

🔒 [howest-gp-wfa](#) / [st-2122-2-s2a-ee-cycling-start-BasixSW](#) Private

st-2122-2-s2a-ee-cycling-start-BasixSW created by GitHub Classroom

📄 MIT license

☆ 0 stars 🍴 0 forks

☆ Star

👁 Watch

Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

🔗 master

...



BasixSW ...

9 minutes ago



[View code](#)

Wielrennen | Examen Web Frontend Advanced AJ2122

Algemene instructie examen (must read)

1. Dit examen bestaat uit 5 deelopdrachten met telkens een hogere moeilijkheidsgraad. Werk de opdrachten in volgorde af om efficiënt te werken.
2. Er werd reeds HTML en CSS voorzien. Volg aandachtig de instructies in deze opdrachtomschrijving om onmiddellijk een mooie opmaak te verkrijgen.
3. Het is toegelaten om de CSS aan te passen. **De HTML behoudt zijn oorspronkelijke structuur, maar het is toegelaten om indien nodig attributen aan de reeds bestaande elementen toe te voegen.**
4. Het examen is **open boek**, dit betekent dat je de volledige cursus kan gebruiken, alsook alle oefeningen die tijdens de lessenreeks werden gebruikt en online bronnen.
5. Iedere vorm van samenwerking is **niet** toegelaten.
6. Maak regelmatig (per functionaliteit) een commit en push naar je online repo. Commit volgens de afspraken.

7. Iedere vaststelling van onregelmatigheid (o.a. GSM, spieken, afkijken) wordt conform het OER gemeld aan de betrokken student en aan de voorzitter van de examencommissie.
8. Dit examen maakt **60%** uit van de evaluatie voor de module Web Frontend Advanced (WFA).
9. De leerresultaten die tijdens dit examen getoetst worden, zijn:
- 2.3.1. Leidt de best mogelijke technische oplossing af voor het softwareproject
 - 2.3.2. Bepaalt de nodige infrastructuur voor de softwareapplicatie
 - 2.3.4. Bepaalt het eigen handelen uit de methodiek en doelstellingen die werden opgesteld
 - 3.2.11. Onderscheidt de verschillende componenten voor het maken van een functionele gebruikersinterface
 - 3.2.12. Brengt de componenten van de gebruikersinterface in verband met de gewenste functionaliteiten van de applicatie
 - 3.3.3. Selecteert de nodige tools en componenten voor het implementeren van de applicatie
 - 3.3.4. Maakt eenvoudige softwareapplicaties
 - 3.3.5. Selecteert de best passende gegevensstructuren
 - 3.3.6. Construeert een logische en samenhangende programmacode
 - 5.3.1. Gebruikt consequent de gemaakte codeerafspraken in afgebakende opdrachten
 - 5.3.2. Programmeert volgens de organisatiestandaarden
 - 6.2.5. Onderscheidt mogelijke oplossingen om tekortkomingen weg te werken

Scoringstabel

De puntenverdeling is de volgende:

Beoordeling	Punten
Naleven conventies	2 pt.
Formulierelementen	2 pt.
Codekwaliteit / Progressie	10 pt.
Pijlers van een webapplicatie (CSS, HTML, JavaScript)	1 pt.
Events	4 pt.
Gebruik Git(Hub)	2 pt.

Beoordeling	Punten
Json uitlezen (GitHub Pages)	2 pt.
Weergeven data (DOM manipulatie)	5 pt.
Kennis array-functions	2 pt.
TOTAAL	30 pt.

Algemene opdrachtomschrijving

Maak een interactieve webapplicatie waarin je meer te weten kan komen over enkele wielerploegen. De applicatie geeft de wielrenners en hun gegevens weer per ploeg.

