points de l'athlète. L'athlète ayant le plus petit nombre de points est le vainqueur.

\$leaderboard est un tableau(matrice) 2D où la ligne fait référence à l'athlète et la colonne :

0 -> identifier, 1 -> nom de l'athlète, 2 -> points, 3 -> age, 4 -> genre

```
for($i=0 ; $i< count($leaderboard) ;$i++){  
    $index = $i;  
  
    for($j = $i+1 ; $j< count($leaderboard) ;$j++){  
        if($leaderboard[$j][2] < $leaderboard[$index][2]){  
            $index = $j;  
        }  
    }  
    $temp = $leaderboard[$i];  
    $leaderboard[$i] = $leaderboard[$index];  
    $leaderboard[$index] = $temp;  
}
```



Classement par rapport à l'ensemble de la compétition



Afin d'afficher le rang final on utilise du php couplé avec du HTML. Les variables utiles ici sont \$sameRankCounter et \$finalRank .

```
if($i == 0 || $leaderboard[$i][2] == $leaderboard[$i - 1][2]){
    $sameRankCounter++;
}>
<td><?php echo $finalrank.' ( '.$leaderboard[$i][2].' points'.' )';?></td>
<?php
}else{?>
    <td>
    <?php
    if($sameRankCounter > 1){
        $finalrank += $sameRankCounter;
        $sameRankCounter = 1;
    }else{
        $finalrank++;
    }

    echo $finalrank.' ( '.$leaderboard[$i][2].' points'.' )';

    ?>
```



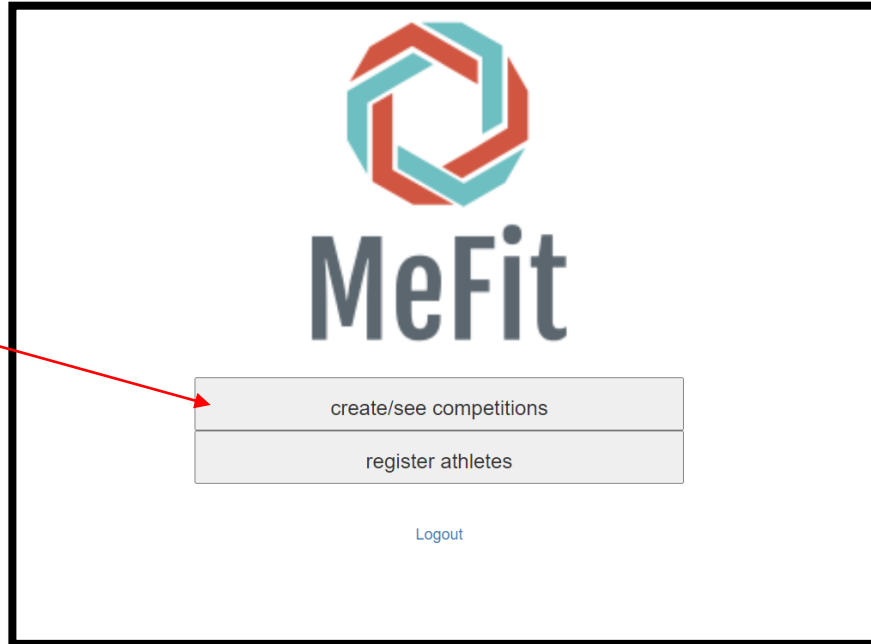
02

Mise à jour des données

Mise à jour des données

Une fois que leur jetons sont vérifiés (voir README.md et api.php) les partenaires ont accès à leur portail privé via une page web.

Voyons comment
nous créons de
nouvelles compétitions



Creation de competitions et évènements



CREATE COMPETITION

Name
Venue
Competition Year
Representative
representative email
Phone Number
Start Date
End Date
Start Time
Number Of Events
Maximum of athletes

COMPETITIONS CREATED

name	venue	date	time	registrations (per gender)	total events	contacts	
Joyfest	Arena Joshau	2021-02-10 2021-02-14	13:30:00	10	2	gallion086@plateau.ca 8133550456	<input type="button" value="Delete"/>
Solar Fest	Solary Beach	2021-03-28 2021-03-29	11:00:00	10	2	jamesfr@gmail.com 8733550472	<input type="button" value="Delete"/>



