








Ronalds Rundāns
Konkursa 2. kārtas uzdevuma risinājums

Saite uz failiem:

<https://github.com/ronaldrsundans/VIAA/tree/main>

 ronaldrsundans Add files via upload	c5bfac4 · 1 hour ago	 3 Commits
 DatuzinatnieksKonkurss.ipynb	Add files via upload	1 hour ago
 visi_dati_ANGAL.csv	Add files via upload	1 hour ago
 visi_dati_MATOL.csv	Add files via upload	1 hour ago
 visi_dati_vidus_2025_.xlsx	Add files via upload	1 hour ago
 README		

Izstrādes vide : DataBricks Free Edition

(nepieciešams reģistrēties, augšupielādēt failus un tad var darba lapu DatuzinatnieksKonkurss.ipynb interaktīvi darbināt)

Failu apraksts:

DatuzinatnieksKonkurss.ipynb - darba lapa ar risinājumiem.

visi_dati_vidus_2025.xlsx - Excel fails ar vidusskolu eksāmenu datiem 2024./2025.m.g.

visi_dati_MATOL.csv - Eksāmena rezultāti matemātikas obligājai daļai CSV formātā.

visi_dati_ANGAL.csv - Eksāmena rezultāti angļu valodā CSV formātā.

Risinājums un attēli ar iegūto rezultātu

#1) Iegūt matemātikas obligātā eksāmena rezultātus dažādos griezumos - kā rezultāti atkarīgi no dažādiem #faktoriem (piemēram, no skolas atbilstošās urbanizācijas, no eksāmena daļas, no skolas tipa utt.);
#2) Iegūtos datu griezumus atainot vizuāli, izvēloties tam piemērotāko risinājumu;
#3) Analizēt iegūtos rezultātus - komentēt redzamās sakarības, izteikt spriedumus.
#Papildu punktus var iegūt, izdarot šo automatizēti visiem eksāmenu veidiem, ne tikai matemātikas #obligātajam eksāmenam (taču analizēt joprojām nepieciešams tikai matemātikas eksāmenu). Lūgums #paveikto noformēt vienā dokumentā .pdf formātā! Ja ir izveidots kāds interaktīvs vizuālais #risinājums, tad priecāsimies, ja tas pievienots dokumentam kā saite.

```
import pyspark
from pyspark.sql.types import *
from pyspark.sql.functions import concat, col, substring, lit

# File location and type
#Matemātikas eksāmena rezultāti
file_location = '/Workspace/Users/maxpower402@inbox.lv/visi_dati_MATOL.csv'
#Angļu valodas eksāmena rezultāti
#file_location = '/Workspace/Users/maxpower402@inbox.lv/visi_dati_ANGAL.csv'

# /Workspace/Users/maxpower402@inbox.lv/visi_dati_MATOL.csv
file_type = "csv"
# CSV options
infer_schema = "false"
first_row_is_header = "true"
delimiter = ","

#Tabulā uzspiež datu tipus kolonnām
customSchema = StructType([\
    StructField("Nr.p.k.", IntegerType(), True),\
    StructField("Norise", StringType(), True),\
    StructField("Iestāde", StringType(), True),\
    StructField("Pašvaldība", StringType(), True),\
    StructField("Punkti Zināšanas, izpratne un prasmes", StringType(), True),\
    StructField("Procenti Zināšanas, izpratne un prasmes", DoubleType(), True),\
    StructField("Punkti Kompleksu problēmu risināšana", StringType(), True),\
    StructField("Procenti Kompleksu problēmu risināšana", DoubleType(), True),\
    StructField("Punkti kopā", StringType(), True),\
    StructField("Procenti kopā", DoubleType(), True),\
    StructField("Tips", StringType(), True),\
    StructField("Urbanizācija", StringType(), True)])

df = spark.read.format(file_type) \
    .schema(customSchema) \
    .option("header", first_row_is_header) \
    .option("sep", delimiter) \
    .load(file_location)
```

```

#option("mode", "DROPMALFORMED") \
#noģēmu, citādi tukša tabula

#df = df.withColumn("Punkti Zināšanas, izpratne un prasmes", col("str").cast("int"))

display(df)

temp_table_name = "my_table"

df.createOrReplaceTempView(temp_table_name)

df.printSchema()

```

DatuzinatnieksKonkurss
+

Python
Tabs: ON
Last edit was 15 minutes ago

See performance (2)

df: pyspark.sql.connect.dataframe.DataFrame = [Nr.p.k.: integer, Norise: string ... 10 more fields]

Table

+

1²₃ Nr.p.k.

