

```

import pandas as pd
import numpy as np

courses = pd.read_csv('courses.csv')
students = pd.read_csv('students.csv')
nov = pd.read_csv('reg-month1.csv')
dec = pd.read_csv('reg-month2.csv')

matches = pd.read_csv('matches.csv')
delivery = pd.read_csv('deliveries.csv')

```

courses

	course_id	course_name	price
0	1	python	2499
1	2	sql	3499
2	3	data analysis	4999
3	4	machine learning	9999
4	5	tableau	2499
5	6	power bi	1899
6	7	ms sxcel	1599
7	8	pandas	1099
8	9	plotly	699
9	10	pyspark	2499
10	11	Numpy	699
11	12	C++	1299

students

	student_id	name	partner
0	1	Kailash Harjo	23
1	2	Esha Butala	1
2	3	Parveen Bhalla	3
3	4	Marlo Dugal	14
4	5	Kusum Bahri	6
5	6	Lakshmi Contractor	10
6	7	Tarun Thaker	9
7	8	Radheshyam Dey	5
8	9	Nitika Chatterjee	4
9	10	Aayushman Sant	8
10	11	David Mukhopadhyay	20
11	12	Radha Dutt	19
12	13	Munni Varghese	24
13	14	Pranab Natarajan	22
14	15	Preet Sha	16
15	16	Elias Dodiya	25
16	17	Yasmin Palan	7
17	18	Fardeen Mahabir	13
18	19	Qabeel Raman	12
19	20	Hanuman Hegde	11

20	21	Seema Kota	15
21	22	Yash Sethi	21
22	23	Chhavi Lachman	18
23	24	Radhika Suri	17
24	25	Shashank D'Alia	2

nov

	student_id	course_id
0	23	1
1	15	5
2	18	6
3	23	4
4	16	9
5	18	1
6	1	1
7	7	8
8	22	3
9	15	1
10	19	4
11	1	6
12	7	10
13	11	7
14	13	3
15	24	4
16	21	1
17	16	5
18	23	3
19	17	7
20	23	6
21	25	1
22	19	2
23	25	10
24	3	3

dec

	student_id	course_id
0	3	5
1	16	7
2	12	10
3	12	1
4	14	9
5	7	7
6	7	2
7	16	3
8	17	10
9	11	8
10	14	6
11	12	5

12	12	7
13	18	8
14	1	10
15	1	9
16	2	5
17	7	6
18	22	5
19	22	6
20	23	9
21	23	5
22	14	4
23	14	1
24	11	10
25	42	9
26	50	8
27	38	1

matches

	id	season	city	date	
team1 \					
0	1	2017	Hyderabad	2017-04-05	Sunrisers Hyderabad
1	2	2017	Pune	2017-04-06	Mumbai Indians
2	3	2017	Rajkot	2017-04-07	Gujarat Lions
3	4	2017	Indore	2017-04-08	Rising Pune Supergiant
4	5	2017	Bangalore	2017-04-08	Royal Challengers Bangalore
..
631	632	2016	Raipur	2016-05-22	Delhi Daredevils
632	633	2016	Bangalore	2016-05-24	Gujarat Lions
633	634	2016	Delhi	2016-05-25	Sunrisers Hyderabad
634	635	2016	Delhi	2016-05-27	Gujarat Lions
635	636	2016	Bangalore	2016-05-29	Sunrisers Hyderabad

	team2	toss_winner
toss_decision \		
0	Royal Challengers Bangalore	Royal Challengers Bangalore
field		
1	Rising Pune Supergiant	Rising Pune Supergiant
field		
2	Kolkata Knight Riders	Kolkata Knight Riders

```

field
3          Kings XI Punjab          Kings XI Punjab
field
4          Delhi Daredevils  Royal Challengers Bangalore
bat
..          ...          ...
...
631 Royal Challengers Bangalore  Royal Challengers Bangalore
field
632 Royal Challengers Bangalore  Royal Challengers Bangalore
field
633          Kolkata Knight Riders          Kolkata Knight Riders
field
634          Sunrisers Hyderabad          Sunrisers Hyderabad
field
635 Royal Challengers Bangalore          Sunrisers Hyderabad
bat

      result  dl_applied          winner  win_by_runs  \
0      normal          0          Sunrisers Hyderabad          35
1      normal          0          Rising Pune Supergiant          0
2      normal          0          Kolkata Knight Riders          0
3      normal          0          Kings XI Punjab          0
4      normal          0  Royal Challengers Bangalore          15
..      ...          ...          ...          ...
631  normal          0  Royal Challengers Bangalore          0
632  normal          0  Royal Challengers Bangalore          0
633  normal          0          Sunrisers Hyderabad          22
634  normal          0          Sunrisers Hyderabad          0
635  normal          0          Sunrisers Hyderabad          8

      win_by_wickets  player_of_match  \
0          0          Yuvraj Singh
1          7          SPD Smith
2         10          CA Lynn
3          6          GJ Maxwell
4          0          KM Jadhav
..      ...          ...
631          6          V Kohli
632          4  AB de Villiers
633          0          MC Henriques
634          4          DA Warner
635          0          BCJ Cutting

          venue          umpire1
\
0          Rajiv Gandhi International Stadium, Uppal          AY Dandekar
1          Maharashtra Cricket Association Stadium          A Nand Kishore

```

2	Saurashtra Cricket Association Stadium	Nitin Menon
3	Holkar Cricket Stadium	AK Chaudhary
4	M Chinnaswamy Stadium	NaN
..
631	Shaheed Veer Narayan Singh International Stadium	A Nand Kishore
632	M Chinnaswamy Stadium	AK Chaudhary
633	Feroz Shah Kotla	M Erasmus
634	Feroz Shah Kotla	M Erasmus
635	M Chinnaswamy Stadium	HDPK Dharmasena

	umpire2	umpire3
0	NJ Llong	NaN
1	S Ravi	NaN
2	CK Nandan	NaN
3	C Shamshuddin	NaN
4	NaN	NaN
..
631	BNJ Oxenford	NaN
632	HDPK Dharmasena	NaN
633	C Shamshuddin	NaN
634	CK Nandan	NaN
635	BNJ Oxenford	NaN

[636 rows x 18 columns]

delivery

	match_id	inning	batting_team	
bowling_team \				
0	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore
1	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore
2	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore
3	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore
4	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore
...
...				
179073	11415	2	Chennai Super Kings	Mumbai

Indians					
179074	11415	2	Chennai Super Kings		Mumbai
Indians					
179075	11415	2	Chennai Super Kings		Mumbai
Indians					
179076	11415	2	Chennai Super Kings		Mumbai
Indians					
179077	11415	2	Chennai Super Kings		Mumbai
Indians					

	over	ball	batsman	non_striker	bowler
is_super_over	...	\			
0	1	1	DA Warner	S Dhawan	TS Mills
0	...				
1	1	2	DA Warner	S Dhawan	TS Mills
0	...				
2	1	3	DA Warner	S Dhawan	TS Mills
0	...				
3	1	4	DA Warner	S Dhawan	TS Mills
0	...				
4	1	5	DA Warner	S Dhawan	TS Mills
0	...				
...
...					
179073	20	2	RA Jadeja	SR Watson	SL Malinga
0	...				
179074	20	3	SR Watson	RA Jadeja	SL Malinga
0	...				
179075	20	4	SR Watson	RA Jadeja	SL Malinga
0	...				
179076	20	5	SN Thakur	RA Jadeja	SL Malinga
0	...				
179077	20	6	SN Thakur	RA Jadeja	SL Malinga
0	...				

	bye_runs	legbye_runs	noball_runs	penalty_runs	batsman_runs
\					
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	4
3	0	0	0	0	0
4	0	0	0	0	0
...
179073	0	0	0	0	1

179074	0	0	0	0	2
179075	0	0	0	0	1
179076	0	0	0	0	2
179077	0	0	0	0	0

	extra_runs	total_runs	player_dismissed	dismissal_kind	
fielder					
0	0	0	NaN	NaN	
NaN					
1	0	0	NaN	NaN	
NaN					
2	0	4	NaN	NaN	
NaN					
3	0	0	NaN	NaN	
NaN					
4	2	2	NaN	NaN	
NaN					
...
...					
179073	0	1	NaN	NaN	
NaN					
179074	0	2	NaN	NaN	
NaN					
179075	0	1	SR Watson	run out	KH
Pandya					
179076	0	2	NaN	NaN	
NaN					
179077	0	0	SN Thakur	lbw	
NaN					

[179078 rows x 21 columns]

some important functions

```
# pd.concat
# df.concat
# ignore_index
# df.append
# mullitindex -> fetch using iloc
# concat dataframes horizontally

# pd.concat    => tow or more file ko concate kar dega
pd.concat([nov,dec])
# abhi jo dataframe hume mila hai usko observe karo wo index ko retain
kar lya hai matlab ki purane ke according hi indexing hai suru zero se
```

hua but

phir bich me zero aa gya agar aap isse bachna chahate ho to ek parameter pass kar sakte ho 'ignore_index=True' jab jarurat ho kar lo

	student_id	course_id
0	23	1
1	15	5
2	18	6
3	23	4
4	16	9
5	18	1
6	1	1
7	7	8
8	22	3
9	15	1
10	19	4
11	1	6
12	7	10
13	11	7
14	13	3
15	24	4
16	21	1
17	16	5
18	23	3
19	17	7
20	23	6
21	25	1
22	19	2
23	25	10
24	3	3
0	3	5
1	16	7
2	12	10
3	12	1
4	14	9
5	7	7
6	7	2
7	16	3
8	17	10
9	11	8
10	14	6
11	12	5
12	12	7
13	18	8
14	1	10
15	1	9
16	2	5
17	7	6
18	22	5
19	22	6

20	23	9
21	23	5
22	14	4
23	14	1
24	11	10
25	42	9
26	50	8
27	38	1

```
pd.concat([nov,dec]).shape
```

```
(53, 2)
```

```
regs = pd.concat([nov,dec],ignore_index=True)
regs
```

	student_id	course_id
0	23	1
1	15	5
2	18	6
3	23	4
4	16	9
5	18	1
6	1	1
7	7	8
8	22	3
9	15	1
10	19	4
11	1	6
12	7	10
13	11	7
14	13	3
15	24	4
16	21	1
17	16	5
18	23	3
19	17	7
20	23	6
21	25	1
22	19	2
23	25	10
24	3	3
25	3	5
26	16	7
27	12	10
28	12	1
29	14	9
30	7	7
31	7	2
32	16	3

33	17	10
34	11	8
35	14	6
36	12	5
37	12	7
38	18	8
39	1	10
40	1	9
41	2	5
42	7	6
43	22	5
44	22	6
45	23	9
46	23	5
47	14	4
48	14	1
49	11	10
50	42	9
51	50	8
52	38	1

multi-index dataframe

what if aap purane bale index ko rakhte hue new index banana chahate ho suppose ki hum sare november ke data ko nov and december ke data ko dec index

provide karana chahate hain

multi = pd.concat([nov,dec],keys=['Nov','Dec']) # keys ke andar list me new index pass kar sakte ho and observe the output aur aache se samajh aa

multi # jayega dhayan rahe ki jitna different dataframe ko concate karoge utna key bhi pass karna hai

	student_id	course_id
Nov 0	23	1
1	15	5
2	18	6
3	23	4
4	16	9
5	18	1
6	1	1
7	7	8
8	22	3
9	15	1
10	19	4
11	1	6
12	7	10
13	11	7

