

# Rapport application NBA

## Objectif du projet

Pour le cours de Logiciel spécialisé (M1 SSD), nous avons pour objectif d'utiliser des fonctionnalités de R que nous ne maîtrisons pas. Comme la création d'une application avec Shiny ou alors la création d'un package. Nous avons aussi la possibilité de découvrir le Julia, qui est un langage de programmation de haut niveau très performant pour le calcul scientifique.

Pour développer le plus de compétences nous avons choisi de réaliser un projet en R et en Julia, afin de comparer les deux langages. Etant amateur de basket, on a choisi un jeu de données sur la NBA pour notre projet. Ensuite, on a décidé de réaliser une analyse statistique de ces données via une application web. En R l'application sera développée avec RShiny et en Julia à l'aide de Dash.

L'objectif est de réaliser deux applications avec les mêmes analyses statistiques et les mêmes fonctionnalités afin de comparer les outils Dash et RShiny.

## Réalisation

### Première page

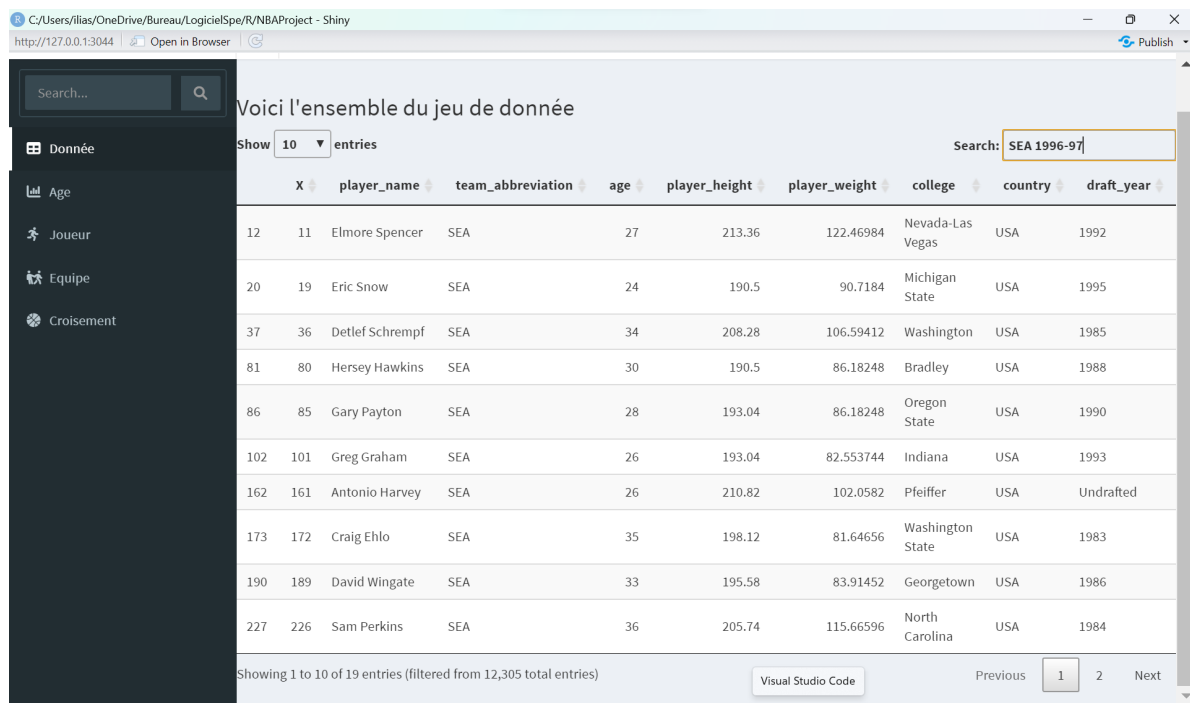
#### Les données utilisées

Les données utilisées lors de ce projet ont été trouvées sur kaggle (<https://www.kaggle.com/datasets/justinas/nba-players-data>). On y retrouve des informations sur tous les joueurs depuis 1996 comme leurs attributs physiques et leurs stats.

## Page RShiny

Afin de visualiser nos données, on a choisi d'afficher notre dataframe sur la première page de l'application, en y ajoutant des filtres.

Pour l'application Shiny il y avait une fonction (`dataTableOutput()`) qui permettait d'afficher le dataframe tout en choisissant le nombre de ligne à afficher et permettant à l'utilisateur de filtrer les données.



The screenshot shows an RShiny application window. On the left is a sidebar with a search bar and a menu containing 'Donnée', 'Age', 'Joueur', 'Equipe', and 'Croisement'. The main area displays a data table titled 'Voici l'ensemble du jeu de donnée'. The table has columns: 'X', 'player\_name', 'team\_abbreviation', 'age', 'player\_height', 'player\_weight', 'college', 'country', and 'draft\_year'. A search filter 'SEA 1996-97' is applied. The table shows 19 entries, with the first 10 displayed. At the bottom, it says 'Showing 1 to 10 of 19 entries (filtered from 12,305 total entries)' and includes a 'Visual Studio Code' button and pagination controls.