A^B_C Norise

A^B_C Iestāde

1	1	Eksāmens matemātikā (optimālais mācību satura apguves līmenis) 2024./2025. m.g.	""VisLe
2	2	Eksāmens matemātikā (optimālais mācību satura apguves līmenis) 2024./2025. m.g.	""VisLe
3	3	Eksāmens matemātikā (optimālais mācību satura apguves līmenis) 2024./2025. m.g.	""VisLe
4	4	Eksāmens matemātikā (optimālais mācību satura apguves līmenis) 2024./2025. m.g.	""VisLe
5	5	Eksāmens matemātikā (optimālais mācību satura apguves līmenis) 2024./2025. m.g.	""VisLe
6	6	Eksāmens matemātikā (optimālais mācību satura apguves līmenis) 2024./2025. m.g.	""VisLe
7	7	Eksāmens matemātikā (optimālais mācību satura apguves līmenis) 2024./2025. m.g.	""VisLe
8	8	Eksāmens matemātikā (optimālais mācību satura apguves līmenis) 2024./2025. m.g.	""VisLe
9	9	Eksāmens matemātikā (optimālais mācību satura apguves līmenis) 2024./2025. m.g.	""VisLe
10	10	Eksāmens matemātikā (optimālais mācību satura apguves līmenis) 2024./2025. m.g.	""VisLe
11	11	Eksāmens matemātikā (optimālais mācību satura apguves līmenis) 2024./2025. m.g.	""VisLe
12	12	Eksāmens matemātikā (optimālais mācību satura apguves līmenis) 2024./2025. m.g.	""VisLe
13	13	Eksāmens matemātikā (optimālais mācību satura apguves līmenis) 2024./2025. m.g.	""VisLe
14	14	Eksāmens matemātikā (optimālais mācību satura apguves līmenis) 2024./2025. m.g.	""VisLe
15	15	Eksāmens matemātikā (optimālais mācību satura apguves līmenis) 2024./2025. m.g.	""VisLe

↓

8,625+ rows | Truncated data | 2.28s runtime

Refreshed 14 minutes ago

```

root
|-- Nr.p.k.: integer (nullable = true)
|-- Norise: string (nullable = true)
|-- Iestāde: string (nullable = true)

```

```
%sql
```

```
/* Vidēji "Procenti kopā" katrā iestādē un to kārtotāju skaits*/
```

```
select `Iestāde` as iestade , avg(`Procenti kopā`) as procenti, count(`Iestāde`) as kartotaji  
from `my_table` group by iestade order by procenti desc
```

Table ▾		+		🔍 🗑 📄 📑	
		1.2 procenti	1.2 kartotaji		
1		95.23684210526316	38		
2		83.79144385026738	187		
3		80.8108108108108	37		
4		80.33333333333333	81		
5		78.66197183098592	71		
6		78.63636363636364	55		
7		74.35064935064935	77		
8	ākslu vidusskola" Emīla Dārziņa mūzikas skola"	74.125	8		
9		73.28	25		
10	ē "Jelgavas Valsts ģimnāzija"	73.22142857142858	140		
11		73.2	5		
12		72.63461538461539	52		
13		72.07407407407408	27		

```
import matplotlib.pyplot as plt
import pandas as pd

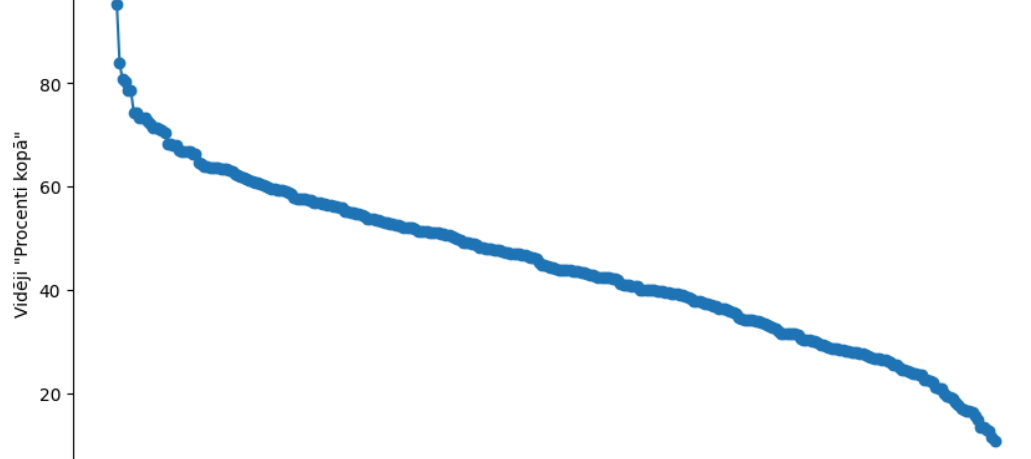
# Advanced data analysis with visualization
salary_trends = spark.sql('''
select `Iestāde` as iestade , avg(`Procenti kopā`) as procenti from `my_table` group by
iestade order by procenti desc
''').toPandas()

# Visualization of ...
plt.figure(figsize=(10, 5))
plt.plot(salary_trends['iestade'], salary_trends['procenti'], marker='o')
plt.xlabel('Iestāde')
plt.ylabel('Vidēji "Procenti kopā"')
plt.title('Vidēji "Procenti kopā" katrā iestādē')

plt.xticks(rotation=90)
plt.show()

#Komentārs: Ļoti labi vērtējumi ir tikai dažās iestādēs, bet zemie vērtējumi ir ļoti daudzās
iestādēs.
```

Vidēji "Procenti kopā"

[illegible]

```
%sql
```

```
/* vidēji "Procenti kopā" katrā pašvaldībā un kārtojēju skaits*/
```

```
select `Pašvaldība` as pasvaldiba , avg(`Procenti kopā`) as procenti, count(`Pašvaldība`) as  
kartotaji from `my_table` group by pasvaldiba order by procenti desc
```

Table

+

	^A _C pasvaldība	1.2 procenti	¹ ₃ kartotāji
1	Ādažu novads	62.59493670886076	79
2	Siguldas novads	59.23121387283237	173
3	Olaines novads	58.95652173913044	46
4	Līvānu novads	57.44642857142857	56
5	Preiļu novads	57.4210526315789...	57
6	Salaspils novads	55.27777777777778	36
7	Ķekavas novads	55.14423076923077	104
8	Mārupes novads	53.7378640776699...	206
9	Ropažu novads	53.2941176470588...	51
10	Madonas novads	52.2885906040268...	149
11	Valkas novads	51.83870967741935	31
12	Daugavpils	48.40384615384615	728
13	Augšdaugavas novads	48.05050505050505	99
14	Limbažu novads	47.09	100
15	Rīga	46.0282056040081...	5389