	14	13	3
	15	24	4
	16	21	1
	17	16	5
	18	23	3
	19	17	7
	20	23	6
	21	25	1
	22	19	2
	23	25	10
	24	3	3
Dec	0	3	5
	1	16	7
	2	12	10
	3	12	1
	4	14	9
	5	7	7
	6	7	2
	7	16	3
	8	17	10
	9	11	8
	10	14	6
	11	12	5
	12	12	7
	13	18	8
	14	1	10
	15	1	9
	16	2	5
	17	7	6
	18	22	5
	19	22	6
	20	23	9
	21	23	5
	22	14	4
	23	14	1
	24	11	10
	25	42	9
	26	50	8
	27	38	1

suppose abhi nov ke sare ko data ko extract karna hai
multi.loc['Nov'] # november ka sara data mil jayega

	student_id	course_id
0	23	1
1	15	5
2	18	6
3	23	4
4	16	9
5	18	1

6	1	1
7	7	8
8	22	3
9	15	1
10	19	4
11	1	6
12	7	10
13	11	7
14	13	3
15	24	4
16	21	1
17	16	5
18	23	3
19	17	7
20	23	6
21	25	1
22	19	2
23	25	10
24	3	3

```
multi.loc['Dec']
```

	student_id	course_id
0	3	5
1	16	7
2	12	10
3	12	1
4	14	9
5	7	7
6	7	2
7	16	3
8	17	10
9	11	8
10	14	6
11	12	5
12	12	7
13	18	8
14	1	10
15	1	9
16	2	5
17	7	6
18	22	5
19	22	6
20	23	9
21	23	5
22	14	4
23	14	1
24	11	10
25	42	9

26	50	8
27	38	1

```
# what if mujhe nov ke kisi particular row ko extract karna hai
multi.loc(['Nov',2]) # jo required hai usse ek tuple me pass kar do
-- november ke 2nd indexed ka student ka data mil jayega
```

```
student_id    18
course_id      6
Name: (Nov, 2), dtype: int64
```

```
multi.loc(['Dec',5]) # december ke 7th indexed ka student ka data
mil jayega
```

```
student_id      7
course_id        7
Name: (Dec, 5), dtype: int64
```

horizontly stacking

```
pd.concat([nov,dec],axis=1) # jaruri nahi ki jab aap horizontally
concat kar rahe ho to sare data frame me no. of column same ho
different ho sakte
# but make sure ki jab aap vertically concate kar rahe ho to sare
dataframe me no of column same ho
```

	student_id	course_id	student_id	course_id
0	23.0	1.0	3	5
1	15.0	5.0	16	7
2	18.0	6.0	12	10
3	23.0	4.0	12	1
4	16.0	9.0	14	9
5	18.0	1.0	7	7
6	1.0	1.0	7	2
7	7.0	8.0	16	3
8	22.0	3.0	17	10
9	15.0	1.0	11	8
10	19.0	4.0	14	6
11	1.0	6.0	12	5
12	7.0	10.0	12	7
13	11.0	7.0	18	8
14	13.0	3.0	1	10
15	24.0	4.0	1	9
16	21.0	1.0	2	5
17	16.0	5.0	7	6
18	23.0	3.0	22	5
19	17.0	7.0	22	6
20	23.0	6.0	23	9
21	25.0	1.0	23	5
22	19.0	2.0	14	4

23	25.0	10.0	14	1
24	3.0	3.0	11	10
25	NaN	NaN	42	9
26	NaN	NaN	50	8
27	NaN	NaN	38	1

```
pd.concat([nov,dec],axis=1).shape    # shape me 28 islye aa raha hai
qki shape jo data frame bada hai uske according ho jata hai and chote
bale
```

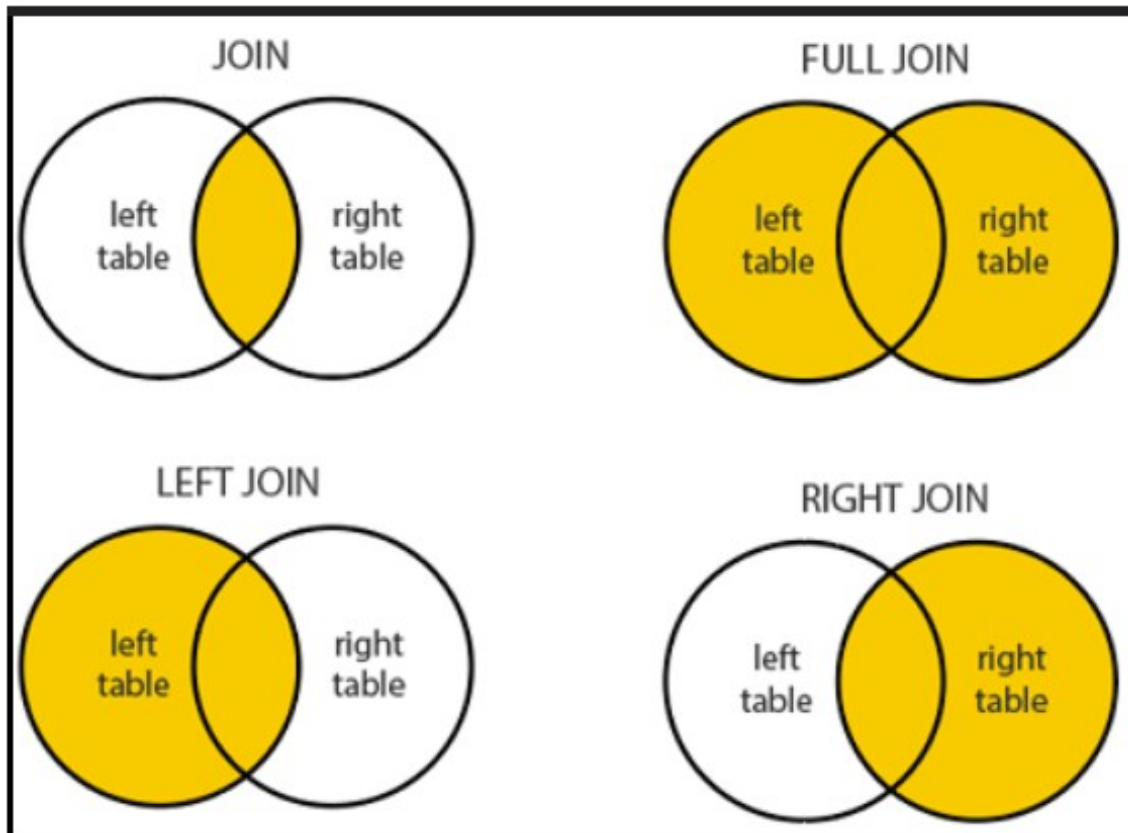
```
# me nan value fill ho jata hai
```

```
(28, 4)
```

Merging

- similar as sql

joins



```
# inner join
students
```

	student_id	name	partner
0	1	Kailash Harjo	23
1	2	Esha Butala	1

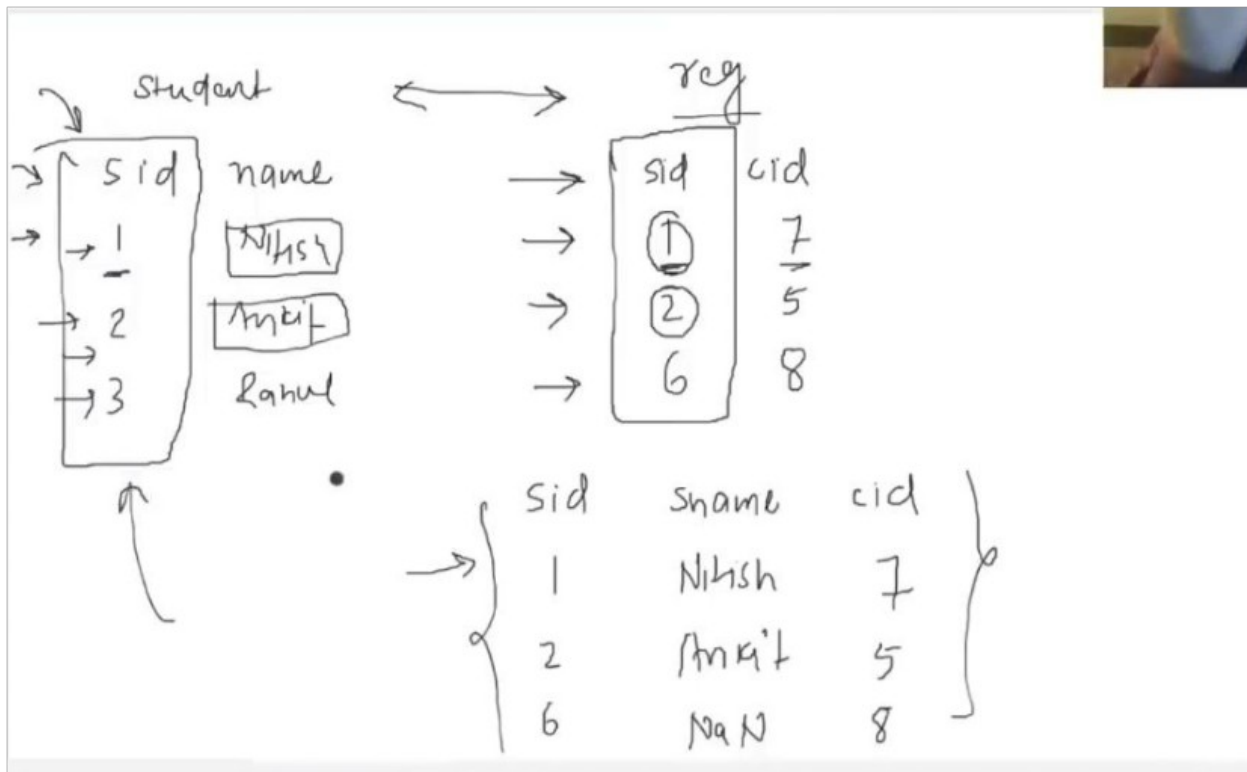
2	3	Parveen Bhalla	3
3	4	Marlo Dugal	14
4	5	Kusum Bahri	6
5	6	Lakshmi Contractor	10
6	7	Tarun Thaker	9
7	8	Radheshyam Dey	5
8	9	Nitika Chatterjee	4
9	10	Aayushman Sant	8
10	11	David Mukhopadhyay	20
11	12	Radha Dutt	19
12	13	Munni Varghese	24
13	14	Pranab Natarajan	22
14	15	Preet Sha	16
15	16	Elias Dodiya	25
16	17	Yasmin Palan	7
17	18	Fardeen Mahabir	13
18	19	Qabeel Raman	12
19	20	Hanuman Hegde	11
20	21	Seema Kota	15
21	22	Yash Sethi	21
22	23	Chhavi Lachman	18
23	24	Radhika Suri	17
24	25	Shashank D'Alia	2

regs *# suppose mujhe iss dataframe me upar jo students dataframe hai usse name iss dataframe me add karna hai*

	student_id	course_id
0	23	1
1	15	5
2	18	6
3	23	4
4	16	9
5	18	1
6	1	1
7	7	8
8	22	3
9	15	1
10	19	4
11	1	6
12	7	10
13	11	7
14	13	3
15	24	4
16	21	1
17	16	5
18	23	3
19	17	7
20	23	6
21	25	1

