Data Prep Exercise

These exercise are challenging [especially the last one]. Its not a test , treat it as a learning experience if you are not able to do it . Dont let this dishearten you . Enjoy and challenges and feel free to discuss with each other .

There is a solution uploaded to LMS , you can have a look after you have tried your hands on these . Your solution might different from the one shared , its perfectly fine until your results match. There is no one definte of doing anything . But do take a note if you find the shared solution to be more clean or efficient.

---

1. Create a data frame using following code

```
import pandas as pd
import numpy as np
d=pd.DataFrame({'id':np.random.choice(range(1,100),30,replace=False),'x':np.ran
dom.randint(1,100,30),
                'y':np.random.randint(1,100,30)})
```

Write the code to find `id` corresponding to maximum absolute difference between `x` and `y` . Then write code to find how many observations have strictly lower value of x , than the value of x corresponding to that id.

Additional info :

Example with a smaller data frame

id	x	y
34	**99**	**56**
1	3	9
7	*11*	*98*
23	**45**	**1**
28	2	16

id corresponding to maximum absolute difference between x and y : `7`

number of rows with value of `x` strictly higher than `11` [value of `x` corresponding to `id` = `7`] = `2`

---

2. Create a dataframe using following code

```
import pandas as pd
import numpy as np
from datetime import date

d1=pd.to_datetime('23-1-2020').toordinal()
d2=pd.to_datetime('23-12-2020').toordinal()

df=pd.DataFrame({
    'date':[date.fromordinal(np.random.randint(d1, d2)) for i in range(100)],
    'sales':np.random.randint(100,500,100),

 'category':np.random.choice(['Apparels','Cosmetics','Toys','Consumables'],100)
})
```

Write code to find average sales across months . Write code to find which category had minimum sales for the second quarter .

Additional Information :

=>You can extract different components [month, year, week etc] from a datetime type pandas series using following data[col_name].dt.month .

=> You can convert an object type column containing dates to datetime type by using pd.to_datetime

---

3. Import data `coupon_item.csv` . Create a data set with following summaries at `coupon_id` level.
    1. Count of how many times a `coupon_id` occurs in the dataset [*Hint : make use of* `value_counts` *and then use* `reset_index` *on the result*]
    2. Number of unique items for each coupon [each item has an unique `item_id`]
    3. Count of each `category` for every coupon [*Hint: Make use of* `crosstab` *and use* `reset_index` *on the result*]
    4. Number of unique categories for each coupon
    5. Max Frequency brand code for each coupon [Identified with column name `brand`]
    6. Number of brands for each coupon which have frequency higher than 10% of how many times that coupon is present in the data
    7. Difference between frequencies of highest occurring and second highest occurring brands as percent of total frequency of the coupon . [e.g. total frequency of the coupon in data is 100. highest occurring brand has frequency 50 and second highest has frequency 30 . then value of this summary will be (50-30)/100 =0.2]

    Additional Suggestions/Info :

    => All of this will not be done in one go , you can create summaries for each sub question and then merge them with previous results

=> This exercise is an example of creating summary features when you are given multiple characteristics to work with . You could very well merge this data back to a bigger training set which has multiple occurrences of each `coupon` across multiple transactions [or customer].
