# **Project Two**

Student: Olga Roginkin, Data Science 10/2018 – 05/2019

# Project Summary:

Read salary estimates as function of job titles, years of experience and job location from DICE.COM. Collect data for all 50 states, and for a range of years of experience. fOCUS on data science-related jobs.

# Data Sources:

1. Dice.com
2. Wikipedia.com
3. Csv file (job titles)

# Data Extraction Method

* We use dice.com to find salary estimations for several data science-related job titles. We create specific url strings by concatenating specific job title, years of experience and state abbreviation for each dice search. The resulting url follows the following pattern:

<https://www.dice.com/salary-calculator?title=Data%2BScientist&location=ca&experience=5>

* We use jupyter notebook to prepare all data, call a python function to scrape a webpage and return the result bac to jupyter notebokk
* First, we read a csv file, containing job titles and load its content into a list
* We have a list containing years of experience
* read Wikipedia.com to extract abbreviations of the 50 US states from the table of states in url = 'https://en.wikipedia.org/wiki/List\_of\_capitals\_in\_the\_United\_States'. This url contains a table, listing 50 US states with their abbreviations. We scrape this webpage, search for the table, and read its content.
* we create three loops: looping through all 50 states, all job titles, and all years of experience. Inside the most inner-loop, we populate a dictionary, containing all three variable and call a scrape function
* inside the scrape function, we construct the specific url string using all three search parameters, load the web page, scrape it looking for the results (min\_salary and average\_salary), populate them into a dictionary and return it back to the jupyter notebook
* In the example above, we are looking for a salary estimate for a job title ‘Data Scientist’, in CA, who has 5 years of experience.
* The result dictionary is added to a list of dictionaries which is written to an excel file