22	19	2
23	25	10
24	3	3
25	3	5
26	16	7
27	12	10
28	12	1
29	14	9
30	7	7
31	7	2
32	16	3
33	17	10
34	11	8
35	14	6
36	12	5
37	12	7
38	18	8
39	1	10
40	1	9
41	2	5
42	7	6
43	22	5
44	22	6
45	23	9
46	23	5
47	14	4
48	14	1
49	11	10
50	42	9
51	50	8
52	38	1

in the both df student_id column are common so on the basis of that we can add



pahle to ek common column hona chahiye dono me jin do ko join karna hai

inner join jab karenge to sirf common student_id bala data hi dikhega

inner join

students.merge(regs , how='inner', on='student_id') # jo ki pahle df ka name likhe hain (i.e students) wo left table and jo ki baad me likhe hain usse

right table bola jata hai and how me bataya jata hai ki aap kon sa join perform kar rahe ho and

on para meter me batate hain ki kon se column ke basis pe join kar rahe hain

	student_id	name	partner	course_id
0	1	Kailash Harjo	23	1
1	1	Kailash Harjo	23	6
2	1	Kailash Harjo	23	10
3	1	Kailash Harjo	23	9
4	2	Esha Butala	1	5
5	3	Parveen Bhalla	3	3
6	3	Parveen Bhalla	3	5
7	7	Tarun Thaker	9	8
8	7	Tarun Thaker	9	10
9	7	Tarun Thaker	9	7

10	7	Tarun Thaker	9	2
11	7	Tarun Thaker	9	6
12	11	David Mukhopadhyay	20	7
13	11	David Mukhopadhyay	20	8
14	11	David Mukhopadhyay	20	10
15	12	Radha Dutt	19	10
16	12	Radha Dutt	19	1
17	12	Radha Dutt	19	5
18	12	Radha Dutt	19	7
19	13	Munni Varghese	24	3
20	14	Pranab Natarajan	22	9
21	14	Pranab Natarajan	22	6
22	14	Pranab Natarajan	22	4
23	14	Pranab Natarajan	22	1
24	15	Preet Sha	16	5
25	15	Preet Sha	16	1
26	16	Elias Dodiya	25	9
27	16	Elias Dodiya	25	5
28	16	Elias Dodiya	25	7
29	16	Elias Dodiya	25	3
30	17	Yasmin Palan	7	7
31	17	Yasmin Palan	7	10
32	18	Fardeen Mahabir	13	6
33	18	Fardeen Mahabir	13	1
34	18	Fardeen Mahabir	13	8
35	19	Qabeel Raman	12	4
36	19	Qabeel Raman	12	2
37	21	Seema Kota	15	1
38	22	Yash Sethi	21	3
39	22	Yash Sethi	21	5
40	22	Yash Sethi	21	6
41	23	Chhavi Lachman	18	1
42	23	Chhavi Lachman	18	4
43	23	Chhavi Lachman	18	3
44	23	Chhavi Lachman	18	6
45	23	Chhavi Lachman	18	9
46	23	Chhavi Lachman	18	5
47	24	Radhika Suri	17	4
48	25	Shashank D'Alia	2	1
49	25	Shashank D'Alia	2	10

left join => not only common chizo ko print karte ho balki wo sare data ko print karte ho jo ki left bale table me hai but right bale table me nahi hai

```
courses.merge(regs,how='left',on='course_id')
```

	course_id	course_name	price	student_id
0	1	python	2499	23.0
1	1	python	2499	18.0

2	1	python	2499	1.0
3	1	python	2499	15.0
4	1	python	2499	21.0
5	1	python	2499	25.0
6	1	python	2499	12.0
7	1	python	2499	14.0
8	1	python	2499	38.0
9	2	sql	3499	19.0
10	2	sql	3499	7.0
11	3	data analysis	4999	22.0
12	3	data analysis	4999	13.0
13	3	data analysis	4999	23.0
14	3	data analysis	4999	3.0
15	3	data analysis	4999	16.0
16	4	machine learning	9999	23.0
17	4	machine learning	9999	19.0
18	4	machine learning	9999	24.0
19	4	machine learning	9999	14.0
20	5	tableau	2499	15.0
21	5	tableau	2499	16.0
22	5	tableau	2499	3.0
23	5	tableau	2499	12.0
24	5	tableau	2499	2.0
25	5	tableau	2499	22.0
26	5	tableau	2499	23.0
27	6	power bi	1899	18.0
28	6	power bi	1899	1.0
29	6	power bi	1899	23.0
30	6	power bi	1899	14.0
31	6	power bi	1899	7.0
32	6	power bi	1899	22.0
33	7	ms sxcel	1599	11.0
34	7	ms sxcel	1599	17.0
35	7	ms sxcel	1599	16.0
36	7	ms sxcel	1599	7.0
37	7	ms sxcel	1599	12.0
38	8	pandas	1099	7.0
39	8	pandas	1099	11.0
40	8	pandas	1099	18.0
41	8	pandas	1099	50.0
42	9	plotly	699	16.0
43	9	plotly	699	14.0
44	9	plotly	699	1.0
45	9	plotly	699	23.0
46	9	plotly	699	42.0
47	10	pyspark	2499	7.0
48	10	pyspark	2499	25.0
49	10	pyspark	2499	12.0
50	10	pyspark	2499	17.0

51	10	pyspark	2499	1.0
52	10	pyspark	2499	11.0
53	11	Numpy	699	NaN
54	12	C++	1299	NaN

right join => common as well right ka data

```
temp_df = pd.DataFrame({
    'student_id': [26, 27, 28],
    'name': ['Nitish', 'Ankit', 'Rahul'],
    'partner': [28, 26, 17]
})
```

```
students = pd.concat([students, temp_df], ignore_index=True)
students
```

	student_id	name	partner
0	1	Kailash Harjo	23
1	2	Esha Butala	1
2	3	Parveen Bhalla	3
3	4	Marlo Dugal	14
4	5	Kusum Bahri	6
5	6	Lakshmi Contractor	10
6	7	Tarun Thaker	9
7	8	Radheshyam Dey	5
8	9	Nitika Chatterjee	4
9	10	Aayushman Sant	8
10	11	David Mukhopadhyay	20
11	12	Radha Dutt	19
12	13	Munni Varghese	24
13	14	Pranab Natarajan	22
14	15	Preet Sha	16
15	16	Elias Dodiya	25
16	17	Yasmin Palan	7
17	18	Fardeen Mahabir	13
18	19	Qabeel Raman	12
19	20	Hanuman Hegde	11
20	21	Seema Kota	15
21	22	Yash Sethi	21
22	23	Chhavi Lachman	18
23	24	Radhika Suri	17
24	25	Shashank D'Alia	2
25	26	Nitish	28
26	27	Ankit	26
27	28	Rahul	17

students.tail() # ye jo niche ke 3 bachhe hain wo kisi bhi course me enroll nahi hua hai

	student_id	name	partner
23	24	Radhika Suri	17
24	25	Shashank D'Alia	2
25	26	Nitish	28
26	27	Ankit	26
27	28	Rahul	17

```
students.merge(regs,how='right',on='student_id')
```

	student_id	name	partner	course_id
0	23	Chhavi Lachman	18.0	1
1	15	Preet Sha	16.0	5
2	18	Fardeen Mahabir	13.0	6
3	23	Chhavi Lachman	18.0	4
4	16	Elias Dodiya	25.0	9
5	18	Fardeen Mahabir	13.0	1
6	1	Kailash Harjo	23.0	1
7	7	Tarun Thaker	9.0	8
8	22	Yash Sethi	21.0	3
9	15	Preet Sha	16.0	1
10	19	Qabeel Raman	12.0	4
11	1	Kailash Harjo	23.0	6
12	7	Tarun Thaker	9.0	10
13	11	David Mukhopadhyay	20.0	7
14	13	Munni Varghese	24.0	3
15	24	Radhika Suri	17.0	4
16	21	Seema Kota	15.0	1
17	16	Elias Dodiya	25.0	5
18	23	Chhavi Lachman	18.0	3
19	17	Yasmin Palan	7.0	7
20	23	Chhavi Lachman	18.0	6
21	25	Shashank D'Alia	2.0	1
22	19	Qabeel Raman	12.0	2
23	25	Shashank D'Alia	2.0	10
24	3	Parveen Bhalla	3.0	3
25	3	Parveen Bhalla	3.0	5
26	16	Elias Dodiya	25.0	7
27	12	Radha Dutt	19.0	10
28	12	Radha Dutt	19.0	1
29	14	Pranab Natarajan	22.0	9
30	7	Tarun Thaker	9.0	7
31	7	Tarun Thaker	9.0	2
32	16	Elias Dodiya	25.0	3
33	17	Yasmin Palan	7.0	10
34	11	David Mukhopadhyay	20.0	8
35	14	Pranab Natarajan	22.0	6
36	12	Radha Dutt	19.0	5
37	12	Radha Dutt	19.0	7
38	18	Fardeen Mahabir	13.0	8
39	1	Kailash Harjo	23.0	10

40	1	Kailash Harjo	23.0	9
41	2	Esha Butala	1.0	5
42	7	Tarun Thaker	9.0	6
43	22	Yash Sethi	21.0	5
44	22	Yash Sethi	21.0	6
45	23	Chhavi Lachman	18.0	9
46	23	Chhavi Lachman	18.0	5
47	14	Pranab Natarajan	22.0	4
48	14	Pranab Natarajan	22.0	1
49	11	David Mukhopadhyay	20.0	10
50	42	NaN	NaN	9
51	50	NaN	NaN	8
52	38	NaN	NaN	1

outer join => sare ka sare data dikhayenge jo left hai wo bhi jo right hai wo bhi and jo common hai wo bhi

```
students.merge(regs,how='outer',on='student_id')
```

	student_id	name	partner	course_id
0	1	Kailash Harjo	23.0	1.0
1	1	Kailash Harjo	23.0	6.0
2	1	Kailash Harjo	23.0	10.0
3	1	Kailash Harjo	23.0	9.0
4	2	Esha Butala	1.0	5.0
..
58	27	Ankit	26.0	NaN
59	28	Rahul	17.0	NaN
60	38	NaN	NaN	1.0
61	42	NaN	NaN	9.0
62	50	NaN	NaN	8.0