X	player_name	team_abbreviation	age	player_height	player_weight	college	country	draft_year
12	Elmore Spencer	SEA	27	213.36	122.46984	Nevada-Las Vegas	USA	1992
20	Eric Snow	SEA	24	190.5	90.7184	Michigan State	USA	1995
37	Detlef Schrempf	SEA	34	208.28	106.59412	Washington	USA	1985
81	Hersey Hawkins	SEA	30	190.5	86.18248	Bradley	USA	1988
86	Gary Payton	SEA	28	193.04	86.18248	Oregon State	USA	1990
102	Greg Graham	SEA	26	193.04	82.553744	Indiana	USA	1993
162	Antonio Harvey	SEA	26	210.82	102.0582	Pfeiffer	USA	Undrafted
173	Craig Ehlo	SEA	35	198.12	81.64656	Washington State	USA	1983
190	David Wingate	SEA	33	195.58	83.91452	Georgetown	USA	1986
227	Sam Perkins	SEA	36	205.74	115.66596	North Carolina	USA	1984

Figure 1: Visualisation des données : RShiny

## Page Dash.jl

Tandis qu'avec l'application Julia la fonction pour afficher le dataframe (`dash_datatable()`) n'avait pas les fonctionnalités de filtre. On a donc choisi d'ajouter des listes déroulantes (dropdown) afin de filtrer différentes variables (saison, équipe, nationalité).

## Page Rshiny VS Page Dash.jl

Tout d'abord il y a une différence sur la façon de filtrer le dataframe, avec RShiny dans la barre de filtre on peut filtrer plusieurs colonnes à la fois. Alors qu'avec Dash.jl il y a un filtre

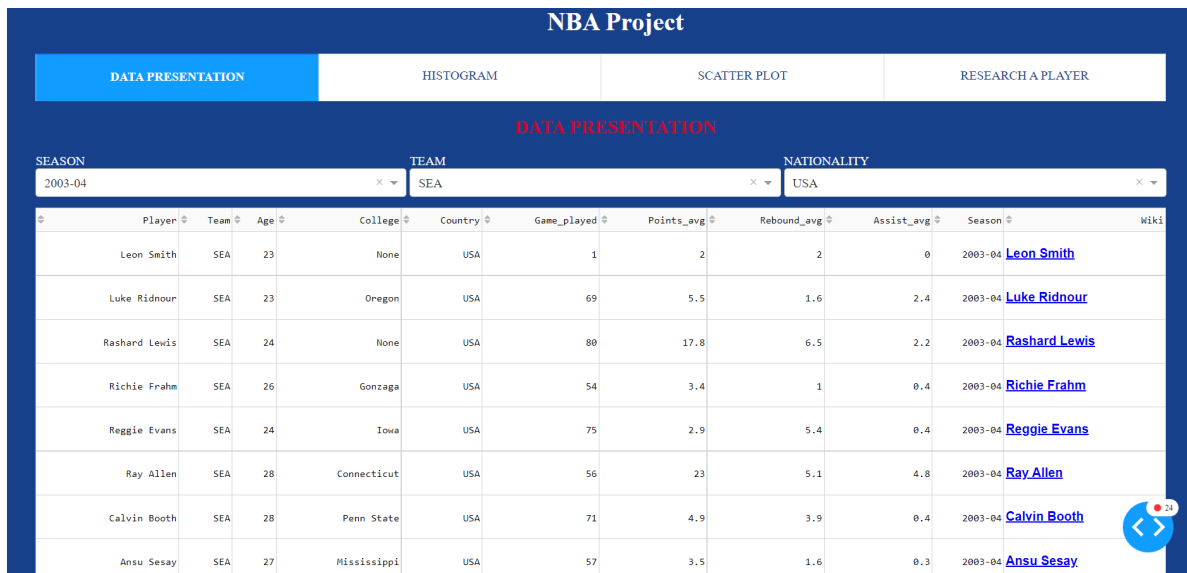


Figure 2: Visualisation des données : Julia

par variable. Sur l'application faites en Julia une colonne a été rajouté, elle permet d'aller directement sur la page web du joueur afin d'avoir plus d'information. Sur l'application RShiny toutes les variables du dataframe sont présente tandis que sur l'autre application un que les plus pertinentes ont été gardé.

## Difficultés rencontrées

### RShiny

L'avantage du filtre par défaut de RShiny s'est avéré être aussi un défaut, car malgré le fait de pouvoir filtrer plusieurs colonne, on ne pouvait pas choisir lesquelles. Par exemple si on voulait filtrer à l'aide d'une valeur numérique, on ne pouvait pas choisir sur quelle variable le faire, donc l'application n'effectuait pas la bonne action. On aurait pu faire les mêmes listes déroulantes que l'application Dash.jl mais nous n'avons pas essayé.

### Julia

La principale difficulté sur cette page est liée au données. En NBA des équipes qui existaient en 1996 n'existe plus maintenant et inversement. Il fallait donc faire des filtres dynamiques dans les listes déroulantes. Par exemple si on sélectionne la saison 1996-97, les équipes OKC et BKL ne doivent pas apparaître dans la liste déroulante des équipes. Grâce aux callback cette difficultés a été gérées.