↓

▾

42 rows | 1.20s runtime

```

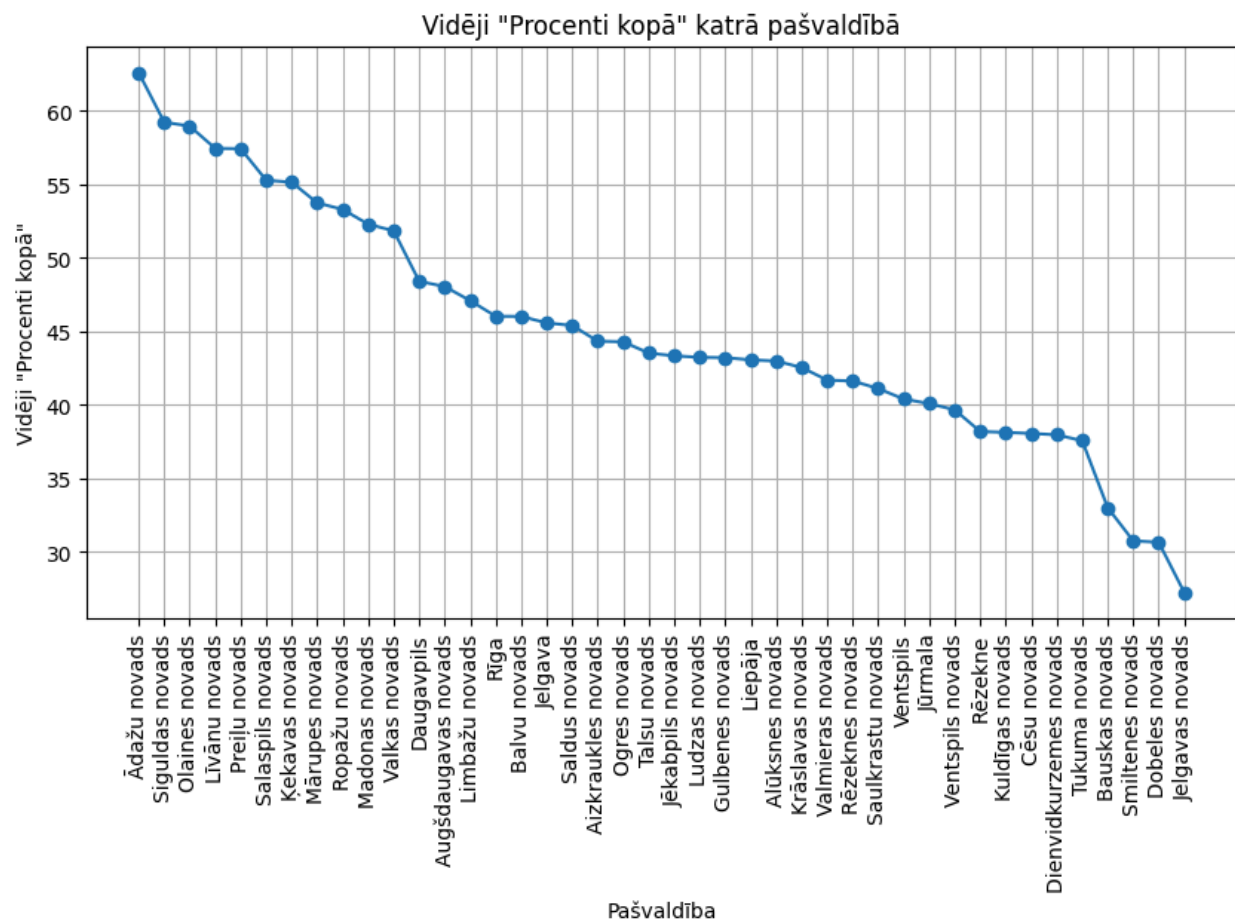
import matplotlib.pyplot as plt
import pandas as pd

# Advanced data analysis with visualization
salary_trends = spark.sql('''
select `Pašvaldība` as pasvaldiba , avg(`Procenti kopā`) as procenti from `my_table` group by
pasvaldiba order by procenti desc
''').toPandas()

# Visualization of ...
plt.figure(figsize=(10, 5))
plt.plot(salary_trends['pasvaldiba'], salary_trends['procenti'], marker='o')
plt.xlabel('Pašvaldība')
plt.ylabel('Vidēji "Procenti kopā"')
plt.title('Vidēji "Procenti kopā" katrā pašvaldībā')
plt.grid(True)
plt.xticks(rotation=90)
plt.show()

#Komentārs : Labākie rezultāti ir Vidzemē (Ādažu novads) un sliktākie ir Zemgalē(Jelgavas
novads).