[63 rows x 4 columns]

1. find total revenue generated by the company

```
regs.merge(courses,how='inner',on='course_id')
```

	student_id	course_id	course_name	price
0	23	1	python	2499
1	15	5	tableau	2499
2	18	6	power bi	1899
3	23	4	machine learning	9999
4	16	9	plotly	699
5	18	1	python	2499
6	1	1	python	2499
7	7	8	pandas	1099
8	22	3	data analysis	4999
9	15	1	python	2499
10	19	4	machine learning	9999
11	1	6	power bi	1899

12	7	10	pyspark	2499
13	11	7	ms sxccl	1599
14	13	3	data analysis	4999
15	24	4	machine learning	9999
16	21	1	python	2499
17	16	5	tableau	2499
18	23	3	data analysis	4999
19	17	7	ms sxccl	1599
20	23	6	power bi	1899
21	25	1	python	2499
22	19	2	sql	3499
23	25	10	pyspark	2499
24	3	3	data analysis	4999
25	3	5	tableau	2499
26	16	7	ms sxccl	1599
27	12	10	pyspark	2499
28	12	1	python	2499
29	14	9	plotly	699
30	7	7	ms sxccl	1599
31	7	2	sql	3499
32	16	3	data analysis	4999
33	17	10	pyspark	2499
34	11	8	pandas	1099
35	14	6	power bi	1899
36	12	5	tableau	2499
37	12	7	ms sxccl	1599
38	18	8	pandas	1099
39	1	10	pyspark	2499
40	1	9	plotly	699
41	2	5	tableau	2499
42	7	6	power bi	1899
43	22	5	tableau	2499
44	22	6	power bi	1899
45	23	9	plotly	699
46	23	5	tableau	2499
47	14	4	machine learning	9999
48	14	1	python	2499
49	11	10	pyspark	2499
50	42	9	plotly	699
51	50	8	pandas	1099
52	38	1	python	2499

```
regs.merge(courses,how='inner',on='course_id')['price']
```

0	2499
1	2499
2	1899
3	9999
4	699
5	2499

6	2499
7	1099
8	4999
9	2499
10	9999
11	1899
12	2499
13	1599
14	4999
15	9999
16	2499
17	2499
18	4999
19	1599
20	1899
21	2499
22	3499
23	2499
24	4999
25	2499
26	1599
27	2499
28	2499
29	699
30	1599
31	3499
32	4999
33	2499
34	1099
35	1899
36	2499
37	1599
38	1099
39	2499
40	699
41	2499
42	1899
43	2499
44	1899
45	699
46	2499
47	9999
48	2499
49	2499
50	699
51	1099
52	2499

Name: price, dtype: int64


```
total=regs.merge(courses,how='inner',on='course_id')['price'].sum()  
total
```

154247

find month by month revenue

nov

	student_id	course_id
0	23	1
1	15	5
2	18	6
3	23	4
4	16	9
5	18	1
6	1	1
7	7	8
8	22	3
9	15	1
10	19	4
11	1	6
12	7	10
13	11	7
14	13	3
15	24	4
16	21	1
17	16	5
18	23	3
19	17	7
20	23	6
21	25	1
22	19	2
23	25	10
24	3	3

dec

	student_id	course_id
0	3	5
1	16	7
2	12	10
3	12	1
4	14	9
5	7	7
6	7	2
7	16	3
8	17	10
9	11	8
10	14	6
11	12	5

12	12	7
13	18	8
14	1	10
15	1	9
16	2	5
17	7	6
18	22	5
19	22	6
20	23	9
21	23	5
22	14	4
23	14	1
24	11	10
25	42	9
26	50	8
27	38	1

```
pd.concat([nov,dec],keys=['Nov','Dec'])
```

		student_id	course_id
Nov	0	23	1
	1	15	5
	2	18	6
	3	23	4
	4	16	9
	5	18	1
	6	1	1
	7	7	8
	8	22	3
	9	15	1
	10	19	4
	11	1	6
	12	7	10
	13	11	7
	14	13	3
	15	24	4
	16	21	1
	17	16	5
	18	23	3
	19	17	7
	20	23	6
	21	25	1
	22	19	2
	23	25	10
	24	3	3
Dec	0	3	5
	1	16	7
	2	12	10
	3	12	1
	4	14	9

5	7	7
6	7	2
7	16	3
8	17	10
9	11	8
10	14	6
11	12	5
12	12	7
13	18	8
14	1	10
15	1	9
16	2	5
17	7	6
18	22	5
19	22	6
20	23	9
21	23	5
22	14	4
23	14	1
24	11	10
25	42	9
26	50	8
27	38	1

```
temp_df=pd.concat([nov,dec],keys=['Nov','Dec']).reset_index() # ek
new index ban jayega
temp_df
```

	level_0	level_1	student_id	course_id
0	Nov	0	23	1
1	Nov	1	15	5
2	Nov	2	18	6
3	Nov	3	23	4
4	Nov	4	16	9
5	Nov	5	18	1
6	Nov	6	1	1
7	Nov	7	7	8
8	Nov	8	22	3
9	Nov	9	15	1
10	Nov	10	19	4
11	Nov	11	1	6
12	Nov	12	7	10
13	Nov	13	11	7
14	Nov	14	13	3
15	Nov	15	24	4
16	Nov	16	21	1
17	Nov	17	16	5
18	Nov	18	23	3
19	Nov	19	17	7
20	Nov	20	23	6

21	Nov	21	25	1
22	Nov	22	19	2
23	Nov	23	25	10
24	Nov	24	3	3
25	Dec	0	3	5
26	Dec	1	16	7
27	Dec	2	12	10
28	Dec	3	12	1
29	Dec	4	14	9
30	Dec	5	7	7
31	Dec	6	7	2
32	Dec	7	16	3
33	Dec	8	17	10
34	Dec	9	11	8
35	Dec	10	14	6
36	Dec	11	12	5
37	Dec	12	12	7
38	Dec	13	18	8
39	Dec	14	1	10
40	Dec	15	1	9
41	Dec	16	2	5
42	Dec	17	7	6
43	Dec	18	22	5
44	Dec	19	22	6
45	Dec	20	23	9
46	Dec	21	23	5
47	Dec	22	14	4
48	Dec	23	14	1
49	Dec	24	11	10
50	Dec	25	42	9
51	Dec	26	50	8
52	Dec	27	38	1

`temp_df.merge(courses,on='course_id')` *# by default inner join hota hai*

	level_0	level_1	student_id	course_id	course_name	price
0	Nov	0	23	1	python	2499
1	Nov	1	15	5	tableau	2499
2	Nov	2	18	6	power bi	1899
3	Nov	3	23	4	machine learning	9999
4	Nov	4	16	9	plotly	699
5	Nov	5	18	1	python	2499
6	Nov	6	1	1	python	2499
7	Nov	7	7	8	pandas	1099
8	Nov	8	22	3	data analysis	4999
9	Nov	9	15	1	python	2499
10	Nov	10	19	4	machine learning	9999
11	Nov	11	1	6	power bi	1899
12	Nov	12	7	10	pyspark	2499

13	Nov	13	11	7	ms excel	1599
14	Nov	14	13	3	data analysis	4999
15	Nov	15	24	4	machine learning	9999
16	Nov	16	21	1	python	2499
17	Nov	17	16	5	tableau	2499
18	Nov	18	23	3	data analysis	4999
19	Nov	19	17	7	ms excel	1599
20	Nov	20	23	6	power bi	1899
21	Nov	21	25	1	python	2499
22	Nov	22	19	2	sql	3499
23	Nov	23	25	10	pyspark	2499
24	Nov	24	3	3	data analysis	4999
25	Dec	0	3	5	tableau	2499
26	Dec	1	16	7	ms excel	1599
27	Dec	2	12	10	pyspark	2499
28	Dec	3	12	1	python	2499
29	Dec	4	14	9	plotly	699
30	Dec	5	7	7	ms excel	1599
31	Dec	6	7	2	sql	3499
32	Dec	7	16	3	data analysis	4999
33	Dec	8	17	10	pyspark	2499
34	Dec	9	11	8	pandas	1099
35	Dec	10	14	6	power bi	1899
36	Dec	11	12	5	tableau	2499
37	Dec	12	12	7	ms excel	1599
38	Dec	13	18	8	pandas	1099
39	Dec	14	1	10	pyspark	2499
40	Dec	15	1	9	plotly	699
41	Dec	16	2	5	tableau	2499
42	Dec	17	7	6	power bi	1899
43	Dec	18	22	5	tableau	2499
44	Dec	19	22	6	power bi	1899
45	Dec	20	23	9	plotly	699
46	Dec	21	23	5	tableau	2499
47	Dec	22	14	4	machine learning	9999
48	Dec	23	14	1	python	2499
49	Dec	24	11	10	pyspark	2499
50	Dec	25	42	9	plotly	699
51	Dec	26	50	8	pandas	1099
52	Dec	27	38	1	python	2499

```
temp_df.merge(courses,on='course_id').groupby('level_0')
```

```
<pandas.core.groupby.generic.DataFrameGroupBy object at 0x000001E0F5CD0FE0>
```

```
temp_df.merge(courses,on='course_id').groupby('level_0')
['price'].sum()
```

```
level_0
Dec      65072
Nov      89175
Name: price, dtype: int64
```

```
temp_df.merge(courses,on='course_id').groupby('level_0')
['price'].sum().sum()
```

```
154247
```

```
# 3. Print the registration table
# cols -> name -> course -> price
regs.merge(students,on='student_id')
```

	student_id	course_id	name	partner
0	23	1	Chhavi Lachman	18
1	15	5	Preet Sha	16
2	18	6	Fardeen Mahabir	13
3	23	4	Chhavi Lachman	18
4	16	9	Elias Dodiya	25
5	18	1	Fardeen Mahabir	13
6	1	1	Kailash Harjo	23
7	7	8	Tarun Thaker	9
8	22	3	Yash Sethi	21
9	15	1	Preet Sha	16
10	19	4	Qabeel Raman	12
11	1	6	Kailash Harjo	23
12	7	10	Tarun Thaker	9
13	11	7	David Mukhopadhyay	20
14	13	3	Munni Varghese	24
15	24	4	Radhika Suri	17
16	21	1	Seema Kota	15
17	16	5	Elias Dodiya	25
18	23	3	Chhavi Lachman	18
19	17	7	Yasmin Palan	7
20	23	6	Chhavi Lachman	18
21	25	1	Shashank D'Alia	2
22	19	2	Qabeel Raman	12
23	25	10	Shashank D'Alia	2
24	3	3	Parveen Bhalla	3
25	3	5	Parveen Bhalla	3
26	16	7	Elias Dodiya	25
27	12	10	Radha Dutt	19
28	12	1	Radha Dutt	19
29	14	9	Pranab Natarajan	22
30	7	7	Tarun Thaker	9
31	7	2	Tarun Thaker	9
32	16	3	Elias Dodiya	25
33	17	10	Yasmin Palan	7
34	11	8	David Mukhopadhyay	20

