1.

teachC = {'a','b','c','p','d','e','f','g','h','i'}

teachS = {'a','m','n','c','i'}

print(f"All teachers who have taken my class : {teachS.union(teachC)}")

print(f"All teachers who have not take your class : {teachC.difference(teachS)}")

print(f"All teachers who have taken class in school : {teachS}")

Output:

All teachers who have taken my class : {'c', 'f', 'n', 'h', 'e', 'g', 'm', 'd', 'i', 'a', 'b', 'p'}

All teachers who have not take your class : {'f', 'h', 'g', 'e', 'd', 'b', 'p'}

All teachers who have taken class in school : {'c', 'n', 'm', 'i', 'a'}

2.

set1 = {('madhurima','painting'),

('arnab','photography'),

('agnibesh','music'),

('dipayan','comedy'),

('abhishek charan','robotics')}

set2 = {('akash','football'),

('madhurima,painting'),

('arnab','photography'),

('raghib','maths'),

('laxmi','voleyball')}

print(f"Set of all friends : {set1.union(set2)}\n")

print(f"Common friends in both school and college : {set1.intersection(set2)}\n")

if(set1.issuperset(set2)):

print("Set1 is the superset of set2\n")

else:

print("Set1 is not the superset of set2\n")

Output:

Set of all friends : {('dipayan', 'comedy'), 'madhurima,painting', ('raghib', 'maths'), ('arnab', 'photography'), ('abhishek charan', 'robotics'), ('akash', 'football'), ('agnibesh', 'music'), ('laxmi', 'voleyball'), ('madhurima', 'painting')}

Common friends in both school and college : {('arnab', 'photography')}

Set1 is not the superset of set2

3.

palin=[]

pri=[]

nat=[]

for num in range(500):

temp=num

rev=0

while(num>0):

dig=num%10

rev=rev\*10+dig

num=num//10

if(temp==rev):

palin.append(temp)

for val in range(500):

if val > 1:

for n in range(2, val):

if (val % n) == 0:

break

else:

pri.append(val)

for n in range(500):

nat.append(n)

s\_palin=set(palin)

s\_pri=set(pri)

s\_nat=set(nat)

print(s\_palin.intersection(s\_pri))

print("\n\n")

print(s\_pri.difference(s\_palin))

print("\n\n")

print(s\_palin.difference(s\_pri))

print("\n\n")

print(s\_nat.difference(s\_pri,s\_palin))

Output:

{353, 2, 3, 131, 5, 101, 7, 383, 11, 181, 373, 151, 313, 191}

{13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 103, 107, 109, 113, 127, 137, 139, 149, 157, 163, 167, 173, 179, 193, 197, 199, 211, 223, 227, 229, 233, 239, 241, 251, 257, 263, 269, 271, 277, 281, 283, 293, 307, 311, 317, 331, 337, 347, 349, 359, 367, 379, 389, 397, 401, 409, 419, 421, 431, 433, 439, 443, 449, 457, 461, 463, 467, 479, 487, 491, 499}

{1, 4, 6, 262, 8, 9, 393, 141, 272, 404, 22, 282, 414, 33, 161, 292, 424, 171, 44, 303, 434, 55, 444, 66, 323, 454, 202, 77, 333, 464, 212, 343, 88, 474, 222, 99, 484, 232, 363, 494, 111, 242, 121, 252}