CREATE EVENTS


Event #1

Name
Date
Primary Score
Primary Tie-Break Score
TimeCap Score
TimeCap Tie-Break Score

Event #2

Name
Date
Primary Score
Primary Tie-Break Score
TimeCap Score
TimeCap Tie-Break Score

Creation de compétitions et d'évènements



CREATE EVENTS

Event #1

Name

Date

Primary Score Pas de valeur null

Primary Tie-Break Score

TimeCap Score

TimeCap Tie-Break Score

Event #2

Name

Date

Primary Score

Primary Tie-Break Score

TimeCap Score

TimeCap Tie-Break Score



CREATE COMPETITION

Competition created successfully

Name

Venue

Competition Year

Representative

representative email

Phone Number

Start Date

End Date

Start Time

Number Of Events

Maximum of athletes

Mis à jour

COMPETITIONS CREATED

name	venue	date	time	registrations (per gender)	total events	contacts	
Joyfest	Arena Joshua	2021-02-10 2021-02-14	13:30:00	10	2	galljon086@plateau.ca 8133550456	<input type="button" value="Delete"/>
Solar Fest	Solitary Beach	2021-03-28 2021-03-29	11:00:00	10	2	jamesfr@gmail.com 8733550472	<input type="button" value="Delete"/>
uOttawa Fit	uOttawa Gym	2021-04-11 2021-04-12	10:00:00		2	ethanjohnd@uottawa.ca 6135687966	<input type="button" value="Delete"/>



Creation de compétitions et d'évènements



D'un point de vue technique les informations sont POSTées via le formulaire de organize_comp.php et create_events.php puis des requêtes SQL sont utilisées afin de pouvoir insérer dans la base de données la nouvelle compétition.

Dans check_competitions.php

```
$sql = 'INSERT INTO competitions(id,partner_id, name,venue,compet_year ,start_date , end_date,
start_time,max_athlete,number_events,representative,email,phone_number)
VALUES($1,$2,$3,$4,$5,$6,$7,$8,$9,$10,$11,$12,$13)';

$result = pg_query_params($dbconn, $sql, [$compet_id,$partner_id,$name,$venue,$compet_year,$start_date,
$end_date,$start_time,$max_athlete,$number_events,$representative,
$email,$phone_number]);

if ($result == true){
    header("Location: create_events.php?id=".$compet_id."&$competData");
}else{
    header("Location: organize_comp.php?error=insertion failed &$competData");
}
```

Dans check_events.php

```
$sql = 'INSERT INTO events(competition_id, name,primary_score, primary_score_tb, time_cap, time_cap_tb,event_date)
VALUES($1,$2,$3,$4,$5,$6,$7)';
$result = pg_query_params($dbconn, $sql, [$compet_id,$name,$primary_score,$primary_score_tb,$time_cap,$time_cap_tb,$event_date]);

if ($result == true){
    header("Location: organize_comp.php?success=Competition created successfully&identifier=".$partnerId);
}else{
    header("Location: create_events.php?error=insertion failed&$eventData");
}
```

Mise à jour des données


Une fois que leur jetons sont vérifiés (voir README.md et api.php) les partenaires ont accès à leur portail privé via une page web.

Voyons comment
nous créons des
athlètes et mettons à
jour les scores



Création des athlètes

Tout comme la création d'une compétition l'enregistrement d'un athlète consiste à remplir le formulaire (register.php) et POSTer les informations à check_athlete.php.



REGISTER ATHLETES

Name

Date Of Birth

Sex

Identified gender

Email

Nationality

Register In

ATHLETES REGISTERED

athlete	nationality	email	registered in
Loc Ky M/20	Burkina Faso	kyloic@gmail.com	Joyfest
Wilfried Sankara M/30	Burkina Faso	wilfsank@gmail.com	Joyfest
Kamagat, Abdoul-Aziz M/19	Canada	kam056@gmail.com	Joyfest
Shein Frank M/25	Canada	shfrank05@gmail.com	Solar Fest
Henock Francis M/20	Congo	henfrancis@gmail.com	Joyfest
Mohamed Konate M/18	France	mkona05@gmail.com	Joyfest Solar Fest
Yoanna Jones	France	joyoanna@gmail.com	Joyfest

