In [61]:
```
import psycopg2
import pandas as pd
```

In [62]:
```
hostname = 'localhost'
database = 'DataEngineering'
username = 'postgres'
pwd = 'Sadhan@1'
port_id = 5432
conn = None
cur = None
```

In [76]:
```
try:
    conn = psycopg2.connect(
                host = hostname,
                dbname = database,
                user = username,
                password = pwd,
                port = port_id)
    cur = conn.cursor()

    cur.execute('DROP TABLE IF EXISTS employee1')
    create_script = '''CREATE TABLE if not exists employee1(id int ,
                                        name varchar(40),
                                        salary int,
                                        dept_id varchar(30),
                                        branch varchar(32),
                                        account_Number varchar(32),
                                        Transaction_id int,
                                        Transaction_status varchar(42),
                                        Transaction_limit  int,
                                        Transaction_year int,
                                        Transaction_month varchar(21),
                                        Transaction_day varchar(32),
                                        Transaction_hour int) '''


    cur.execute(create_script)
    insert_script = '''insert into employee1(id,name,salary,dept_id,branch,account_Num
    values (101,'James', 12000, 'D1',NULL,'SBI00123',10235,'successful',50000,2022,'ma
    (102,'suresh', 25000, 'D1','secunderabad','SBI00133',10205,'unsuccessful',15000,20
    (103,'raju', Null, 'D2','boduppal','SBI00122',10265,'successful',25000,2022,'auges
    (104,'rani', 65000, 'D3','ramanthapur','SBI00165',10005,'successful',90000,2022,'j
    (105,'mahesh', 35000, 'D6','amberpet','SBI00135',10665,'unsuccessful',10000,2021,'
    (106,'ganesh', 62000, Null,'begumpet','SBI00178',10465,'None',30000,2022,'april','
    cur.execute(insert_script)
    cur.execute('select * from employee1')
    data = cur.fetchall()

    df1 = pd.DataFrame(data=data,columns=['id','name','salary','dept_id','branch','acc
    #print(df1)
    print(df1.isna().sum())
    print(df1.isna().mean())

    cur.execute('''SELECT avg(Transaction_hour),Transaction_day FROM employee1
    group by Transaction_day''')
    data = cur.fetchall()
    conn.commit()
```

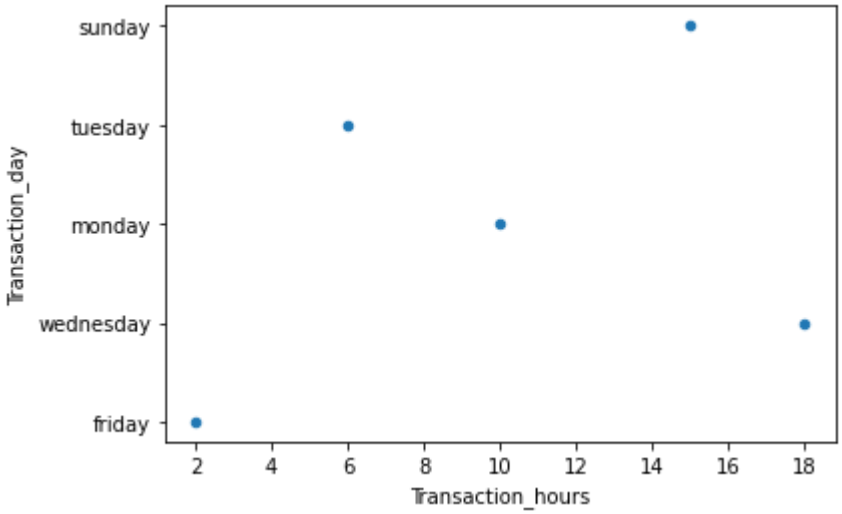
```
    df = pd.DataFrame(data=data,columns=['hour','day'])
    print(df)


except Exception as error:
    print(error)
finally:
    if cur is not None:
        cur.close()
    if conn is not None:
        conn.close()
```

```
id                    0
name                  0
salary                1
dept_id               1
branch                1
account_Number        0
Transaction_id        0
Transaction_status    0
Transaction_limit     0
Transaction_year      0
Transaction_month     0
Transaction_day       0
Transaction_hour      1
dtype: int64
id                    0.000000
name                  0.000000
salary                0.166667
dept_id               0.166667
branch                0.166667
account_Number        0.000000
Transaction_id        0.000000
Transaction_status    0.000000
Transaction_limit     0.000000
Transaction_year      0.000000
Transaction_month     0.000000
Transaction_day       0.000000
Transaction_hour      0.166667
dtype: float64
                  hour        day
0   2.0000000000000000     friday
1  18.0000000000000000  wednesday
2  10.0000000000000000     monday
3   6.0000000000000000    tuesday
4  15.0000000000000000     sunday
```