{0, 10, 12, 14, 15, 16, 18, 20, 21, 24, 25, 26, 27, 28, 30, 32, 34, 35, 36, 38, 39, 40, 42, 45, 46, 48, 49, 50, 51, 52, 54, 56, 57, 58, 60, 62, 63, 64, 65, 68, 69, 70, 72, 74, 75, 76, 78, 80, 81, 82, 84, 85, 86, 87, 90, 91, 92, 93, 94, 95, 96, 98, 100, 102, 104, 105, 106, 108, 110, 112, 114, 115, 116, 117, 118, 119, 120, 122, 123, 124, 125, 126, 128, 129, 130, 132, 133, 134, 135, 136, 138, 140, 142, 143, 144, 145, 146, 147, 148, 150, 152, 153, 154, 155, 156, 158, 159, 160, 162, 164, 165, 166, 168, 169, 170, 172, 174, 175, 176, 177, 178, 180, 182, 183, 184, 185, 186, 187, 188, 189, 190, 192, 194, 195, 196, 198, 200, 201, 203, 204, 205, 206, 207, 208, 209, 210, 213, 214, 215, 216, 217, 218, 219, 220, 221, 224, 225, 226, 228, 230, 231, 234, 235, 236, 237, 238, 240, 243, 244, 245, 246, 247, 248, 249, 250, 253, 254, 255, 256, 258, 259, 260, 261, 264, 265, 266, 267, 268, 270, 273, 274, 275, 276, 278, 279, 280, 284, 285, 286, 287, 288, 289, 290, 291, 294, 295, 296, 297, 298, 299, 300, 301, 302, 304, 305, 306, 308, 309, 310, 312, 314, 315, 316, 318, 319, 320, 321, 322, 324, 325, 326, 327, 328, 329, 330, 332, 334, 335, 336, 338, 339, 340, 341, 342, 344, 345, 346, 348, 350, 351, 352, 354, 355, 356, 357, 358, 360, 361, 362, 364, 365, 366, 368, 369, 370, 371, 372, 374, 375, 376, 377, 378, 380, 381, 382, 384, 385, 386, 387, 388, 390, 391, 392, 394, 395, 396, 398, 399, 400, 402, 403, 405, 406, 407, 408, 410, 411, 412, 413, 415, 416, 417, 418, 420, 422, 423, 425, 426, 427, 428, 429, 430, 432, 435, 436, 437, 438, 440, 441, 442, 445, 446, 447, 448, 450, 451, 452, 453, 455, 456, 458, 459, 460, 462, 465, 466, 468, 469, 470, 471, 472, 473, 475, 476, 477, 478, 480, 481, 482, 483, 485, 486, 488, 489, 490, 492, 493, 495, 496, 497, 498}

4.

sports = {"badminton":["P.V. Sindhu","Saina Nehwal","Jwala Gutta"],

"basketball":["Satnam Singh","Prashanti Singh"],

"cricket":["Dhoni","Sachin","Sourav"],

"football":["Sunil Chhetri","Gurpreet Singh Sandhu"]}

sport=input("Select a Sport : ")

players=sports.get(sport)

for ply in players:

print("\n",ply)

Output:

Select a Sport : badminton

P.V. Sindhu

Saina Nehwal

Jwala Gutta

5.

import random

words={

"muricide":"killing or killer of mice or rats",

"acrophonic":"using a symbol for the initial sound of a thing",

"mogadore":"ribbed silk used in making neckties",

"zebrule":"hybrid offspring of male zebra and female horse",

"cachinnate":"to laugh loudly and inappropriately",

"wiredraw":"to reduce fluid pressure by passing it through a small orifice",

"crystallomancy":"divination by means of clear objects",

"thanatism":"belief that the soul dies with the body",

"isograph":"line connecting points of same linguistic usage in some respect",

"timbrophily":"love or fondness for stamps; stamp-collecting"

}

vals=[]

for x in words.values():

vals.append(x)

choice=random.choice(vals)

print(choice)

ans=input("Guess the word : ")

for word,mean in words.items():

if ans==word:

print("Your Answer is Correct!")

elif choice==mean:

print("Correct answer is ",word)

Output:

using a symbol for the initial sound of a thing

Guess the word : acrophonic

Your Answer is Correct!

6.

batsman = {'dhoni','virat kohl','sachin','gautam gambhir','rahul dravid','jadeja','laximane','maxwell','zampa','shaqib','avisek'}

bowler={'bumrah','bhubaneshwar','shaqib','zampa','jadeja','stark','malinga','shoaib','waqar','agnibesh','avisek'}

print(batsman.difference(bowler))

print(bowler.difference(batsman))

print(batsman.intersection(bowler))

Output:

{'dhoni', 'laximane', 'rahul dravid', 'virat kohl', 'maxwell', 'sachin', 'gautam gambhir'}

{'malinga', 'bumrah', 'waqar', 'stark', 'shoaib', 'agnibesh', 'bhubaneshwar'}

{'avisek', 'shaqib', 'zampa', 'jadeja'}