Création des athlètes

Le fichier check_athlete.php va tout d'abord vérifier si l'athlète enregistré existe dans la base de données,

```
$sql = "SELECT * FROM athlete WHERE name = $1 AND date_of_birth = $2 AND sex = $3 AND nationality = $4";
$result = pg_query_params($dbconn, $sql, [$athlete_name, $dob, $sex, $nationality]);
$num_row = pg_num_rows($result);
$data = pg_fetch_all($result);

if ($num_row > 0) {
    // there is an athlete with the same informations
    $athlete_id = $data[0]['id'];
    $result = true;
}
```

Sinon

On crée un nouvel athlete

```
}else{
    $sql = "SELECT nextval('athletes_id_seq') as id";
    $result = pg_query($dbconn, $sql);
    $athlete_id = pg_fetch_all($result)[0]['id'];
    $sql = 'INSERT INTO athlete(id, name, date_of_birth, nationality, sex, email)
VALUES($1, $2, $3, $4, $5, $6)';
    $result = pg_query_params($dbconn, $sql, [$athlete_id, $athlete_name, $dob, $nationality, $sex, $email]);
}
```

Création des athlètes

À présent nous allons essayer d'inscrire l'athlète à la compétition choisie si le nombre maximum pour la compétition est atteint on affiche un message d'erreur. Dans le cas où le nombre maximum n'est pas atteint ou est null on inscrit l'athlète à la compétition.

```
if ($result == true){  
  
    $sql = "SELECT max_athlete FROM competitions WHERE competitions.id = $1";  
    $result = pg_query_params($dbconn, $sql, [$competition_id]);  
    $max_athlete = pg_fetch_all($result)[0]['max_athlete'];  
  
    $sql = 'SELECT count(gender)as num_athlete FROM register JOIN competitions ON competitions.id = register.competition_id WHERE gender=$1 AND  
            competition_id = $2 ';  
    $result = pg_query_params($dbconn, $sql, [$gender,$competition_id]);  
    $num_athlete = pg_fetch_all($result)[0]['num_athlete'];  
  
    if($max_athlete > $num_athlete || empty($max_athlete)){  
        $sql = "INSERT INTO register(athlete_id,competition_id,gender)  
                VALUES($1,$2,$3)";  
        $result = pg_query_params($dbconn, $sql, [$athlete_id,$competition_id,$gender]);  
        if($result == true){  
            header("Location: register.php?success=Registered successfully &identifier=$partner_identfier");  
        }else{  
            header("Location: register.php?error=insertion failed athlete already registered &$athleteData");  
        }  
    }else{  
        header("Location: register.php?error=registration failed, maximum athlete reached for this competition &$athleteData");  
    }  
}else{  
    header("Location: register.php?error=insertion failed &$athleteData");  
}
```

... Mettre à jour les scores des athlètes



REGISTER ATHLETES

Name
Date Of Birth
Sex
Identified gender
Email
Nationality
Register in

ATHLETES REGISTERED

athlete	nationality	email	registered in
Locc Ky M/20	Burkina Faso	kyloic@gmail.com	Joyfest
Wilfried Sankara M/30	Burkina Faso	wilfsank@gmail.com	Joyfest
Kamagat, Abdoul-Aziz M/19	Canada	kam056@gmail.com	Joyfest
Shein Frank M/25	Canada	shfrank05@gmail.com	Solar Fest
Henock Francis M/20	Congo	henfrancis@gmail.com	Joyfest
Mohamed Konate M/18	France	mkona05@gmail.com	Joyfest Solar Fest
Yoanna Jones	France	joyoanna@gmail.com	Joyfest