```




```

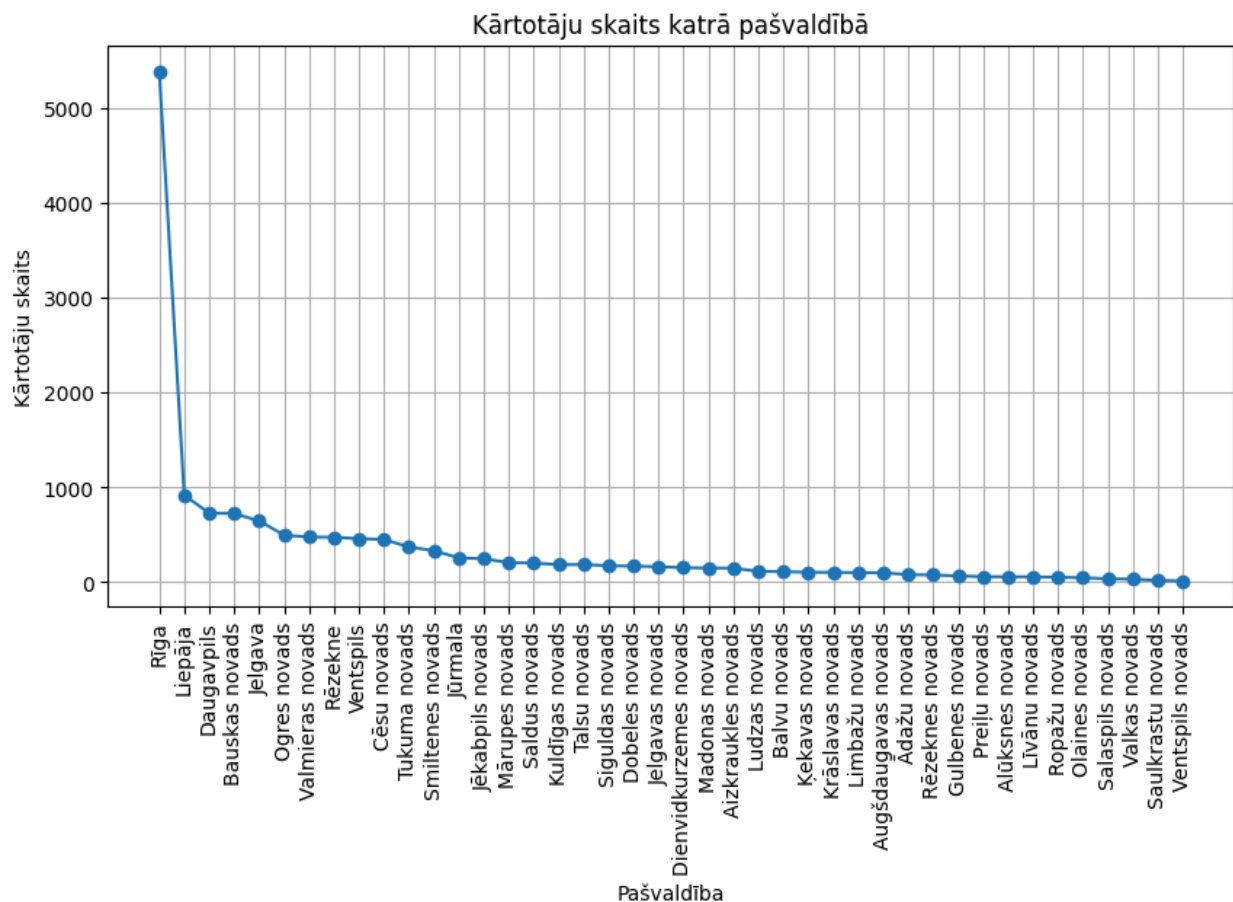
import matplotlib.pyplot as plt
import pandas as pd

# Advanced data analysis with visualization
salary_trends = spark.sql('''
select `Pašvaldība` as pasvaldiba , count(`Procenti kopā`) as procenti from `my_table` group
by pasvaldiba order by procenti desc
''').toPandas()

# Visualization of ...
plt.figure(figsize=(10, 5))
plt.plot(salary_trends['pasvaldiba'], salary_trends['procenti'], marker='o')
plt.xlabel('Pašvaldība')
plt.ylabel('Kārtotāju skaits')
plt.title('Kārtotāju skaits katrā pašvaldībā')
plt.grid(True)
plt.xticks(rotation=90)
plt.show()

# Komentārs: Rīga izceļas starp pašvaldībām ar kārtotāju skaitu(vismaz 5 reizes vairāk nekā
citur).

```



```
%sql
```

```
/* vidēji "Procenti kopā" katrā reģionā un kārtotāju skaits*/
```

```
select `Urbanizācija` as urbanizacija , avg(`Procenti kopā`) as procenti,  
count(`Urbanizācija`) as kartotaji from `my_table` group by urbanizacija order by procenti  
desc
```

Table

+

	^A _C urbanizacija	1.2 procenti	¹ ₃ kartotaji
1	Rīga	49.6558817227340...	4667
2	Novadu pilsētas	46.2662782727895...	2918
3	Valstspilsētas	44.44649446494465	4336
4	Lauki	36.35871743486974	1497
5	Tālmācības vidusskola	25.32345013477089	1484