Opdracht 1: dropdown vullen met categorieën (ploegen)

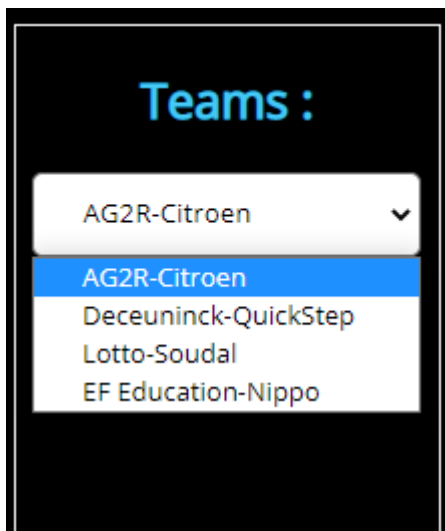
Lees de data van de ploegen uit. Kies hiervoor één van deze twee manieren om de data uit te lezen:

- Gebruik ofwel *Github Pages* om `docs/api/data.json` uit te lezen (online uitlezen)
- ... of gebruik de `js/data.js` (lokaal uitlezen)

Lokaal uitlezen kan, maar betekent een verlies van 2 punten op de totale score.

De data bevat een json-object waarin de wielrenners worden gecatalogeerd onder hun respectievelijke ploeg.

Eens je de data uitgelezen hebt, vul je het `select`-element op met de keywaardes uit het object dat je net uitlas.



☰ README.md



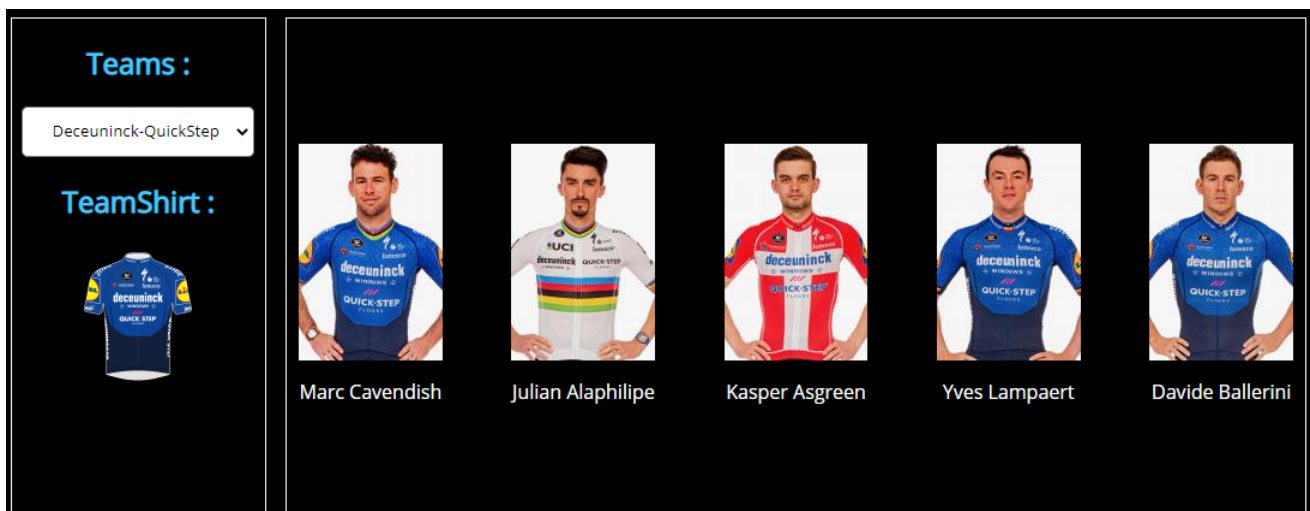
Opdracht 2: Weergeven renners en trui

Wanneer een gebruiker een ploeg heeft geselecteerd, wordt er een lijst van de wielrenners in die ploeg getoond in het HTML-element met class `overview`. Daarnaast toon je ook het truitje (shirt) van het team in de div met id `divShirt`.

Maak hiervoor HTML-elementen waarin je de naam van de renner toont, samen met een afbeelding ervan (`img` HTML-element).

Zorg ervoor dat het id attribuut van de `img` de `short` van de renner bevat.

De afbeeldingen zijn lokaal te vinden in de `img/` map.




