1. Question 1
2. Question 2
3. Question 3
4. Question 4

For the same reasons as in Question 2, we use Beautifulsoup to scrap the franchise pages. More precisely, we use this package to navigate into the html tree, but we still need the regex package to make simple operations on string variables that can be found on the leaves of the tree.

* 1. Question 4.a

First of all, we want to scrap the basic team information. These are contained in the header of the franchise page. The html of this header can be found in the first "div" tag whose class is "mobile\_text". This tag is very badly organized because it consists in a succession of tags included in each other. Instead of having a list of sibling tags, all its successors are included in each other. Therefore, the Beautifulsoup code can be very tedious to write as each tag can only be reached by going down the hierarchy. Fortunately, we can avoid this problem with the function get\_text() that automatically searches the text element in the html code and concatenate them. When we apply it to the header of the franchise page, we obtain a string containing all the basic information of the team. If we take the example of Atlanta Hawks franchise page, we get the following text :

*Location: Atlanta, Georgia*

*Team Names: Atlanta Hawks, St. Louis Hawks, Milwaukee Hawks, Tri-Cities Blackhawks*

*Seasons: 66; 1949-50 to 2014-15*

*Record: 2584-2609, .498 W-L%*

*Playoff Appearances: 43*

*Championships: 1*

Once we have this text, we just have to use the appropriate regex expressions to retrieve all the information we are interested in. In our project, we scrap them all, except the team names (we will scrap them in another table). You can notice that some piece of information are not atomic. The "Location" field for example ,contains two information : city and state. Thus, we split all information that are not atomic into several pieces. *Location* is split into city and state, *Seasons* into number of seasons, first season and last season and *Record* into number of wins, number of losses and win-loss percentage. We choose to identify a season with the civil year of the beginning of the season, that is to say that the season 2014-15 is denoted as 201. Besides, we have to choose a unique identifier for each franchise, since each one can have several names throughout the years. This id is a three letters abbreviation of the franchise name as it is used in the url of the pages of the franchise. For example, the abbreviation of Atlanta Hawks is "ATL" because the url of its page is http://www.basketball-reference.com/teams/ATL/.

All the basic information of all teams are scraped by the script named *team\_scraper\_BS* and they are stored in the same table named *teams\_basic\_info.csv*.

* 1. Question 4.b

Now we want to retrieve all the team statistics by season. These data are available on the same page as the team basic information and are already well-organized as we can find them in a table below the header. As this scraping job is very similar to what we did in question 2, we will not describe it again. However, we can point out the fact that the name of the team can change throughout the seasons and that we don't use it as it is. Instead, we use a three letters abbreviation that can be found in the url pointing to the team roster of the corresponding season. The table containing the scraping result thus has both "Franchise\_id" and "Team\_id" columns. All statistics of all teams are scraped buy the script gathered in a csv file named *teams\_statistics.*

These data are not sufficient to answer all questions of part two. In order to get players experience, we also have to scrap rosters of all teams for each season. These data are available in the csv file named *roster*\_*statistics*.

1. Question 5

We thought that very well paid salaries may be explained by a phenomenon of exaggerated enthusiasm for specific players. To investigate this hypothesis, we decided to scrap the playoff statistics, that is to say statistics of teams that reached the final part of the competition. We thought that focusing on best teams and best players for each season could help us explain the highest salaries of players. These data were scraped on the page <http://www.basketball-reference.com/playoffs/> with the same technique as question 2 and 4, since they are also stored in a well-structured table on the website. Our recommendation to team owners is to consider players that appear in this page differently than other players. Even if we cannot verify our hypothesis that players appearing in playoffs roster are better paid than other, it would prove that high salaries are not decided in a reasoned and logical way. This piece of information could help them not to pay salaries that are not deserved and use this money for more efficient players.