↓

▼

5 rows | 1.09s runtime

This result is stored as `sqldf` and can be used in other [Python](#) and [SQL](#)

```

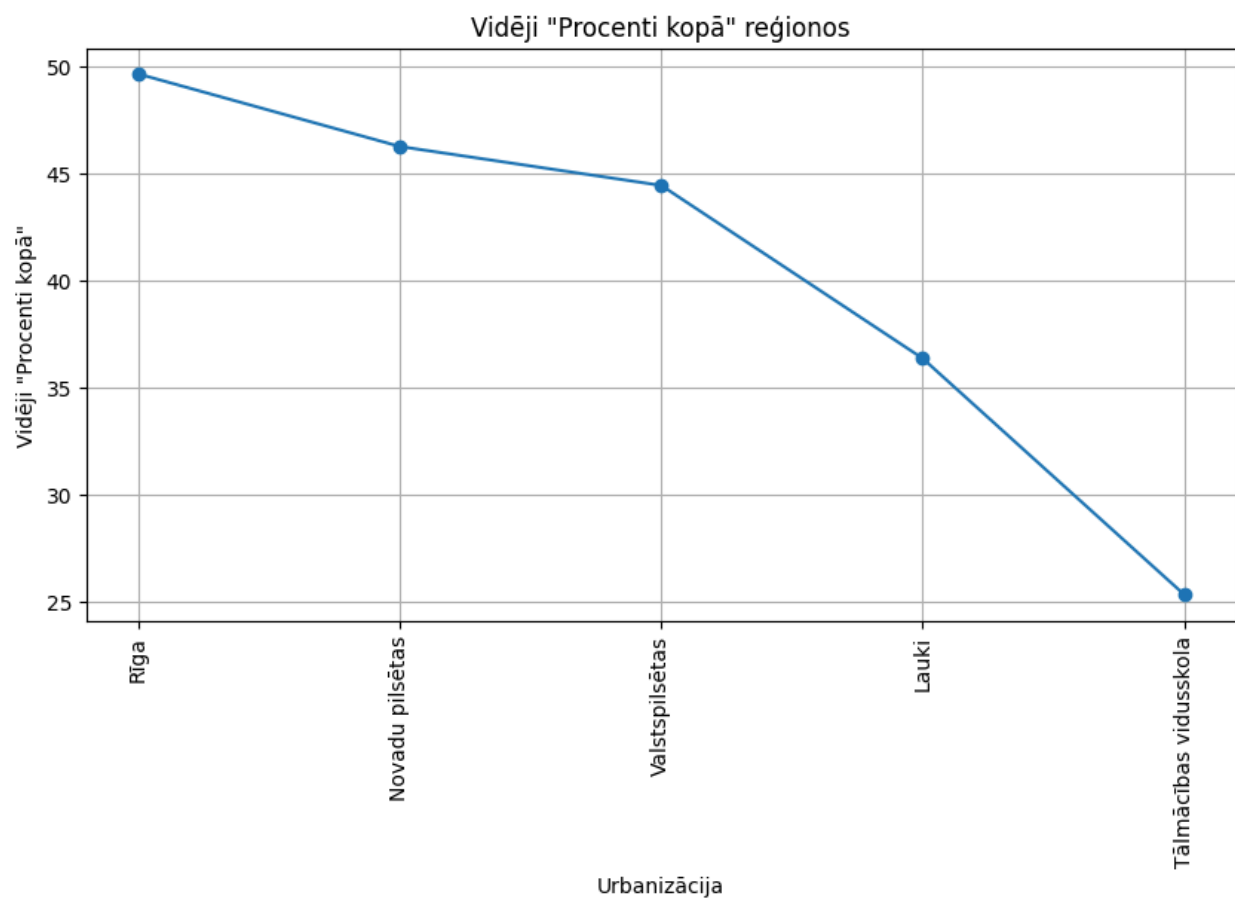
import matplotlib.pyplot as plt
import pandas as pd

# Advanced data analysis with visualization
salary_trends = spark.sql('''
select `Urbanizācija` as urbanizacija , avg(`Procenti kopā`) as procenti from `my_table` group
by urbanizacija order by procenti desc
''').toPandas()

# Visualization of ...
plt.figure(figsize=(10, 5))
plt.plot(salary_trends['urbanizacija'], salary_trends['procenti'], marker='o')
plt.xlabel('Urbanizācija')
plt.ylabel('Vidēji "Procenti kopā"')
plt.title('Vidēji "Procenti kopā" reģionos')
plt.grid(True)
plt.xticks(rotation=90)
plt.show()

#Komentārs : Skatoties pēc urbanizācijas, augstākie rezultāti ir Rīgā un zemākie ir Tālmācības
vidusskolās.

```



```
%sql
```

```
/* vidēji "Procenti kopā" katrā pašvaldībā */
```

```
select `Pašvaldība` as pasvaldiba, avg(`Procenti Zināšanas, izpratne un prasmes`) as izpratne
,avg(`Procenti Kompleksu problēmu risināšana`) as kompleksi,avg(`Procenti kopā`) as procenti,
count(`Pašvaldība`) as kartotaji from `my_table` group by pasvaldiba order by procenti desc
```

	Table ▾	+				
	pasvaldiba	1.2 izpratne	1.2 kompleksi	1.2 procenti	1.2 kartotaji	
1	Ādažu novads	70.49367088607595	38.65822784810127	62.59493670886076	79	
2	Siguldas novads	66.34682080924856	37.83815028901734	59.23121387283237	173	
3	Olaines novads	67.93478260869566	31.73913043478261	58.95652173913044	46	
4	Līvānu novads	61.8928571428571...	43.7857142857142...	57.44642857142857	56	
5	Preiļu novads	63.87719298245614	38.14035087719298	57.4210526315789...	57	
6	Salaspils novads	62.97222222222222	32.44444444444444	55.27777777777778	36	
7	Ķekavas novads	61.99038461538461	34.57692307692308	55.14423076923077	104	
8	Mārupes novads	61.3592233009708...	30.6990291262135...	53.7378640776699...	206	
9	Ropažu novads	59.6666666666666...	33.5294117647058...	53.2941176470588...	51	
10	Madonas novads	59.87919463087248	29.3154362416107...	52.2885906040268...	149	
11	Valkas novads	60.0645161290322...	27.0967741935483...	51.83870967741935	31	
12	Daugavpils	52.94505494505494	34.5824175824175...	48.40384615384615	728	
13	Augšdaugavas novads	52.1010101010101...	35.55555555555556	48.05050505050505	99	
14	Limbažu novads	53.55	27.54	47.09	100	
15	Rīga	51.99443310447207	27.8864353312302...	46.0282056040081...	5389	