35	14	6	Pranab Natarajan	22
36	12	5	Radha Dutt	19
37	12	7	Radha Dutt	19
38	18	8	Fardeen Mahabir	13
39	1	10	Kailash Harjo	23
40	1	9	Kailash Harjo	23
41	2	5	Esha Butala	1
42	7	6	Tarun Thaker	9
43	22	5	Yash Sethi	21
44	22	6	Yash Sethi	21
45	23	9	Chhavi Lachman	18
46	23	5	Chhavi Lachman	18
47	14	4	Pranab Natarajan	22
48	14	1	Pranab Natarajan	22
49	11	10	David Mukhopadhyay	20

```
regs.merge(students,on='student_id').merge(courses,on='course_id')
```

	student_id	course_id	name	partner	
course_name \					
0	23	1	Chhavi Lachman	18	
python					
1	15	5	Preet Sha	16	
tableau					
2	18	6	Fardeen Mahabir	13	power
bi					
3	23	4	Chhavi Lachman	18	machine
learning					
4	16	9	Elias Dodiya	25	
plotly					
5	18	1	Fardeen Mahabir	13	
python					
6	1	1	Kailash Harjo	23	
python					
7	7	8	Tarun Thaker	9	
pandas					
8	22	3	Yash Sethi	21	data
analysis					
9	15	1	Preet Sha	16	
python					
10	19	4	Qabeel Raman	12	machine
learning					
11	1	6	Kailash Harjo	23	power
bi					
12	7	10	Tarun Thaker	9	
pyspark					
13	11	7	David Mukhopadhyay	20	ms
sxcel					
14	13	3	Munni Varghese	24	data
analysis					

15	24	4	Radhika Suri	17	machine
learning					
16	21	1	Seema Kota	15	
python					
17	16	5	Elias Dodiya	25	
tableau					
18	23	3	Chhavi Lachman	18	data
analysis					
19	17	7	Yasmin Palan	7	ms
sxcel					
20	23	6	Chhavi Lachman	18	power
bi					
21	25	1	Shashank D'Alia	2	
python					
22	19	2	Qabeel Raman	12	
sql					
23	25	10	Shashank D'Alia	2	
pyspark					
24	3	3	Parveen Bhalla	3	data
analysis					
25	3	5	Parveen Bhalla	3	
tableau					
26	16	7	Elias Dodiya	25	ms
sxcel					
27	12	10	Radha Dutt	19	
pyspark					
28	12	1	Radha Dutt	19	
python					
29	14	9	Pranab Natarajan	22	
plotly					
30	7	7	Tarun Thaker	9	ms
sxcel					
31	7	2	Tarun Thaker	9	
sql					
32	16	3	Elias Dodiya	25	data
analysis					
33	17	10	Yasmin Palan	7	
pyspark					
34	11	8	David Mukhopadhyay	20	
pandas					
35	14	6	Pranab Natarajan	22	power
bi					
36	12	5	Radha Dutt	19	
tableau					
37	12	7	Radha Dutt	19	ms
sxcel					
38	18	8	Fardeen Mahabir	13	
pandas					
39	1	10	Kailash Harjo	23	

pyspark					
40	1	9	Kailash Harjo	23	
plotly					
41	2	5	Esha Butala	1	
tableau					
42	7	6	Tarun Thaker	9	power
bi					
43	22	5	Yash Sethi	21	
tableau					
44	22	6	Yash Sethi	21	power
bi					
45	23	9	Chhavi Lachman	18	
plotly					
46	23	5	Chhavi Lachman	18	
tableau					
47	14	4	Pranab Natarajan	22	machine
learning					
48	14	1	Pranab Natarajan	22	
python					
49	11	10	David Mukhopadhyay	20	
pyspark					

	price
0	2499
1	2499
2	1899
3	9999
4	699
5	2499
6	2499
7	1099
8	4999
9	2499
10	9999
11	1899
12	2499
13	1599
14	4999
15	9999
16	2499
17	2499
18	4999
19	1599
20	1899
21	2499
22	3499
23	2499
24	4999
25	2499

26 1599
 27 2499
 28 2499
 29 699
 30 1599
 31 3499
 32 4999
 33 2499
 34 1099
 35 1899
 36 2499
 37 1599
 38 1099
 39 2499
 40 699
 41 2499
 42 1899
 43 2499
 44 1899
 45 699
 46 2499
 47 9999
 48 2499
 49 2499

```
regs.merge(students,on='student_id').merge(courses,on='course_id')
[['name','course_name','price']]
```

	name	course_name	price
0	Chhavi Lachman	python	2499
1	Preet Sha	tableau	2499
2	Fardeen Mahabir	power bi	1899
3	Chhavi Lachman	machine learning	9999
4	Elias Dodiya	plotly	699
5	Fardeen Mahabir	python	2499
6	Kailash Harjo	python	2499
7	Tarun Thaker	pandas	1099
8	Yash Sethi	data analysis	4999
9	Preet Sha	python	2499
10	Qabeel Raman	machine learning	9999
11	Kailash Harjo	power bi	1899
12	Tarun Thaker	pyspark	2499
13	David Mukhopadhyay	ms sxccl	1599
14	Munni Varghese	data analysis	4999
15	Radhika Suri	machine learning	9999
16	Seema Kota	python	2499
17	Elias Dodiya	tableau	2499
18	Chhavi Lachman	data analysis	4999
19	Yasmin Palan	ms sxccl	1599
20	Chhavi Lachman	power bi	1899

21	Shashank D'Alia	python	2499
22	Qabeel Raman	sql	3499
23	Shashank D'Alia	pyspark	2499
24	Parveen Bhalla	data analysis	4999
25	Parveen Bhalla	tableau	2499
26	Elias Dodiya	ms sxccl	1599
27	Radha Dutt	pyspark	2499
28	Radha Dutt	python	2499
29	Pranab Natarajan	plotly	699
30	Tarun Thaker	ms sxccl	1599
31	Tarun Thaker	sql	3499
32	Elias Dodiya	data analysis	4999
33	Yasmin Palan	pyspark	2499
34	David Mukhopadhyay	pandas	1099
35	Pranab Natarajan	power bi	1899
36	Radha Dutt	tableau	2499
37	Radha Dutt	ms sxccl	1599
38	Fardeen Mahabir	pandas	1099
39	Kailash Harjo	pyspark	2499
40	Kailash Harjo	plotly	699
41	Esha Butala	tableau	2499
42	Tarun Thaker	power bi	1899
43	Yash Sethi	tableau	2499
44	Yash Sethi	power bi	1899
45	Chhavi Lachman	plotly	699
46	Chhavi Lachman	tableau	2499
47	Pranab Natarajan	machine learning	9999
48	Pranab Natarajan	python	2499
49	David Mukhopadhyay	pyspark	2499

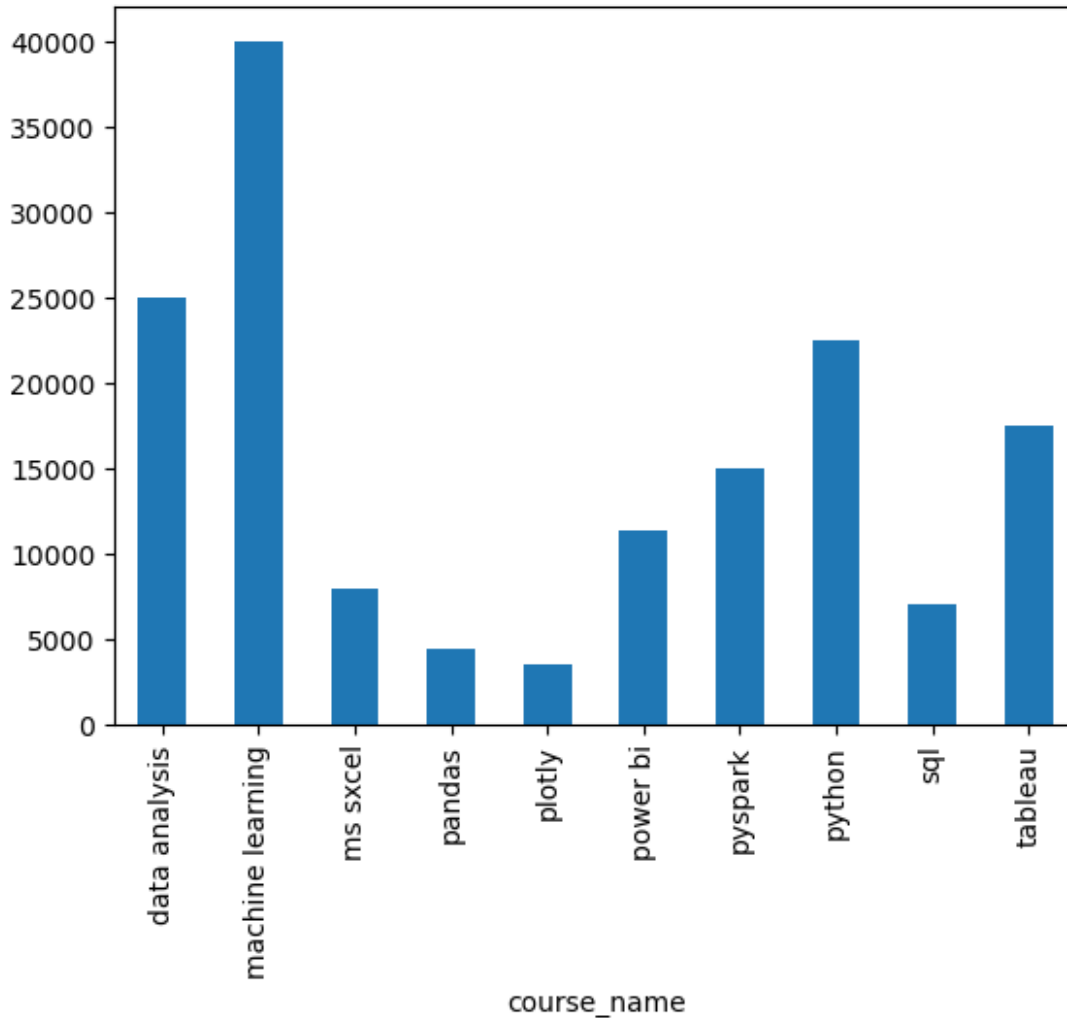
4. Plot bar chart for revenue/course

```
regs.merge(courses,on='course_id').groupby('course_name')
['price'].sum()
```

```
course_name
data analysis    24995
machine learning  39996
ms sxccl         7995
pandas           4396
plotly           3495
power bi         11394
pyspark          14994
python           22491
sql              6998
tableau          17493
Name: price, dtype: int64
```

```
regs.merge(courses,on='course_id').groupby('course_name')
['price'].sum().plot(kind='bar')
```

