

Příklad výrobní firmy, zpracování a analýza dat, test Logio s.r.o.

K testu lze přistupovat různě. Není nezbytně nutné dosáhnout u jednotlivých úkolů kompletního řešení, ale lze prezentovat myšlenku postupu k dosažení řešení.

Zadání

Zobrazit zadání ▾

Řešení

Náklady na výrobu jednotlivých náhradních dílů

Předpříprava dat

1. příprava dat pro analýzu nákladů na výrobu jednotlivých náhradních dílů:

```
#creating df_ciselniky and df_kusovnik dataframe
df_ciselniky = pd.read_excel(excel_file, sheet_name='ciselniky', usecols='G:H', ski
df_kusovnik = pd.read_excel(excel_file, sheet_name='matice_vyroby')
```

2. spojení tabulek ciselniky a matice výroby + přidání částečného součtu nákladů komponent:

```
df_naklady = pd.merge(df_kusovnik, df_ciselniky, how="left", on = "ID_komponenty")
df_naklady['Celkova_cena_komponenty'] = df_naklady['Mnozstvi'] * df_naklady['Poriz
```

	ID_produktu	ID_komponenty	Mnozstvi	Porizovaci_cena	Celkova_cena_komponenty
14	ND103	KO135	4	6,600	26,400
15	ND105	KO111	3	4,650	13,950
16	ND105	KO135	3	8,150	24,450
17	ND106	KO028	3	11,300	33,900
18	ND106	KO040	2	600	1,200
19	ND106	KO094	1	13,350	13,350
20	ND106	KO019	2	7,800	15,600
21	ND107	KO029	2	11,400	22,800
22	ND107	KO021	3	14,200	42,600
23	ND108	KO108	2	10,650	21,300
24	ND108	KO196	3	7,900	23,700

3. groupby funkce a agregace pro celkovou cenu komponenty:

```
df_naklady.groupby('ID_produktu')[['Mnozstvi', 'Celkova_cena_komponenty']].sum().r
```

	ID_produktu	Mnozstvi	Celkova_cena_komponenty
0	ND101	7	29,650
1	ND102	3	12,650
2	ND103	8	52,650
3	ND104	4	20,600
4	ND105	8	52,000
5	ND106	8	64,050
6	ND107	5	65,400
7	ND108	5	45,000
8	ND109	10	64,800
9	ND110	8	43,600

Vývoj výroby za poslední rok po měsících v jednotlivých závodech

1. příprava dat - tabulka výroba:

```
# extracting manufacturing data from csv_excel file
wb = op.load_workbook(excel_file)
sheet = wb['vyroba_text']

#creating data frame vyroba
csv_data = get_csv_data(sheet)
df_vyroba = pd.DataFrame(csv_data[1:], columns=csv_data[0])
```

jelikož data byla ve formátu csv, bylo nutné je přeformátovat do tabulky. K tomu jsem použil custom funkce která vytvořila list listů, kde každý list obsahoval jeden řádek z csv souboru.

```
def get_csv_data(sheet):
    csv_data = []
```

```
for row in sheet.iter_rows(values_only=True):
    row_list = []
    for element in row[0].split(';'):
        row_list.append(element)
    csv_data.append(row_list)
return csv_data
```

2. přidání sloupce 'Rok-Mesic' a převedení dat do formátu datetime

```
# Converting datatypes (default is object)
df_vyroba['Datum'] = df_vyroba['Datum'].apply(date_parser)
df_vyroba['Mnozstvi'] = pd.to_numeric(df_vyroba['Mnozstvi'],).astype('Int64')
# Adding colum 'mesic';
df_vyroba['Mesic'] = df_vyroba['Datum'].dt.month
df_vyroba['Rok'] = df_vyroba['Datum'].dt.year
df_vyroba['Rok-Mesic'] = df_vyroba['Datum'].dt.strftime('%Y-%m')
```

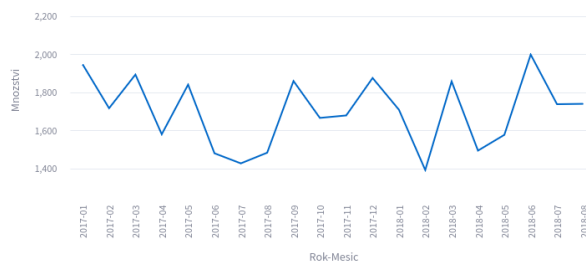
3. groupby (ID_zavodu, Rok-mesic) --> unikátní záznamy pro měsíc a danou továrnu aggregate množství vyrobených dílů pro danou továrnu

```
df_vyroba_mesice = df_vyroba.groupby(['ID_zavodu', 'Rok', 'Rok-Mesic'])['Mnozstvi']
df_vyroba_mesice_tovarny = pd.merge(df_vyroba_mesice, df_tovarny, on='ID_zavodu',
```