↓ ▾

42 rows | 1.09s runtime

Refreshed 8 minutes ago

```

import matplotlib.pyplot as plt
import pandas as pd

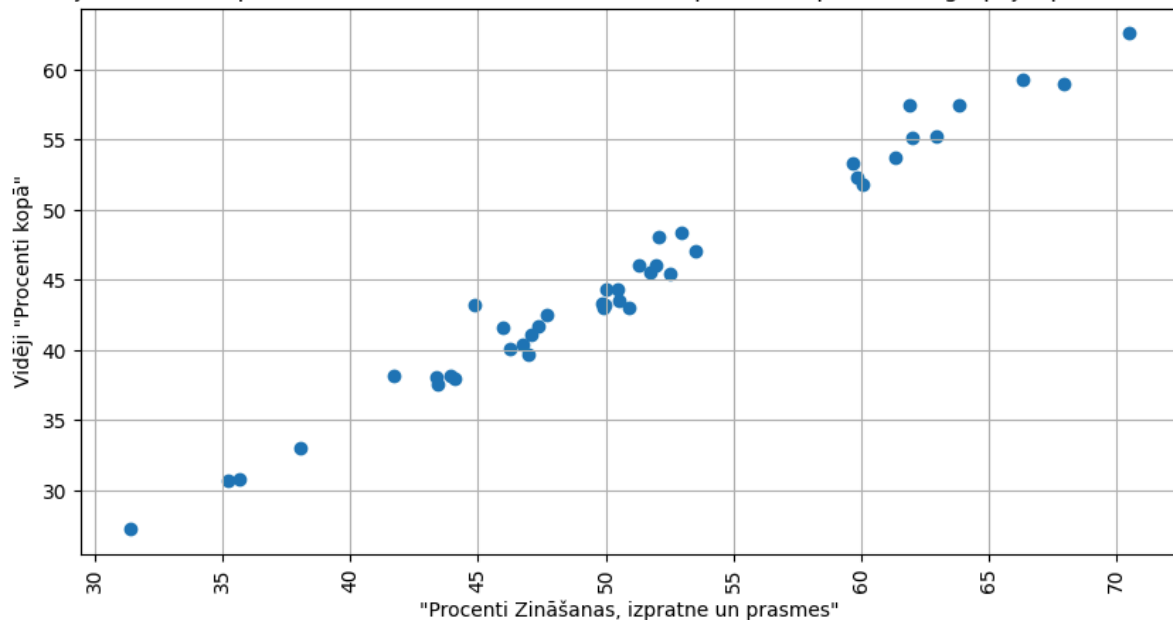
# Advanced data analysis with visualization
salary_trends = spark.sql('''
select `Pašvaldība` as pašvaldiba, avg(`Procenti Zināšanas, izpratne un prasmes`) as izpratne
,avg(`Procenti Kompleksu problēmu risināšana`) as kompleksi,avg(`Procenti kopā`) as procenti,
count(`Pašvaldība`) as kartotaji from `my_table` group by pašvaldiba order by procenti desc
''').toPandas()

# Visualization of ...
plt.figure(figsize=(10, 5))
plt.scatter(salary_trends['izpratne'], salary_trends['procenti'], marker='o')
plt.xlabel('"Procenti Zināšanas, izpratne un prasmes"')
plt.ylabel('Vidēji "Procenti kopā"')
plt.title('Vidēji "Procenti kopā" atkarība no "Procenti Zināšanas, izpratne un prasmes",
grupējot pa Pašvaldībām')
plt.grid(True)
plt.xticks(rotation=90)
plt.show()

#Komentārs: Var redzēt lineāru atkarību kā "Procenti kopā" mainās no iegūtā rezultāta eksāmena
daļā "Procenti Zināšanas, izpratne un prasmes". Tātad var prognozēt "Procenti kopā" zinot
"Procenti Zināšanas, izpratne un prasmes" vērtējumu.