Opdracht 3: Details van een renner tonen

Zorg er vervolgens voor dat, wanneer een gebruiker over een specifieke renner beweegt met de muis, de specifieke gegevens van de renner worden getoond in het HTML-element met class `details`.

Beweegt een gebruiker met de muis weg van de afbeelding, dan wordt de detailinhoud weer geleegd.

De details die je minimaal toont zijn:

- de naam van de renner
- de nationaliteit van de renner
- de leeftijd van de renner



Marc Cavendish Julian Alaphilipe Kasper Asgreen Yves Lampaert Davide Ballerini

Name
Kasper Asgreen


Nationality
Denmark

Age
26

Opdracht 4: Achievements van renners

De meeste renners die worden weergegeven hebben reeds een 'palmares'. Dit zijn de gewonnen dagkoersen.

Toon in het detailvenster per renner welke `achievements` hij op zijn palmares heeft wanneer een gebruiker over de renner hoovert, deze info komt bij de vorige details (zie 1.3). Voorzie hiervoor een `eigen opmaak` die `uniek` is.



Marc Cavendish Julian Alaphilipe Kasper Asgreen Yves Lampaert Davide Ballerini

Name
Kasper Asgreen


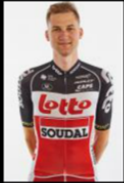
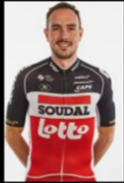


Nationality
Denmark

Age
26

Achievements

- Ronde van Vlaanderen
- E3 Saxo Bank Classic
- Kuurne-Brussel-Kuurne
- National Championships Denmark

Bij renners zonder `achievement` wordt de tekst "No Races won yet!" weergegeven in een andere kleur.

    					<p>Name</p> <p>Filipo Conca</p> <p>Nationality</p> <p>Italian</p> <p>Age</p> <p>23</p> <p>Achievements</p> <p>No Races won yet !</p>
Phillipe Gilbert	Tim Wellens	John Degenkolb	Caleb Ewan	Filipo Conca	


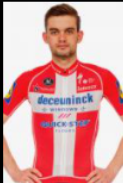


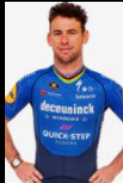
Opdracht 5: Renners sorteren

Voorzie tot slot een quizfunctionaliteit.

In het HTML-element met class `orderbox` staat reeds standaard de tekst *"Can you order on World Ranking? Click on the team shirt to see the result!"* om de aandacht van de gebruiker te trekken. Het is de bedoeling om hier een gesorteerde lijst te tonen van de huidige geselecteerde renners op basis van hun `worldRanking` (laag naar hoog).

De flow is als volgt:

- Standaard zie je de tekst *"Can you order on World Ranking? Click on the team shirt to see the result!"*
- Na het klikken op het teamshirt (dat je eerder toevoegde in `divShirt`) wordt het HTML-element met class `orderbox` opgevuld met de huidige geselecteerde renners in juiste volgorde volgens hun `worldRanking`. Maak hiervoor een kaartje (een `div`-element, bijvoorbeeld) dat andere `div`-elementen bevat om de `name` en `worldRanking` weer te geven, samen met een `img`-element voor de foto van de renner.
- Indien de gebruiker een nieuwe categorie van renners selecteert, dan verschijnt de tekst *"Can you order on World Ranking? Click on the team shirt to see the result!"* weer.

    				
Julian Alaphilipe WR : 4	Kasper Asgreen WR : 26	Davide Ballerini WR : 52	Yves Lampaert WR : 79	Marc Cavendish WR : 136


ScreenCast


Cycling Assignment

Teams :


AG2R-Citroen

TeamShirt :







Greg Van Avermaat




Tony Gallopin



Bob Jungels



Nans Peters



Olivier Naesen

Can you Order on World Ranking ?

Releases

No releases published

Packages

No packages published

Languages

JavaScript 84.5%

CSS 10.6%

HTML 4.9%

https://github.com/howest-gp-wfa/st-2122-2-s2a-ee-cycling-start-BasixSW

7/7