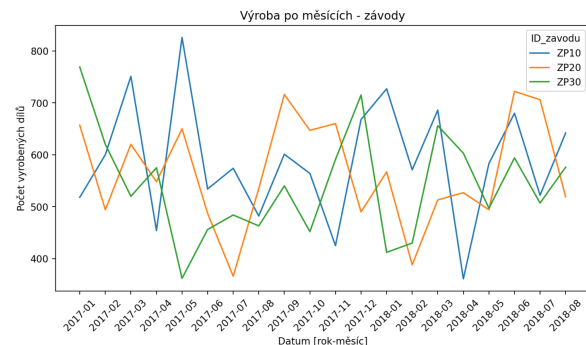
4. join s číselníky --> pro název továrny dle lokality

	ID_zavodu	Místo	Rok-Mesic	Mnozstvi
0	ZP10	Plzeň	2017-01	518
1	ZP10	Plzeň	2017-02	600
2	ZP10	Plzeň	2017-03	751
3	ZP10	Plzeň	2017-04	454
4	ZP10	Plzeň	2017-05	826
5	ZP10	Plzeň	2017-06	534
6	ZP10	Plzeň	2017-07	574
7	ZP10	Plzeň	2017-08	482
8	ZP10	Plzeň	2017-09	601
9	ZP10	Plzeň	2017-10	564

Celková výroba po měsících ve všech závodech



Celková výroba po měsících v jednotlivých závodech



Dodávky se zpožděním

1. Načtení dat z excelu do pandas dataframe 2. Přidání sloupce 'Zpozdeno', 'Deadline' a 'Dodaci_doba'

```
df_dodavky['Zpozdeno'] = (df_dodavky['Datum_dodání'] - df_dodavky['Datum_objednání'])
df_dodavky['Deadline'] = (df_dodavky['Datum_objednání'] + pd.Timedelta(days=7))
df_dodavky['Dodaci_doba'] = df_dodavky['Datum_dodání'] - df_dodavky['Datum_objednání']
df_dodavky_zpozdeno = df_dodavky.query('Zpozdeno == True')
```

Časové období: 2017-01-01 00:00:00 2018-08-31 00:00:00

Celkový počet dodávek v datasetu: 2058

Celkový počet zpožděných dodávek v datasetu: 239

% Zpožděných dodávek: 11.61

	ty	Mnozství	Datum_objednani	Datum_dodani	Zpozdeno	Deadline	Dodaci_doba
95		11	2017-08-02 00:00:00	2017-08-16 00:00:00	<input checked="" type="checkbox"/>	2017-08-09 00:00:00	14 days
100		7	2017-06-30 00:00:00	2017-07-12 00:00:00	<input checked="" type="checkbox"/>	2017-07-07 00:00:00	12 days
108		1	2017-05-14 00:00:00	2017-05-23 00:00:00	<input checked="" type="checkbox"/>	2017-05-21 00:00:00	9 days
115		15	2017-06-22 00:00:00	2017-07-02 00:00:00	<input checked="" type="checkbox"/>	2017-06-29 00:00:00	10 days
124		13	2018-08-30 00:00:00	2018-09-10 00:00:00	<input checked="" type="checkbox"/>	2018-09-06 00:00:00	11 days
128		4	2018-01-07 00:00:00	2018-01-22 00:00:00	<input checked="" type="checkbox"/>	2018-01-14 00:00:00	15 days
130		3	2017-12-22 00:00:00	2018-01-02 00:00:00	<input checked="" type="checkbox"/>	2017-12-29 00:00:00	11 days
134		16	2017-10-30 00:00:00	2017-11-13 00:00:00	<input checked="" type="checkbox"/>	2017-11-06 00:00:00	14 days
141		14	2017-07-13 00:00:00	2017-07-23 00:00:00	<input checked="" type="checkbox"/>	2017-07-20 00:00:00	10 days
143		3	2017-01-19 00:00:00	2017-02-02 00:00:00	<input checked="" type="checkbox"/>	2017-01-26 00:00:00	14 days

Počet dodávek od dodavatele a počet zpožděných dodávek

```
df_dodavky.groupby('ID_dodavatele')['ID_komponenty'].count().reset_index(name='Pocet_objednavky')
df_dodavky_zpozdeno.groupby('ID_dodavatele').size().reset_index(name='Pocet_zpozdeno')
```

	ID_dodavatele	Pocet_objednavek
0	102ADF	527
1	25KOL9	535
2	47EFKT	509
3	999DJT8	487

	ID_dodavatele	Pocet_zpozdenych_objednavek
0	102ADF	60
1	25KOL9	64
2	47EFKT	64
3	999DJT8	51

Počet náhradních dílů od každého dodavatele

```
df_dodavky.groupby('ID_dodavatele')['Mnozství'].sum().reset_index(name='Celkove_mnozstvi')
```

	ID_dodavatele	Celkove_mnozstvi
0	102ADF	5,698
1	25KOL9	5,527
2	47EFKT	5,325
3	999DJT8	4,955

Statistický popis zpoždění

```
st.dataframe(df_dodavky_zpozdeno['Dodaci_doba'].describe())
```

	Dodaci_doba
count	239
mean	11 days 11:32:53.221757322
std	2 days 06:41:08.251690338
min	8 days 00:00:00
25%	10 days 00:00:00
50%	11 days 00:00:00
75%	13 days 00:00:00
max	15 days 00:00:00