```

Vidēji "Procenti kopā" atkarība no "Procenti Zināšanas, izpratne un prasmes", grupējot pa Pašvaldībām



```

import matplotlib.pyplot as plt
import pandas as pd

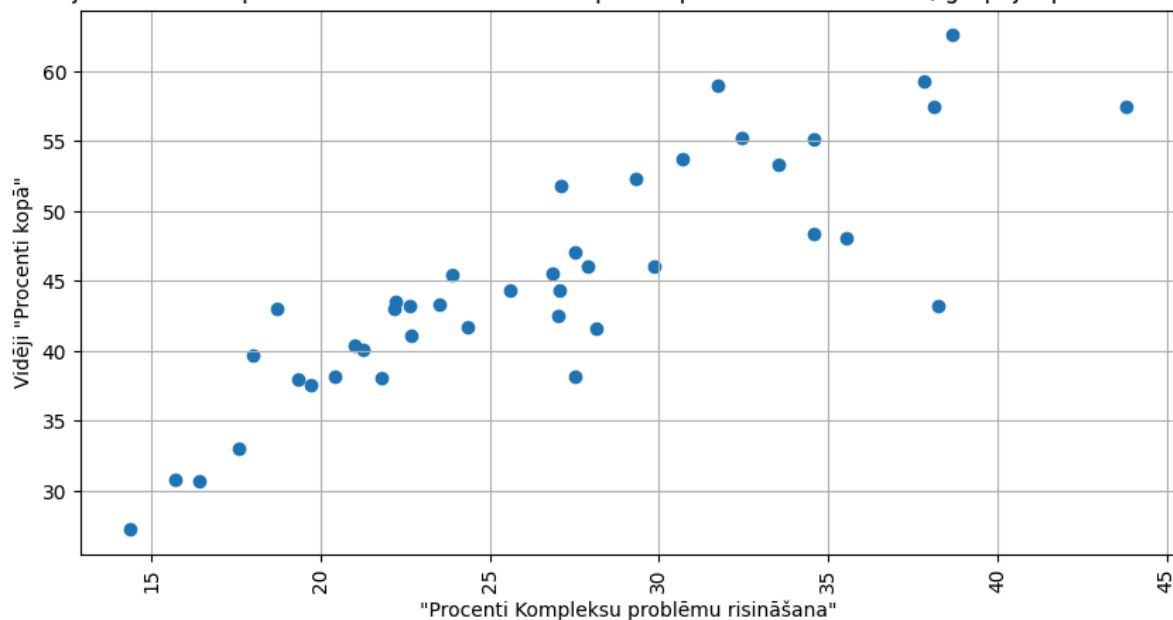
# Advanced data analysis with visualization
salary_trends = spark.sql('''
select `Pašvaldība` as pašvaldība, avg(`Procenti Zināšanas, izpratne un prasmes`) as izpratne
,avg(`Procenti Kompleksu problēmu risināšana`) as kompleksi,avg(`Procenti kopā`) as procenti,
count(`Pašvaldība`) as kartotaji from `my_table` group by pašvaldība order by procenti desc
''').toPandas()

# Visualization of ...
plt.figure(figsize=(10, 5))
plt.scatter(salary_trends['kompleksi'], salary_trends['procenti'], marker='o')
plt.xlabel('"Procenti Kompleksu problēmu risināšana"')
plt.ylabel('Vidēji "Procenti kopā"')
plt.title('Vidējo "Procenti kopā" atkarība no "Procenti Kompleksu problēmu risināšana",
grupējot pa Pašvaldībām')
plt.grid(True)
plt.xticks(rotation=90)
plt.show()

#Komentārs: Var redzēt lineāru atkarību kā vidēji "Procenti kopā" mainās no iegūtā rezultāta
eksāmena daļā "Procenti Kompleksu problēmu risināšana", bet nav tik izteikti kā iepriekšējā
grafikā.

```

Vidējo "Procenti kopā" atkarība no "Procenti Kompleksu problēmu risināšana", grupējot pa Pašvaldībām



Papildus punkti





Ja nomaina CSV failu, kas satur angļu valodas eksāmena rezultātus, tad var lietot darba lapu kā matemātikas eksāmenam, bet tabulām atšķiras kolonnu skaits un tādēļ arī atšķirās kolonnu secība. Kolonnu nosaukumi tiek apstrādes sākumā "iešūti", lai datu tipi uzspiesti šīm kolonnām, tādēļ būtu jāievieš vienāds šablons visām tabulām vai risinājums jāpapildina ar kārtošanu.

```
from pyspark.sql.functions import concat, col, substring, lit

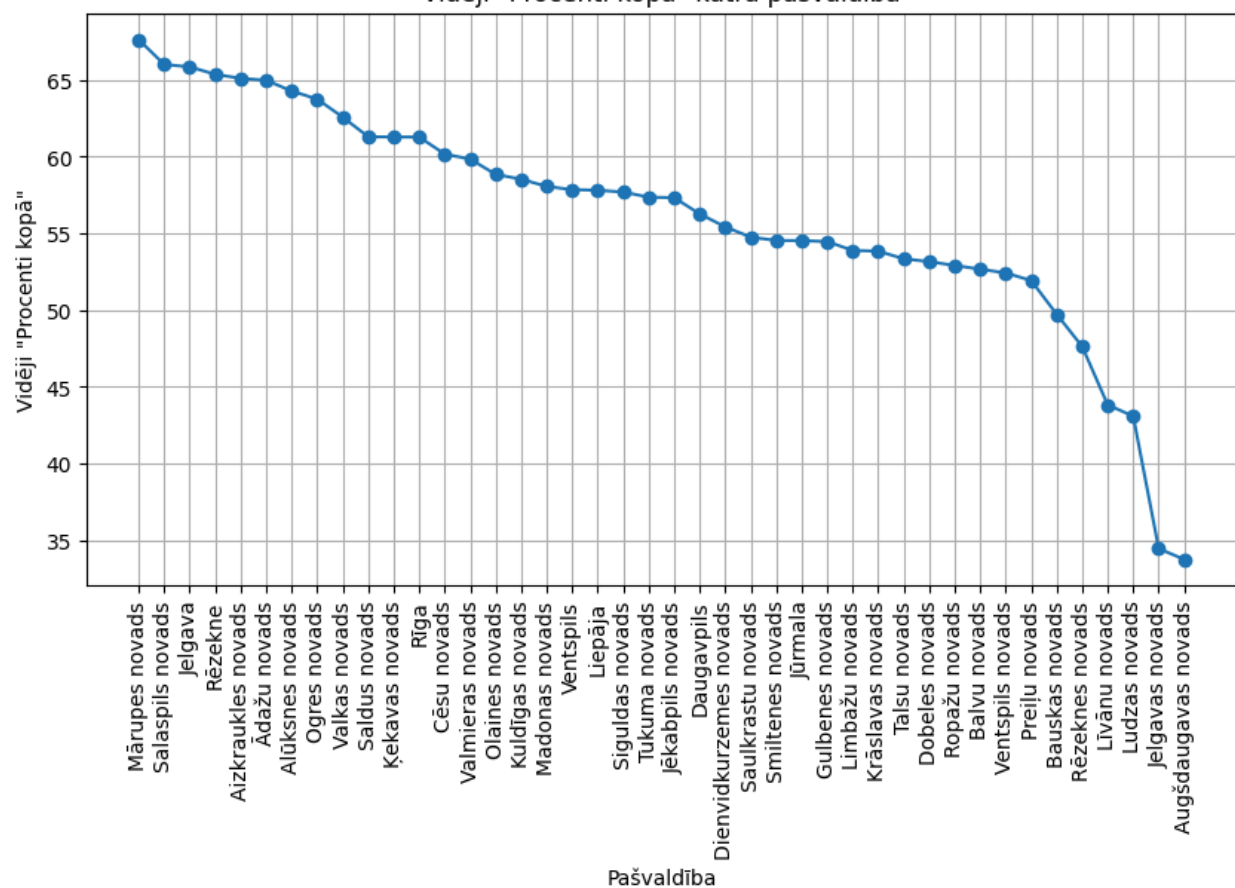
# File location and type
#Matemātikas eksāmena rezultāti
#file_location = '/Workspace/Users/maxpower402@inbox.lv/visi_dati_MATOL.csv'
#Angļu valodas eksāmena rezultāti
file_location = '/Workspace/Users/maxpower402@inbox.lv/visi_dati_ANGAL.csv'

#/Workspace/Users/maxpower402@inbox.lv/visi_dati_MATOL.csv
file_type = "csv"
# CSV options
infer_schema = "false"
first_row_is_header = "true"
delimiter = ","
```