<Axes: xlabel='course_name'>



```
# 5. find students who enrolled in both the months
common_student_id=np.intersect1d(nov['student_id'],dec['student_id'])
#np.intersect1d -> intersection jaisa kaam karta hai
common_student_id
```

```
array([ 1,  3,  7, 11, 16, 17, 18, 22, 23], dtype=int64)
```

```
students['student_id'].isin(common_student_id)
```

```
0    True
1   False
2    True
3   False
```

```

4      False
5      False
6       True
7      False
8      False
9      False
10     True
11     False
12     False
13     False
14     False
15     True
16     True
17     True
18     False
19     False
20     False
21     True
22     True
23     False
24     False
25     False
26     False
27     False

```

Name: student_id, dtype: bool

```
students[students['student_id'].isin(common_student_id)]
```

	student_id	name	partner
0	1	Kailash Harjo	23
2	3	Parveen Bhalla	3
6	7	Tarun Thaker	9
10	11	David Mukhopadhyay	20
15	16	Elias Dodiya	25
16	17	Yasmin Palan	7
17	18	Fardeen Mahabir	13
21	22	Yash Sethi	21
22	23	Chhavi Lachman	18

6. find course that got no enrollment

courses['course_id']

regs['course_id']

*# hume aaisa data nikalne hain jo ki courses bale table me to hai but
regs bale table me nahi hai*

```
courses['course_id']
```

```

0      1
1      2
2      3
3      4

```

```
4      5
5      6
6      7
7      8
8      9
9     10
10     11
11     12
Name: course_id, dtype: int64
```

```
regs['course_id']
```

```
0      1
1      5
2      6
3      4
4      9
5      1
6      1
7      8
8      3
9      1
10     4
11     6
12    10
13     7
14     3
15     4
16     1
17     5
18     3
19     7
20     6
21     1
22     2
23    10
24     3
25     5
26     7
27    10
28     1
29     9
30     7
31     2
32     3
33    10
34     8
35     6
36     5
37     7
```

```

38      8
39     10
40      9
41      5
42      6
43      5
44      6
45      9
46      5
47      4
48      1
49     10
50      9
51      8
52      1
Name: course_id, dtype: int64

course_id_list=np.setdiff1d(courses['course_id'],regs['course_id'])
course_id_list
# set difference nikalta hai

array([11, 12], dtype=int64)

courses[courses['course_id'].isin(course_id_list)]

   course_id  course_name  price
10         11         Numpy   699
11         12          C++  1299

# 7. find students who did not enroll into any courses
student_id_list=np.setdiff1d(students['student_id'],regs['course_id'])

student_id_list

array([11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26,
27,
      28], dtype=int64)

students[students['student_id'].isin(student_id_list)]

   student_id  name  partner
10         11  David Mukhopadhyay    20
11         12    Radha Dutt    19
12         13  Munni Varghese    24
13         14  Pranab Natarajan    22
14         15    Preet Sha    16
15         16  Elias Dodiya    25
16         17  Yasmin Palan     7
17         18  Fardeen Mahabir    13
18         19    Qabeel Raman    12
19         20  Hanuman Hegde    11

```

20	21	Seema Kota	15
21	22	Yash Sethi	21
22	23	Chhavi Lachman	18
23	24	Radhika Suri	17
24	25	Shashank D'Alia	2
25	26	Nitish	28
26	27	Ankit	26
27	28	Rahul	17

8. Print student name -> partner name for all enrolled students

self join

`students.merge(students,how='inner',left_on='partner',right_on='student_id')`

	student_id_x	name_x	partner_x	student_id_y	\
0	1	Kailash Harjo	23	23	
1	2	Esha Butala	1	1	
2	3	Parveen Bhalla	3	3	
3	4	Marlo Dugal	14	14	
4	5	Kusum Bahri	6	6	
5	6	Lakshmi Contractor	10	10	
6	7	Tarun Thaker	9	9	
7	8	Radheshyam Dey	5	5	
8	9	Nitika Chatterjee	4	4	
9	10	Aayushman Sant	8	8	
10	11	David Mukhopadhyay	20	20	
11	12	Radha Dutt	19	19	
12	13	Munni Varghese	24	24	
13	14	Pranab Natarajan	22	22	
14	15	Preet Sha	16	16	
15	16	Elias Dodiya	25	25	
16	17	Yasmin Palan	7	7	
17	18	Fardeen Mahabir	13	13	
18	19	Qabeel Raman	12	12	
19	20	Hanuman Hegde	11	11	
20	21	Seema Kota	15	15	
21	22	Yash Sethi	21	21	
22	23	Chhavi Lachman	18	18	
23	24	Radhika Suri	17	17	
24	25	Shashank D'Alia	2	2	
25	26	Nitish	28	28	
26	27	Ankit	26	26	
27	28	Rahul	17	17	

	name_y	partner_y
0	Chhavi Lachman	18
1	Kailash Harjo	23
2	Parveen Bhalla	3
3	Pranab Natarajan	22
4	Lakshmi Contractor	10

5	Aayushman Sant	8
6	Nitika Chatterjee	4
7	Kusum Bahri	6
8	Marlo Dugal	14
9	Radheshyam Dey	5
10	Hanuman Hegde	11
11	Qabeel Raman	12
12	Radhika Suri	17
13	Yash Sethi	21
14	Elias Dodiya	25
15	Shashank D'Alia	2
16	Tarun Thaker	9
17	Munni Varghese	24
18	Radha Dutt	19
19	David Mukhopadhyay	20
20	Preet Sha	16
21	Seema Kota	15
22	Fardeen Mahabir	13
23	Yasmin Palan	7
24	Esha Butala	1
25	Rahul	17
26	Nitish	28
27	Yasmin Palan	7

```
students.merge(students,how='inner',left_on='partner',right_on='student_id')[['name_x','name_y']]
```

	name_x	name_y
0	Kailash Harjo	Chhavi Lachman
1	Esha Butala	Kailash Harjo
2	Parveen Bhalla	Parveen Bhalla
3	Marlo Dugal	Pranab Natarajan
4	Kusum Bahri	Lakshmi Contractor
5	Lakshmi Contractor	Aayushman Sant
6	Tarun Thaker	Nitika Chatterjee
7	Radheshyam Dey	Kusum Bahri
8	Nitika Chatterjee	Marlo Dugal
9	Aayushman Sant	Radheshyam Dey
10	David Mukhopadhyay	Hanuman Hegde
11	Radha Dutt	Qabeel Raman
12	Munni Varghese	Radhika Suri
13	Pranab Natarajan	Yash Sethi
14	Preet Sha	Elias Dodiya
15	Elias Dodiya	Shashank D'Alia
16	Yasmin Palan	Tarun Thaker
17	Fardeen Mahabir	Munni Varghese
18	Qabeel Raman	Radha Dutt
19	Hanuman Hegde	David Mukhopadhyay
20	Seema Kota	Preet Sha
21	Yash Sethi	Seema Kota

22	Chhavi Lachman	Fardeen Mahabir
23	Radhika Suri	Yasmin Palan
24	Shashank D'Alia	Esha Butala
25	Nitish	Rahul
26	Ankit	Nitish
27	Rahul	Yasmin Palan

9. find top 3 students who did most number enrollments

`regs.merge(students,on='student_id')['name'].value_counts()` *# abhi agar value count laga rahe hain to there are possibilities ki same name ke do bachhe ho*

```
name
Chhavi Lachman      6
Tarun Thaker        5
Elias Dodiya        4
Kailash Harjo       4
Pranab Natarajan    4
Radha Dutt          4
Fardeen Mahabir     3
Yash Sethi          3
David Mukhopadhyay  3
Parveen Bhalla      2
Qabeel Raman        2
Preet Sha           2
Yasmin Palan        2
Shashank D'Alia     2
Munni Varghese      1
Seema Kota          1
Radhika Suri        1
Esha Butala         1
Name: count, dtype: int64
```

```
regs.merge(students,on='student_id').groupby(['student_id','name'])
['name']
```

```
<pandas.core.groupby.generic.SeriesGroupBy object at
0x000001E0F75F8740>
```

```
regs.merge(students,on='student_id').groupby(['student_id','name'])
['name'].count().sort_values(ascending=False).head(3)
```

```
student_id  name
23          Chhavi Lachman    6
7           Tarun Thaker     5
1           Kailash Harjo    4
Name: name, dtype: int64
```

10. find top 3 students who spent most amount of money on courses

`regs.merge(students,on='student_id').merge(courses,on='course_id').gro`

```
upby(['student_id', 'name'])
['price'].sum().sort_values(ascending=False).head(3)
```

```
student_id  name
23          Chhavi Lachman      22594
14          Pranab Natarajan    15096
19          Qabeel Raman       13498
Name: price, dtype: int64
```

Alternate syntax for merge

```
# 1st one is : students.merge(regs)
```

```
pd.merge(students, regs, how='inner', on='student_id') # jo pahle aaya
wo first table and jo baad me aaya wo second table
```

	student_id	name	partner	course_id
0	1	Kailash Harjo	23	1
1	1	Kailash Harjo	23	6
2	1	Kailash Harjo	23	10
3	1	Kailash Harjo	23	9
4	2	Esha Butala	1	5
5	3	Parveen Bhalla	3	3
6	3	Parveen Bhalla	3	5
7	7	Tarun Thaker	9	8
8	7	Tarun Thaker	9	10
9	7	Tarun Thaker	9	7
10	7	Tarun Thaker	9	2
11	7	Tarun Thaker	9	6
12	11	David Mukhopadhyay	20	7
13	11	David Mukhopadhyay	20	8
14	11	David Mukhopadhyay	20	10
15	12	Radha Dutt	19	10
16	12	Radha Dutt	19	1
17	12	Radha Dutt	19	5
18	12	Radha Dutt	19	7
19	13	Munni Varghese	24	3
20	14	Pranab Natarajan	22	9
21	14	Pranab Natarajan	22	6
22	14	Pranab Natarajan	22	4
23	14	Pranab Natarajan	22	1
24	15	Preet Sha	16	5
25	15	Preet Sha	16	1
26	16	Elias Dodiya	25	9
27	16	Elias Dodiya	25	5
28	16	Elias Dodiya	25	7
29	16	Elias Dodiya	25	3
30	17	Yasmin Palan	7	7
31	17	Yasmin Palan	7	10
32	18	Fardeen Mahabir	13	6

33	18	Fardeen Mahabir	13	1
34	18	Fardeen Mahabir	13	8
35	19	Qabeel Raman	12	4
36	19	Qabeel Raman	12	2
37	21	Seema Kota	15	1
38	22	Yash Sethi	21	3
39	22	Yash Sethi	21	5
40	22	Yash Sethi	21	6
41	23	Chhavi Lachman	18	1
42	23	Chhavi Lachman	18	4
43	23	Chhavi Lachman	18	3
44	23	Chhavi Lachman	18	6
45	23	Chhavi Lachman	18	9
46	23	Chhavi Lachman	18	5
47	24	Radhika Suri	17	4
48	25	Shashank D'Alia	2	1
49	25	Shashank D'Alia	2	10

