



WELCOME TO THE SPORTS ANALYTICS CHALLENGE

To test out your skills and give you the chance to prove you're an expert in both science and sport, we're giving you a data set provided by Opta.

It's up to you to work out the data science/machine learning algorithm that best captures the information contained in the match data.

Your result should be in the **Python or R programming languages** and use Opta F24 files. Your code should be quick and short. You should use a specific package. The jury will look at how your code performs when ranking the finalists. Calculation time on test file: **5 seconds' maximum** for the test base file on a laptop computer.

Details:

- Each file must be sent separately in your participation space. A .zip file will not be considered.
- The code should only contain relative paths (not absolute path). All sent files will be downloaded and stored by the organizers in a common.

INSTRUCTIONS IN PYTHON:

- Version 3.5.
- Authorized packages: no restrictions, however use packages that can be installed using *pip*.
- Submit at least these 3 files: *main_psgx.py*, *install_psgx.py*, *readme.txt*. Each file must be sent separately.
 - The *main_psgx.py* file must contain a "Result" function which must take as argument an XML file identical to the test file provided and **return a.csv file (without header) named res_psgx.csv in the same folder, with the next 4 quantities separated by a comma**:
 - The player's identity must be a real corresponding to his ID.
 - The next event is a real number worth 1 or 0.
 - The next y and x are real numbers.



The test will consist in launching the "Result(xml_1)" function with:

- `xml_1 = lxml.etree.parse(path)`
 - `path` = path of an xml file identical to the one presented in the resources under the name: "Example test base file - f24-24-2016-xxxxxx-eventdetails_test_hackaton_2.xml"
- 1) The *install_psgx.py* file must contain the list of packages to be installed. For each package used, write the command: `os.system()`. Are you experienced? Do not hesitate to use the `pipenv` package and to indicate it.
 - 2) The *readme.txt* file must contain:
 - The packages used.
 - A brief description of each sub-folder contained in the submitted directory.

INSTRUCTIONS IN R:

- Version 3.4
 - Authorized packages: packages available on CRAN.
 - Submit at least 3 files: *main_psgx.R*, *install_psgx.R*, *readme.txt*. Each file must be sent separately.
- 1) The *main_psgx.R* file must contain a "Result" function which must take as argument an XML file identical to the test file provided and **return a.csv file (without header) named res_psgx.csv in the same folder, with the next 4 quantities separated by a comma**:
 - The player's identity must be a real corresponding to his ID which is an integer.
 - The next event is a real number worth 1 or 0.
 - The next y and x are real numbers.

The test will consist in launching the "Result(xml_1)" function with:

- `xml_1 <- xmlParse(path)`
 - `path` = path of an xml file identical to the one presented in the resources under the name: "Example test base file - f24-24-2016-xxxxxx-eventdetails_test_hackaton_2.xml"
- 2) Packages instructions for use: the *install_psgx.R* file must contain the list of packages to be installed. For each package used, write the command *install.packages()*.
 - 3) The *readme.txt* file must contain:
 - The packages used.
 - A brief description of each sub-folder contained in the submitted directory.

Good luck!

The Sports Analytics Challenge team

