CS 210: Data Management for Data Science

Midterm 2

Spring 2023

Name: Khizar Saud NetID: KS881

This is a closed book, closed notes exam.

No electronic devices are permitted.

Name: Khizar Saud

1. (15 points) Write a regular expression to detect variable names in a language, where a variable name can be of any non-zero length, may only contain letters (upper or lowercase), digits, or underscores, and may not start with a digit or underscore.

r'[A-ZA-Z][A-Za-0-9]*'

2. (15 points) A string carries information about a book, in the form `"title",year,price`. For example, `"Girl, Interrupted",1994,$11.99`

The fields are separated by a comma with no whitespace around the commas, but there may be leading or trailing whitespaces around the whole string. The title is within double quotes, and may contain any character except double quotes. The year is a 4-digit number, for any year of the form 19xx or 20xx, up to and including 2021 but not after. The price has a dollar sign followed by at 1 or 2 digits (no leading zero), decimal point, and exactly 2 digits after the decimal.

Write a regular expression to extract the title, year, and price components. The extracted title must not include double quotes. The extracted price should include the dollar sign.

r'\s+ '[\w\s]*',(9\d\d|20[01]0-9)'[1-9[0-1]],

\s([[1-9][0-9]'.[0-9]

3. (15 points) Write a regular expression to detect whether a 5-character string is a palindrome, i.e. it reads the same forward and backward. The string can contain any characters, but whitespaces are to be ignored for the palindrome check.

Examples of palindromes: `'a1b1a'`, `' a 1b 1a'`, `'#a?a # '`

r'

r'(?:1)\1b\w\s* \w\s* \w\s* \w\s* \1\3)

Name: Khizar Saud

4. (15 points) Suppose we create a dataframe as follows:

```
df = pd.DataFrame(np.random.randint(0,100,24).reshape(4,6),
                    columns=list("abcdef"), index=list("xyzw"))
```

which produces the following dataframe:

```
    a   b   c   d   e   f
x   3  70  90  83  64  47
y  67   3  40  46  81  11
z  92  71  71  37  21  49
w  28  57  21  73  47  66
```

For each of the following, give the value of the result, or “error” if an error would occur.

(a) `df[a]`

error

(b) `df[x]` error

(c) `df['b']`

70
3
71
57

(d) `df['y']`

67
3
40
46
81

(e) `df[1]`

67, 3, 40, 46, 81, 11

(f) `df.loc[1]` error

Name: Khizar Saud

(g) `df.loc['a']`

3
67
92
26

(h) `df.loc['z']`

92, 71, 71, 37, 21, 49

(i) `df.iloc[2]`

92, 71, 71, 37, 21, 49

(j) `df['b']['y']`

3

(k) `df['z']['d']`

37

(l) `df['c'][1]`

40

(m) `df[1]['e']`

81

(n) `df.iloc['b']`

error

(o) `df.iloc['w']`

error

Name: Khizar Saud

5. (15 points)

Consider numeric values that are written with an exponent, such as 431e12, 1.23E5, .56e1, 0e1. The exponent notation may be written with uppercase E or lowercase e, with no spaces before or after.

The power must be a non-zero integer, without any leading zeros.

If there is a decimal point in the part preceding the exponent, there must be at least one digit after the decimal, so something like 23.e2 is not admissible.

Write a regular expression to tell whether a string is an acceptable numeric value with an exponent according to these rules.

r' [0-9]*(.)? (?=[0-9])e|E [0-9]+'

6. (10 points) Suppose we construct a Series as follows:

```
ser = pd.Series([ np.nan, 2, np.nan, 4, np.nan ])
```

(a) What would `ser.ffill()` return?

2, 2, 4, 4, 4

(b) What would `ser.bfill()` return?

[2, 2, 4, 4 NaN]

Name:

7. (15 points)

A dataframe named `scores` has rows for students, and columns for quizzes, and each row lists a student's scores on all quizzes. For example:

```
       q1  q2  q3
Alice  17  23  14
Bob     5  19  12
Carol  24  23  16
Derek  32  30  25
```

Extract into a new dataframe all rows for which that student's quiz total is greater than or equal to the average of all students' quiz totals.

For instance, in this example the students have quiz totals of 54, 36, 63, and 87. Then the mean of the totals is 60, and the new dataframe should contain the rows with totals of 63 (Carol) and 87 (Derek):

```
       q1  q2  q3
Carol  24  23  16
Derek  32  30  25
```

```
import numpy as np
importpandas pd

def Calcmean(row.ros)
        return Rows.n nanmean

 defcall(i (Student Avg Scores

      data{
      for mare inScores
    if  Calculate.mean(name ?Scores.mean
         data(name) = {name(d) addrows 02
                              Chad(names (a}
  df = pd. Dat,(fname Scres
   return
```