IPL Problems

find top 3 studiums with highest sixes per match ratio
find orange cap holder of all the seasons

matches

	id	season	city	date	
team1 \					
0	1	2017	Hyderabad	2017-04-05	Sunrisers Hyderabad
1	2	2017	Pune	2017-04-06	Mumbai Indians
2	3	2017	Rajkot	2017-04-07	Gujarat Lions
3	4	2017	Indore	2017-04-08	Rising Pune Supergiant
4	5	2017	Bangalore	2017-04-08	Royal Challengers Bangalore
..
631	632	2016	Raipur	2016-05-22	Delhi Daredevils
632	633	2016	Bangalore	2016-05-24	Gujarat Lions
633	634	2016	Delhi	2016-05-25	Sunrisers Hyderabad
634	635	2016	Delhi	2016-05-27	Gujarat Lions
635	636	2016	Bangalore	2016-05-29	Sunrisers Hyderabad
team2				toss_winner	

```

toss_decision \
0   Royal Challengers Bangalore   Royal Challengers Bangalore
field
1   Rising Pune Supergiant        Rising Pune Supergiant
field
2   Kolkata Knight Riders         Kolkata Knight Riders
field
3   Kings XI Punjab              Kings XI Punjab
field
4   Delhi Daredevils             Royal Challengers Bangalore
bat
..
...
631 Royal Challengers Bangalore   Royal Challengers Bangalore
field
632 Royal Challengers Bangalore   Royal Challengers Bangalore
field
633 Kolkata Knight Riders         Kolkata Knight Riders
field
634 Sunrisers Hyderabad          Sunrisers Hyderabad
field
635 Royal Challengers Bangalore   Sunrisers Hyderabad
bat

```

```

      result  dl_applied      winner  win_by_runs \
0   normal      0      Sunrisers Hyderabad      35
1   normal      0      Rising Pune Supergiant      0
2   normal      0      Kolkata Knight Riders      0
3   normal      0      Kings XI Punjab          0
4   normal      0   Royal Challengers Bangalore      15
..
...
631 normal      0   Royal Challengers Bangalore      0
632 normal      0   Royal Challengers Bangalore      0
633 normal      0      Sunrisers Hyderabad      22
634 normal      0      Sunrisers Hyderabad      0
635 normal      0      Sunrisers Hyderabad      8

```

```

      win_by_wickets  player_of_match \
0           0      Yuvraj Singh
1           7      SPD Smith
2          10      CA Lynn
3           6      GJ Maxwell
4           0      KM Jadhav
..
...
631          6      V Kohli
632          4   AB de Villiers
633          0      MC Henriques
634          4      DA Warner
635          0      BCJ Cutting

```

		venue	umpire1
\	0	Rajiv Gandhi International Stadium, Uppal	AY Dandekar
1		Maharashtra Cricket Association Stadium	A Nand Kishore
2		Saurashtra Cricket Association Stadium	Nitin Menon
3		Holkar Cricket Stadium	AK Chaudhary
4		M Chinnaswamy Stadium	NaN
..	
631	Shaheed Veer Narayan Singh International Stadium	A Nand Kishore	
632		M Chinnaswamy Stadium	AK Chaudhary
633		Feroz Shah Kotla	M Erasmus
634		Feroz Shah Kotla	M Erasmus
635		M Chinnaswamy Stadium	HDPK Dharmasena

		umpire2	umpire3
0		NJ Llong	NaN
1		S Ravi	NaN
2		CK Nandan	NaN
3	C Shamshuddin		NaN
4		NaN	NaN
..	
631	BNJ Oxenford		NaN
632	HDPK Dharmasena		NaN
633	C Shamshuddin		NaN
634	CK Nandan		NaN
635	BNJ Oxenford		NaN

[636 rows x 18 columns]

delivery

		match_id	inning	batting_team	
bowling_team	\				
0		1	1	Sunrisers Hyderabad	Royal Challengers Bangalore
1		1	1	Sunrisers Hyderabad	Royal Challengers Bangalore
2		1	1	Sunrisers Hyderabad	Royal Challengers Bangalore
3		1	1	Sunrisers Hyderabad	Royal Challengers Bangalore

Bangalore					
4	1	1	Sunrisers Hyderabad	Royal Challengers	
Bangalore					
...	
...					
179073	11415	2	Chennai Super Kings		Mumbai
Indians					
179074	11415	2	Chennai Super Kings		Mumbai
Indians					
179075	11415	2	Chennai Super Kings		Mumbai
Indians					
179076	11415	2	Chennai Super Kings		Mumbai
Indians					
179077	11415	2	Chennai Super Kings		Mumbai
Indians					

	over	ball	batsman	non_striker	bowler
is_super_over	...	\			
0	1	1	DA Warner	S Dhawan	TS Mills
0	...				
1	1	2	DA Warner	S Dhawan	TS Mills
0	...				
2	1	3	DA Warner	S Dhawan	TS Mills
0	...				
3	1	4	DA Warner	S Dhawan	TS Mills
0	...				
4	1	5	DA Warner	S Dhawan	TS Mills
0	...				
...
...					
179073	20	2	RA Jadeja	SR Watson	SL Malinga
0	...				
179074	20	3	SR Watson	RA Jadeja	SL Malinga
0	...				
179075	20	4	SR Watson	RA Jadeja	SL Malinga
0	...				
179076	20	5	SN Thakur	RA Jadeja	SL Malinga
0	...				
179077	20	6	SN Thakur	RA Jadeja	SL Malinga
0	...				

	bye_runs	legbye_runs	noball_runs	penalty_runs	batsman_runs
\					
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	4
3	0	0	0	0	0

4	0	0	0	0	0
...
179073	0	0	0	0	1
179074	0	0	0	0	2
179075	0	0	0	0	1
179076	0	0	0	0	2
179077	0	0	0	0	0

	extra_runs	total_runs	player_dismissed	dismissal_kind
fielder				
0	0	0	NaN	NaN
NaN				
1	0	0	NaN	NaN
NaN				
2	0	4	NaN	NaN
NaN				
3	0	0	NaN	NaN
NaN				
4	2	2	NaN	NaN
NaN				
...
...				
179073	0	1	NaN	NaN
NaN				
179074	0	2	NaN	NaN
NaN				
179075	0	1	SR Watson	run out KH
Pandya				
179076	0	2	NaN	NaN
NaN				
179077	0	0	SN Thakur	lbw
NaN				

[179078 rows x 21 columns]

stadium ka info matches bale table me hai and ball pe chakka laga ki nahi wo delivery bale table me hai

so dono table ko merge karenge on the basis of match id but dono table me name different (ek me match_id and another me id) hai

```
temp_df=delivery.merge(matches,left_on='match_id',right_on='id')
temp_df
```


	match_id	inning	batting_team \	
0	1	1	Sunrisers	Hyderabad
1	1	1	Sunrisers	Hyderabad
2	1	1	Sunrisers	Hyderabad
3	1	1	Sunrisers	Hyderabad
4	1	1	Sunrisers	Hyderabad
...
150455	636	2	Royal Challengers	Bangalore
150456	636	2	Royal Challengers	Bangalore
150457	636	2	Royal Challengers	Bangalore
150458	636	2	Royal Challengers	Bangalore
150459	636	2	Royal Challengers	Bangalore

	bowling_team		over	ball	batsman	
non_striker \						
0	Royal Challengers	Bangalore	1	1	DA Warner	S
Dhawan	Royal Challengers	Bangalore	1	2	DA Warner	S
1	Royal Challengers	Bangalore	1	3	DA Warner	S
Dhawan	Royal Challengers	Bangalore	1	4	DA Warner	S
2	Royal Challengers	Bangalore	1	5	DA Warner	S
Dhawan
...
...	Sunrisers	Hyderabad	20	2	Sachin Baby	CJ
150455	Sunrisers	Hyderabad	20	3	Sachin Baby	CJ
150456	Sunrisers	Hyderabad	20	4	Iqbal Abdulla	
150457	Sunrisers	Hyderabad	20	5	Sachin Baby	Iqbal
150458	Sunrisers	Hyderabad	20	6	Iqbal Abdulla	
150459	Sunrisers	Hyderabad	20	6	Iqbal Abdulla	
Sachin Baby						

	bowler	is_super_over	...	result	dl_applied
winner \					
0	TS Mills	0	...	normal	0
Hyderabad	TS Mills	0	...	normal	0
1	TS Mills	0	...	normal	0
Hyderabad	TS Mills	0	...	normal	0
2	TS Mills	0	...	normal	0
Hyderabad	TS Mills	0	...	normal	0
3	TS Mills	0	...	normal	0
Hyderabad	TS Mills	0	...	normal	0
4	TS Mills	0	...	normal	0
Hyderabad					

...
150455	B Kumar	0	...	normal	0 Sunrisers
Hyderabad					
150456	B Kumar	0	...	normal	0 Sunrisers
Hyderabad					
150457	B Kumar	0	...	normal	0 Sunrisers
Hyderabad					
150458	B Kumar	0	...	normal	0 Sunrisers
Hyderabad					
150459	B Kumar	0	...	normal	0 Sunrisers
Hyderabad					

	win_by_runs	win_by_wickets	player_of_match	\
0	35	0	Yuvraj Singh	
1	35	0	Yuvraj Singh	
2	35	0	Yuvraj Singh	
3	35	0	Yuvraj Singh	
4	35	0	Yuvraj Singh	
...	
150455	8	0	BCJ Cutting	
150456	8	0	BCJ Cutting	
150457	8	0	BCJ Cutting	
150458	8	0	BCJ Cutting	
150459	8	0	BCJ Cutting	

		venue	umpire1	\
0	Rajiv Gandhi International Stadium, Uppal		AY Dandekar	
1	Rajiv Gandhi International Stadium, Uppal		AY Dandekar	
2	Rajiv Gandhi International Stadium, Uppal		AY Dandekar	
3	Rajiv Gandhi International Stadium, Uppal		AY Dandekar	
4	Rajiv Gandhi International Stadium, Uppal		AY Dandekar	
...		
150455	M Chinnaswamy Stadium	HDPK Dharmasena		
150456	M Chinnaswamy Stadium	HDPK Dharmasena		
150457	M Chinnaswamy Stadium	HDPK Dharmasena		
150458	M Chinnaswamy Stadium	HDPK Dharmasena		
150459	M Chinnaswamy Stadium	HDPK Dharmasena		

	umpire2	umpire3
0	NJ Llong	NaN
1	NJ Llong	NaN
2	NJ Llong	NaN
3	NJ Llong	NaN
4	NJ Llong	NaN
...
150455	BNJ Oxenford	NaN
150456	BNJ Oxenford	NaN
150457	BNJ Oxenford	NaN
150458	BNJ Oxenford	NaN

