* Se dau urmatoarele liste:

employees = ['Michal', 'Harry', 'Susan', 'Dan', 'Christen']

email = ['michal@comp.com', 'harryf@comp.com',

'susan@comp.com', 'dan2@comp.com', '[chris@comp.com](mailto:chris@comp.com)']

Scrie un program care adauga intr-un dictionar perechile cheie:valoare, reprezentate de elementele din liste. Care vor fi cheile?

* Se da urmatorul dictionar care contine date despre vremea din Bucuresti.

{

"coord": {

"lon": 26.1063,

"lat": 44.4323

},

"weather": [

{

"id": 800,

"main": "Clear",

"description": "clear sky",

"icon": "01n"

}

],

"base": "stations",

"main": {

"temp": 286.87,

"feels\_like": 286.07,

"temp\_min": 284.2,

"temp\_max": 289.74,

"pressure": 1022,

"humidity": 68

},

"visibility": 10000,

"wind": {

"speed": 1.03,

"deg": 260

},

"clouds": {

"all": 0

},

"dt": 1666984646,

"sys": {

"type": 2,

"id": 2037828,

"country": "RO",

"sunrise": 1666932421,

"sunset": 1666969892

},

"timezone": 10800,

"id": 683506,

"name": "Bucharest",

"cod": 200

}

Afisati cate grade Celsius sunt in Bucuresti (temperatura din dictionar este in grade Kelvin) si cum este cerul (clouds, clear sky, etc)

* Se da un dictionar cu notele de la matematica ale unor elevi, notele fiind valori:

math\_grades = {

'Marius': 8.0,

'Andreea': 9.5,

'Adrian': 7.9,

'Bianca': 10}

Afisati elevul/elevii cu nota maxima.

* Se da urmatorul dictionar.

employee = {

1: {'name':'Andrei', 'salary':100},

2: {'name':'Vlad', 'salary':500},

3: {'name':'Ioana', 'salary':330}

}

Modificati salariul unui angajat pe baza id-ului de angajat introdus de la tastatura.

Daca id-ul nu este al unui angajat se va cere reintroducere.

Noul salariu se introduce tot de la tastatura.

* Afisati de cate ori apare fiecare intreg din urmatoarea lista:

numbers = [1, 1, 3, 7, 9, 2, 3, 100, 9, 333, 2.0, 5, 1, 3, 3]

Rezultatul va fi stocat intr-un dictionar unde cheile sunt elementele si valorile frecventa lor de aparitie

* Fie dictionarul:

info\_grades = {

'Maria': [8, 9, 10],

'Bogdan': [8.6, 7.3, 9.9, 10],

'Ilinca': [10, 10],

'Andra': [9.5, 7, 9],

'Daniel': [6, 10, 9.7]

}

* Stocati intr-un tuplu toti elevii care au 3 note
* Afisati un dictionar care are perechile cheie: valoare formate din numele elevilor (cheile) si media tuturor notelor lor (valorile)
* Care este elevul cu media cea mai mare?
* Se da o lista cu fisiere:

files = ['mare\_2023.jpeg', 'test.txt', 'liste.py', 'cv.pdf']

Afiseaza care este calea unde trebuie salvate fisierele din aceasta lista tinand cont de dictionarul de mai jos

paths = {

'C://Downloads//Images': ['jpg', 'png', 'jpeg'],

'C://Downloads//Text': ['txt'],

'C://Downloads//Python\_files': ['py'],

'C://Downloads//PDF': ['pdf'],

}

* cars = {

'Dacia': 15000,

'Toyota': 20000,

'BMW': 50000,

'Audi': 45000,

'Hyundai': 16500,

'Mercedes': 70000

}

Afisati un alt dictionar cu masinile mai ieftine de 40000 Euro.

Afisati un dictionar in care pretul masinilor este in RON.

EX 2

bucuresti = {  
 "coord": {  
 "lon": 26.1063,  
 "lat": 44.4323  
 },  
 "weather": [  
 {  
 "id": 800,  
 "main": "Clear",  
 "description": "clear sky",  
 "icon": "01n"  
 }  
 ],  
 "base": "stations",  
 "main": {  
 "temp": 286.87,  
 "feels\_like": 286.07,  
 "temp\_min": 284.2,  
 "temp\_max": 289.74,  
 "pressure": 1022,  
 "humidity": 68  
 },  
 "visibility": 10000,  
 "wind": {  
 "speed": 1.03,  
 "deg": 260  
 },  
 "clouds": {  
 "all": 0  
 },  
 "dt": 1666984646,  
 "sys": {  
 "type": 2,  
 "id": 2037828,  
 "country": "RO",  
 "sunrise": 1666932421,  
 "sunset": 1666969892  
 },  
 "timezone": 10800,  
 "id": 683506,  
 "name": "Bucharest",  
 "cod": 200  
}  
main1 = bucuresti["main"]  
temp\_celsius = (f"{bucuresti["main"]["temp"] - 273.15:.1f}")  
oras = bucuresti["name"]  
cerul = bucuresti["weather"][0]["description"]  
print(f"Temperatura in orasul {oras} este de {temp\_celsius} grade celsius si cerul este {cerul}.")