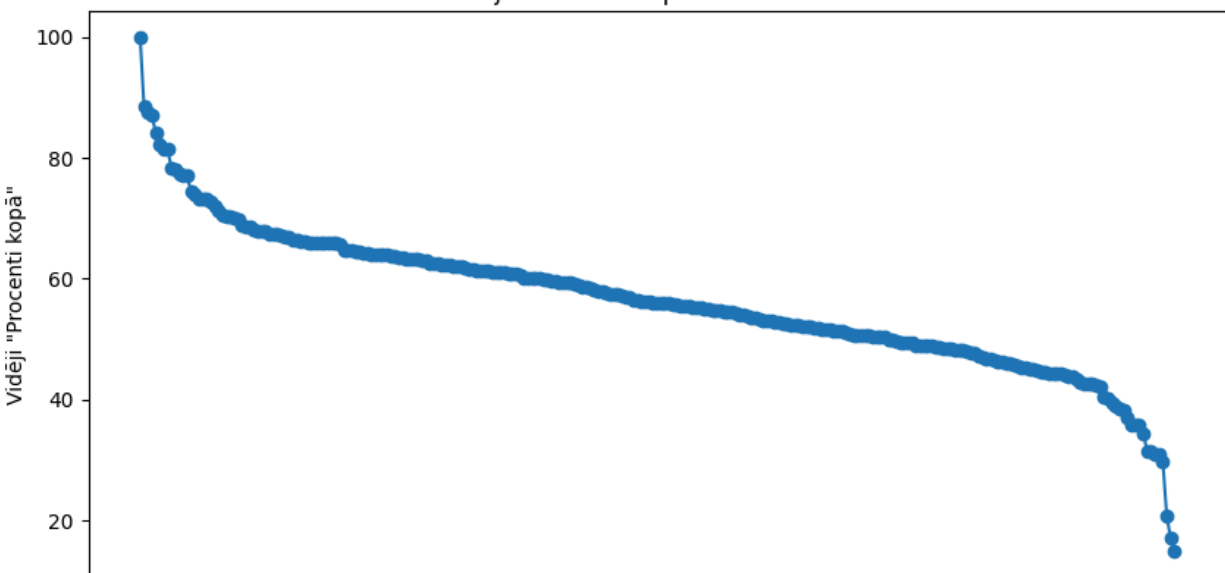
> df: pyspark.sql.connect.dataframe.DataFrame = [Nr.p.k.: integer, Norise: string ... 10 more fields]

Table ▾		+		   	
	1 ² ₃ Nr.p.k.	Δ ^B _C Norise	Δ ^B _C Iestāde		
1		1 Eksāmens angļu valodā (augstākais mācību satura apguves līmenis) 2024./2025. m...	""VisLatvijas viduss		
2		2 Eksāmens angļu valodā (augstākais mācību satura apguves līmenis) 2024./2025. m...	""VisLatvijas viduss		
3		3 Eksāmens angļu valodā (augstākais mācību satura apguves līmenis) 2024./2025. m...	""VisLatvijas viduss		
4		4 Eksāmens angļu valodā (augstākais mācību satura apguves līmenis) 2024./2025. m...	""VisLatvijas viduss		
5		5 Eksāmens angļu valodā (augstākais mācību satura apguves līmenis) 2024./2025. m...	""VisLatvijas viduss		

Vidēji "Procenti kopā" katrā pašvaldībā



Vidēji "Procenti kopā" katrā iestādē

[illegible]