150459 BNJ Oxenford NaN

[150460 rows x 39 columns]

```
six_df=temp_df[temp_df['batsman_runs']==6]
six_df
```

	match_id	inning	batting_team \	
10	1	1	Sunrisers	Hyderabad
47	1	1	Sunrisers	Hyderabad
75	1	1	Sunrisers	Hyderabad
89	1	1	Sunrisers	Hyderabad
91	1	1	Sunrisers	Hyderabad
...		
150395	636	2	Royal Challengers	Bangalore
150413	636	2	Royal Challengers	Bangalore
150428	636	2	Royal Challengers	Bangalore
150440	636	2	Royal Challengers	Bangalore
150453	636	2	Royal Challengers	Bangalore

	bowling_team	over	ball	batsman	
non_striker \					
10	Royal Challengers Bangalore	2	4	DA Warner	S
Dhawan					
47	Royal Challengers Bangalore	8	4	MC Henriques	S
Dhawan					
75	Royal Challengers Bangalore	13	2	Yuvraj Singh	MC
Henriques					
89	Royal Challengers Bangalore	15	3	Yuvraj Singh	MC
Henriques					
91	Royal Challengers Bangalore	15	5	MC Henriques	
Yuvraj Singh					
...	
...					
150395	Sunrisers Hyderabad	10	2	V Kohli	
CH Gayle					
150413	Sunrisers Hyderabad	13	2	V Kohli	AB de
Villiers					
150428	Sunrisers Hyderabad	15	5	SR Watson	
KL Rahul					
150440	Sunrisers Hyderabad	17	5	STR Binny	
Sachin Baby					
150453	Sunrisers Hyderabad	19	6	Sachin Baby	CJ
Jordan					

	bowler	is_super_over	...	result	dl_applied \
10	A Choudhary	0	...	normal	0
47	TM Head	0	...	normal	0
75	A Choudhary	0	...	normal	0
89	S Aravind	0	...	normal	0

91	S Aravind	0	...	normal	0
...
150395	Mustafizur Rahman	0	...	normal	0
150413	BB Sran	0	...	normal	0
150428	MC Henriques	0	...	normal	0
150440	Mustafizur Rahman	0	...	normal	0
150453	Mustafizur Rahman	0	...	normal	0
winner win_by_runs win_by_wickets					
player_of_match \					
10	Sunrisers Hyderabad	35		0	Yuvraj Singh
47	Sunrisers Hyderabad	35		0	Yuvraj Singh
75	Sunrisers Hyderabad	35		0	Yuvraj Singh
89	Sunrisers Hyderabad	35		0	Yuvraj Singh
91	Sunrisers Hyderabad	35		0	Yuvraj Singh
...
...					
150395	Sunrisers Hyderabad	8		0	BCJ Cutting
150413	Sunrisers Hyderabad	8		0	BCJ Cutting
150428	Sunrisers Hyderabad	8		0	BCJ Cutting
150440	Sunrisers Hyderabad	8		0	BCJ Cutting
150453	Sunrisers Hyderabad	8		0	BCJ Cutting
venue umpire1 \					
10	Rajiv Gandhi International Stadium, Uppal			AY Dandekar	
47	Rajiv Gandhi International Stadium, Uppal			AY Dandekar	
75	Rajiv Gandhi International Stadium, Uppal			AY Dandekar	
89	Rajiv Gandhi International Stadium, Uppal			AY Dandekar	
91	Rajiv Gandhi International Stadium, Uppal			AY Dandekar	
...
150395	M Chinnaswamy Stadium	HDPK	Dharmasena		
150413	M Chinnaswamy Stadium	HDPK	Dharmasena		
150428	M Chinnaswamy Stadium	HDPK	Dharmasena		
150440	M Chinnaswamy Stadium	HDPK	Dharmasena		
150453	M Chinnaswamy Stadium	HDPK	Dharmasena		
umpire2 umpire3					
10	NJ Llong	NaN			
47	NJ Llong	NaN			
75	NJ Llong	NaN			

```

89      NJ Llong      NaN
91      NJ Llong      NaN
...      ...      ...
150395  BNJ Oxenford   NaN
150413  BNJ Oxenford   NaN
150428  BNJ Oxenford   NaN
150440  BNJ Oxenford   NaN
150453  BNJ Oxenford   NaN

```

```
[6523 rows x 39 columns]
```

```

num_sixes=six_df.groupby('venue')['venue'].count()
num_sixes

```

```

venue
Barabati Stadium      68
Brabourne Stadium    114
Buffalo Park          27
De Beers Diamond Oval 34
Dr DY Patil Sports Academy 173
Dr. Y.S. Rajasekhara Reddy ACA-VDCA Cricket Stadium 129
Dubai International Cricket Stadium 58
Eden Gardens          594
Feroz Shah Kotla      636
Green Park            36
Himachal Pradesh Cricket Association Stadium 108
Holkar Cricket Stadium 88
JSCA International Stadium Complex 76
Kingsmead            130
M Chinnaswamy Stadium 873
MA Chidambaram Stadium, Chepauk 507
Maharashtra Cricket Association Stadium 169
Nehru Stadium         39
New Wanderers Stadium 64
Newlands              63
OUTsurance Oval       8
Punjab Cricket Association IS Bindra Stadium, Mohali 87
Punjab Cricket Association Stadium, Mohali 307
Rajiv Gandhi International Stadium, Uppal 492
Sardar Patel Stadium, Motera 130
Saurashtra Cricket Association Stadium 104
Sawai Mansingh Stadium 243
Shaheed Veer Narayan Singh International Stadium 44
Sharjah Cricket Stadium 76
Sheikh Zayed Stadium  40
St George's Park      62
Subrata Roy Sahara Stadium 141
SuperSport Park       120
Vidarbha Cricket Association Stadium, Jamtha 26

```

```

Wankhede Stadium 657
Name: venue, dtype: int64

num_matches=matches['venue'].value_counts()
num_matches

venue
M Chinnaswamy Stadium 66
Eden Gardens 61
Feroz Shah Kotla 60
Wankhede Stadium 57
Rajiv Gandhi International Stadium, Uppal 49
MA Chidambaram Stadium, Chepauk 48
Punjab Cricket Association Stadium, Mohali 35
Sawai Mansingh Stadium 33
Dr DY Patil Sports Academy 17
Subrata Roy Sahara Stadium 17
Maharashtra Cricket Association Stadium 15
Kingsmead 15
Sardar Patel Stadium, Motera 12
SuperSport Park 12
Punjab Cricket Association IS Bindra Stadium, Mohali 11
Dr. Y.S. Rajasekhara Reddy ACA-VDCA Cricket Stadium 11
Brabourne Stadium 11
Saurashtra Cricket Association Stadium 10
Himachal Pradesh Cricket Association Stadium 9
New Wanderers Stadium 8
Newlands 7
St George's Park 7
Sheikh Zayed Stadium 7
JSCA International Stadium Complex 7
Barabati Stadium 7
Dubai International Cricket Stadium 7
Shaheed Veer Narayan Singh International Stadium 6
Sharjah Cricket Stadium 6
Holkar Cricket Stadium 5
Nehru Stadium 5
Green Park 4
Vidarbha Cricket Association Stadium, Jamtha 3
De Beers Diamond Oval 3
Buffalo Park 3
OUTsurance Oval 2
Name: count, dtype: int64

num_sixes/num_matches

venue
Barabati Stadium 9.714286
Brabourne Stadium 10.363636
Buffalo Park 9.000000

```

De Beers Diamond Oval	11.333333
Dr DY Patil Sports Academy	10.176471
Dr. Y.S. Rajasekhara Reddy ACA-VDCA Cricket Stadium	11.727273
Dubai International Cricket Stadium	8.285714
Eden Gardens	9.737705
Feroz Shah Kotla	10.600000
Green Park	9.000000
Himachal Pradesh Cricket Association Stadium	12.000000
Holkar Cricket Stadium	17.600000
JSCA International Stadium Complex	10.857143
Kingsmead	8.666667
M Chinnaswamy Stadium	13.227273
MA Chidambaram Stadium, Chepauk	10.562500
Maharashtra Cricket Association Stadium	11.266667
Nehru Stadium	7.800000
New Wanderers Stadium	8.000000
Newlands	9.000000
OUTsurance Oval	4.000000
Punjab Cricket Association IS Bindra Stadium, Mohali	7.909091
Punjab Cricket Association Stadium, Mohali	8.771429
Rajiv Gandhi International Stadium, Uppal	10.040816
Sardar Patel Stadium, Motera	10.833333
Saurashtra Cricket Association Stadium	10.400000
Sawai Mansingh Stadium	7.363636
Shaheed Veer Narayan Singh International Stadium	7.333333
Sharjah Cricket Stadium	12.666667
Sheikh Zayed Stadium	5.714286
St George's Park	8.857143
Subrata Roy Sahara Stadium	8.294118
SuperSport Park	10.000000
Vidarbha Cricket Association Stadium, Jamtha	8.666667
Wankhede Stadium	11.526316

dtype: float64

```
(num_sixes/num_matches).sort_values(ascending=False).head(3)
```

venue	
Holkar Cricket Stadium	17.600000
M Chinnaswamy Stadium	13.227273
Sharjah Cricket Stadium	12.666667

dtype: float64

```
# find orange cup holder of all the seasons
```

```
temp_df.groupby(['season', 'batsman'])['batsman_runs'].sum()
```

season	batsman	
2008	A Chopra	42
	A Kumble	13
	A Mishra	37
	A Mukund	0

	A Nehra	3
	...	
2017	Washington Sundar	9
	YK Pathan	143
	YS Chahal	13
	Yuvraj Singh	252
	Z Khan	4

Name: batsman_runs, Length: 1531, dtype: int64

```
temp_df.groupby(['season', 'batsman'])
['batsman_runs'].sum().reset_index()
```

	season	batsman	batsman_runs
0	2008	A Chopra	42
1	2008	A Kumble	13
2	2008	A Mishra	37
3	2008	A Mukund	0
4	2008	A Nehra	3
...
1526	2017	Washington Sundar	9
1527	2017	YK Pathan	143
1528	2017	YS Chahal	13
1529	2017	Yuvraj Singh	252
1530	2017	Z Khan	4

[1531 rows x 3 columns]

```
temp_df.groupby(['season', 'batsman'])
['batsman_runs'].sum().reset_index().sort_values('batsman_runs', ascending=False).drop_duplicates(subset=['season'], keep='first')
```

	season	batsman	batsman_runs
1383	2016	V Kohli	973
910	2013	MEK Hussey	733
684	2012	CH Gayle	733
1088	2014	RV Uthappa	660
1422	2017	DA Warner	641
446	2010	SR Tendulkar	618
115	2008	SE Marsh	616
502	2011	CH Gayle	608
229	2009	ML Hayden	572
1148	2015	DA Warner	562

```
temp_df.groupby(['season', 'batsman'])
['batsman_runs'].sum().reset_index().sort_values('batsman_runs', ascending=False).drop_duplicates(subset=['season'], keep='first').sort_values('batsman_runs', ascending=False)
```

	season	batsman	batsman_runs
1383	2016	V Kohli	973
910	2013	MEK Hussey	733

684	2012	CH Gayle	733
1088	2014	RV Uthappa	660
1422	2017	DA Warner	641
446	2010	SR Tendulkar	618
115	2008	SE Marsh	616
502	2011	CH Gayle	608
229	2009	ML Hayden	572
1148	2015	DA Warner	562