EX 3

math\_grades = {  
 'Marius': 8.0,  
 'Andreea': 9.5,  
 'Adrian': 7.9,  
 'Bianca': 10  
}  
nota\_maxima = max(math\_grades.values())  
for elev, nota in math\_grades.items():  
 if nota == nota\_maxima:  
 print(f"{elev} a luat nota maxima ({nota}).")

EX 4

employee = {  
 1: {'name':'Andrei', 'salary':100},  
 2: {'name':'Vlad', 'salary':500},  
 3: {'name':'Ioana', 'salary':330}  
}  
while True:  
 nume = input("Scrie aici numele de angajat:")  
 if nume.title() == employee[1]['name']:  
 print(f"Salariul este de: {employee[1]['salary']}")  
 employee[1]['salary'] = input("Scrie aici noul salariu:")  
 print(f"Noul salariu este de: {employee[1]['salary']}")  
 break  
 elif nume.title() == employee[2]['name']:  
 print(f"Salariul este de: {employee[2]['salary']}")  
 employee[2]['salary'] = input("Scrie aici noul salariu:")  
 print(f"Noul salariu este de: {employee[2]['salary']}")  
 break  
 elif nume.title() == employee[3]['name']:  
 print(f"Salariul este de: {employee[3]['salary']}")  
 employee[3]['salary'] = input("Scrie aici noul salariu:")  
 print(f"Noul salariu este de: {employee[3]['salary']}")  
 break  
 else:  
 print(f"Nu am gasit numele de angajat: {nume} ")

EX 5

numbers = [1, 1, 3, 7, 9, 2, 3, 100, 9, 333, 2.0, 5, 1, 3, 3]  
numaratori = {}  
for numar in numbers:  
 if isinstance(numar, (int, float)):  
 if numar not in numaratori:  
 numaratori[numar] = 1  
 else:  
 numaratori[numar] += 1  
  
print(numaratori)

EX 6

info\_grades = {  
 'Maria': [8, 9, 10],  
 'Bogdan': [8.6, 7.3, 9.9, 10],  
 'Ilinca': [10, 10],  
 'Andra': [9.5, 7, 9],  
 'Daniel': [6, 10, 9.7]  
}  
nr\_note\_3 = ()  
medie = {}  
total\_g = 0  
  
  
  
for name, grade in info\_grades.items():  
 total\_g += (grade)  
 avarage = total\_g / len(grade)  
 if len(grade) == 3:  
 nr\_note\_3 += (name,)  
 print(grade)  
  
  
print(nr\_note\_3)  
print(medie)

Nu reusesc sa il fac!!

EX 7

files = ['mare\_2023.jpeg', 'test.txt', 'liste.py', 'cv.pdf']  
paths = {  
 'C://Downloads//Images': ['jpg', 'png', 'jpeg'],  
 'C://Downloads//Text': ['txt'],  
 'C://Downloads//Python\_files': ['py'],  
 'C://Downloads//PDF': ['pdf'],  
}  
locatie\_files = {}  
for path, fisiere in paths.items() :  
 if fisiere == paths['C://Downloads//Images']:  
 locatie\_files[files[0]] = 'C://Downloads//Images'  
 elif fisiere == paths['C://Downloads//Text']:  
 locatie\_files[files[1]] = 'C://Downloads//Text'  
 elif fisiere == paths['C://Downloads//Python\_files']:  
 locatie\_files[files[2]] = 'C://Downloads//Python\_files'  
 elif fisiere == paths['C://Downloads//PDF']:  
 locatie\_files[files[3]] = 'C://Downloads//PDF'  
  
print(locatie\_files)

EX 8

cars = {  
 'Dacia': 15000,  
 'Toyota': 20000,  
 'BMW': 50000,  
 'Audi': 45000,  
 'Hyundai': 16500,  
 'Mercedes': 70000  
}  
sub\_pret = {}  
pret\_ron = {}  
for car, pret in cars.items():  
 if pret < 40000:  
 sub\_pret[car] = pret  
print(sub\_pret)  
for car, ron in cars.items():  
 pret\_ron[car] = int(ron \* 5.08)  
print(pret\_ron)