In [64]:
```
df = pd.DataFrame(data=data, columns=['Transaction_hours','Transaction_day'])
df.plot.scatter(x='Transaction_hours', y='Transaction_day')
```

Out[64]:
```
<AxesSubplot:xlabel='Transaction_hours', ylabel='Transaction_day'>
```



In [65]:
```
df1.tail(6)
```

Out[65]:

	id	name	salary	dept_id	branch	account_Number	Transaction_id	Transaction_status
0	101	James	12000.0	D1	None	SBI00123	10235	successful
1	102	suresh	25000.0	D1	secunderabad	SBI00133	10205	unsuccessful
2	103	raju	NaN	D2	boduppal	SBI00122	10265	successful
3	104	rani	65000.0	D3	ramanthapur	SBI00165	10005	successful
4	105	mahesh	35000.0	D6	amberpet	SBI00135	10665	unsuccessful
5	106	ganesh	62000.0	None	begumpet	SBI00178	10465	None

In [66]:
```
df
```

Out[66]:

	Transaction_hours	Transaction_day
0	2.0000000000000000	friday
1	18.0000000000000000	wednesday
2	10.0000000000000000	monday
3	6.0000000000000000	tuesday
4	15.0000000000000000	sunday

In [67]:
```
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6 entries, 0 to 5
Data columns (total 13 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   id                  6 non-null      int64  
 1   name                6 non-null      object 
 2   salary              5 non-null      float64
 3   dept_id             5 non-null      object 
 4   branch              5 non-null      object 
 5   account_Number      6 non-null      object 
 6   Transaction_id      6 non-null      int64  
 7   Transaction_status  6 non-null      object 
 8   Transaction_limit   6 non-null      int64  
 9   Transaction_year    6 non-null      int64  
 10  Transaction_month   6 non-null      object 
 11  Transaction_day     6 non-null      object 
 12  Transaction_hour    5 non-null      float64
dtypes: float64(2), int64(4), object(7)
memory usage: 752.0+ bytes
```

In [68]:
```
subset = ['salary', 'Transaction_hour']
df1.loc[:, subset] = df1.loc[:, subset].fillna(0)
```

In [69]:
```
df1.head()
```

Out[69]:

	id	name	salary	dept_id	branch	account_Number	Transaction_id	Transaction_status
0	101	James	12000.0	D1	None	SBI00123	10235	successful
1	102	suresh	25000.0	D1	secunderabad	SBI00133	10205	unsuccessful
2	103	raju	0.0	D2	boduppal	SBI00122	10265	successful
3	104	rani	65000.0	D3	ramanthapur	SBI00165	10005	successful
4	105	mahesh	35000.0	D6	amberpet	SBI00135	10665	unsuccessful

In [70]:
```
df1.fillna("NA",inplace=True)
```

In [71]:
```
df1.head()
```

Out[71]:

	id	name	salary	dept_id	branch	account_Number	Transaction_id	Transaction_status
0	101	James	12000.0	D1	NA	SBI00123	10235	successful
1	102	suresh	25000.0	D1	secunderabad	SBI00133	10205	unsuccessful
2	103	raju	0.0	D2	boduppal	SBI00122	10265	successful
3	104	rani	65000.0	D3	ramanthapur	SBI00165	10005	successful
4	105	mahesh	35000.0	D6	amberpet	SBI00135	10665	unsuccessful

In [78]:
```
df1['salary'].replace(np.NAN,df1['salary'].mean()).head(6)
```

Out[78]:
```
0    12000.0
1    25000.0
2    39800.0
3    65000.0
4    35000.0
5    62000.0
Name: salary, dtype: float64
```