Un clic sur la
compétition afin de
remplir les scores



FILL SCORES

Event
Reps
Finish
Best Set Time



FILL SCORES

Event
Reps
Finish Time
Best Set Time

... Mettre à jour les scores des athlètes

Lors du clic sur la compétition, nous POSTons l'id de la compétition au fichier scores.php qui se charge d'afficher un formulaire contenant les différents événements pour lesquels des scores doivent être remplis. Une fois les scores remplis les données sont POSTées et récupérées par check_scores.php. On va soit insérer de nouveaux scores ou mettre à jour les scores déjà entrés dans la base de données. ...

```
$sql = "SELECT * FROM scores WHERE athlete_id = $1 AND event_id = $2";
$result = pg_query_params($dbconn, $sql, [$athlete_id, $event_id]);

if(pg_num_rows($result)>0){
    $sql = "UPDATE scores SET finish_time = $1, best_set_time = $2, number_reps = $3
    WHERE athlete_id = $4 AND event_id = $5";
    $result = pg_query_params($dbconn, $sql, [$finish_time, $best_set_time, $number_reps, $athlete_id, $event_id]);
    if ($result == true){
        header("Location: register.php?success=Scores updated successfully&identifier=$partnerID");
    }else{
        header("Location: scores.php.php?error=update failed&$scoresData");
    }
}else{
    $sql = 'INSERT INTO scores(id, athlete_id, event_id, finish_time, best_set_time, number_reps)
    VALUES($1,$2,$3,$4,$5,$6)';
    $result = pg_query_params($dbconn, $sql, [$score_id, $athlete_id, $event_id, $finish_time, $best_set_time, $number_reps]);

    if ($result == true){
        header("Location: register.php?success=Scores updated successfully&identifier=$partnerID");
    }else{
        header("Location: scores.php?error=update failed&$scoresData");
    }
}
```

... Mettre à jour les scores des athlètes



REGISTER ATHLETES

Scores updated successfully

Name

Date Of Birth 

Sex ▼

Identified gender ▼

Email

Nationality

Register in ▼

ATHLETES REGISTERED


03

Gestion des recherches/filtres

Gestion des recherches/filtres

Nous avons créer une entrée où l'utilisateur peut rechercher un athlète à partir de son nom, Lorsqu'il appuie sur le bouton submit le nom entré est ensuite POSTé. Lors de l'affichage du tableau du leaderboard l'on va cacher toutes les rangées qui ont un nom différent de celui recherché par l'utilisateur.

Nous avons créer un menu où l'utilisateur peut filtrer les informations du leaderboard par rang ou sexe. Lorsqu'il appuie sur submit l'information est POSTé et stockée dans la variable \$filter ainsi on cache les rangées qui ne sont pas conformes au filtre.



MeFit

JOYFEST LEADERBOARD

Select

rank

male

female

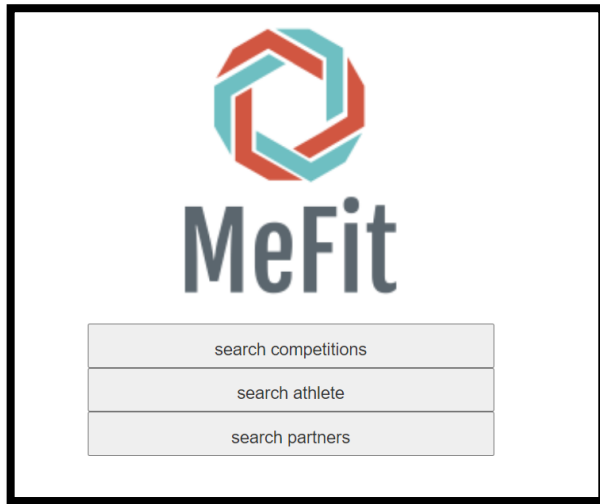
Enter the full name of the

filter

points	500 burpees in 15mn	maximum weight moved
1 (3 points)	rank: 2 (13:53:00) 500 reps best set time: 09:45:00	rank: 1 () 568 reps best set time:
Jean Francois M/25	2 (4 points)	rank: 3 () 505 reps best set time:

Gestion des recherches/filtres

Cette même procédure nous permet de rechercher des compétitions, des partenaires ou des athlètes.



NB: Dans les diapositives précédentes on recherchait des athlètes à partir d'un leaderboard. En cliquant sur search athlete on pourra rechercher parmi tous les athlètes présents dans notre base de données.

Gestion des recherches/filtres

```
for($i = 0 ; $i < count($leaderboard) ; $i++){
    if($filter == 'rank'){
        $v='';
    }else if($filter == 'male'){
        if($leaderboard[$i][4] == 'F'){
            $v = 'visibility:collapse';
        }else{
            $v='';
        }
    }else if($filter == 'female'){
        if($leaderboard[$i][4] == 'M'){
            $v = 'visibility:collapse';
        }else{
            $v = '';
        }
    }else{
        $v = '';
    }

    if(!empty($search)){
        if($leaderboard[$i][1] != $search){
            $v='visibility:collapse';
        }else{
            $v='';
        }
    }
}

?>
<tr style='<?php echo $v?>'>
```