**Python. Итоговое задание.**

**Данные сотрудников:**

| **ФИО** | **Должность** | **Дата найма** | **Оклад** | **Премия** | **Праздничная премия** | **Индексация** |
| --- | --- | --- | --- | --- | --- | --- |
| Иванов Иван Иванович | Менеджер | 22.10.2013 | 250000 | 0 | 2000 | 17500 |
| Сорокина Екатерина Матвеевна | Аналитик | 12.03.2020 | 75000 | 0 | 2000 | 3750 |
| Струков Иван Сергеевич | Старший программист | 23.04.2012 | 150000 | 4500 | 2000 | 7500 |
| Корнеева Анна Игоревна | Ведущий программист | 22.02.2015 | 120000 | 3600 | 2000 | 6000 |
| Старчиков Сергей Анатольевич | Младший программист | 12.11.2021 | 50000 | 1500 | 2000 | 2500 |
| Бутенко Артем Андреевич | Архитектор | 12.02.2010 | 200000 | 0 | 2000 | 14000 |
| Савченко Алина Сергеевна | Старший аналитик | 13.04.2016 | 100000 | 0 | 2000 | 5000 |

Код программы:

import csv

import json

from datetime import datetime

# Исходные данные сотрудников

employees = [

{"ФИО": "Иванов Иван Иванович", "Должность": "Менеджер", "Дата найма": "22.10.2013", "Оклад": 250000, "Пол": "Мужчина"},

{"ФИО": "Сорокина Екатерина Матвеевна", "Должность": "Аналитик", "Дата найма": "12.03.2020", "Оклад": 75000, "Пол": "Женщина"},

{"ФИО": "Струков Иван Сергеевич", "Должность": "Старший программист", "Дата найма": "23.04.2012", "Оклад": 150000, "Пол": "Мужчина"},

{"ФИО": "Корнеева Анна Игоревна", "Должность": "Ведущий программист", "Дата найма": "22.02.2015", "Оклад": 120000, "Пол": "Женщина"},

{"ФИО": "Старчиков Сергей Анатольевич", "Должность": "Младший программист", "Дата найма": "12.11.2021", "Оклад": 50000, "Пол": "Мужчина"},

{"ФИО": "Бутенко Артем Андреевич", "Должность": "Архитектор", "Дата найма": "12.02.2010", "Оклад": 200000, "Пол": "Мужчина"},

{"ФИО": "Савченко Алина Сергеевна", "Должность": "Старший аналитик", "Дата найма": "13.04.2016", "Оклад": 100000, "Пол": "Женщина"}

]

# 1. Функция расчета премии программистам (3%)

def calculate\_programmer\_bonus(employees):

for employee in employees:

if "программист" in employee["Должность"].lower():

employee["Премия"] = employee["Оклад"] \* 0.03

else:

employee["Премия"] = 0

return employees

# 2. Функция расчета праздничной премии (2000 рублей)

def calculate\_holiday\_bonus(employees):

for employee in employees:

if employee["Пол"] == "Мужчина":

employee["Праздничная премия"] = 2000 # Премия к 23 февраля

else:

employee["Праздничная премия"] = 2000 # Премия к 8 марта

return employees

# 3. Функция индексации зарплат

def calculate\_salary\_indexation(employees):

for employee in employees:

date\_hired = datetime.strptime(employee["Дата найма"], "%d.%m.%Y")

years\_worked = (datetime.now() - date\_hired).days / 365

if years\_worked > 10:

employee["Индексация"] = employee["Оклад"] \* 0.07

else:

employee["Индексация"] = employee["Оклад"] \* 0.05

return employees

# 4. Функция для составления списка сотрудников, отработавших более 6 месяцев

def get\_employees\_for\_vacation(employees):

vacation\_list = []

for employee in employees:

date\_hired = datetime.strptime(employee["Дата найма"], "%d.%m.%Y")

months\_worked = (datetime.now() - date\_hired).days / 30

if months\_worked > 6:

vacation\_list.append(employee)

return vacation\_list

# 5. Функция записи в CSV файл

def save\_to\_csv(employees, filename):

fieldnames = ["ФИО", "Должность", "Дата найма", "Оклад", "Премия", "Праздничная премия", "Индексация"]

with open(filename, 'w', newline='', encoding='utf-8') as csvfile:

writer = csv.DictWriter(csvfile, fieldnames=fieldnames)

writer.writeheader()

writer.writerows(employees)

# 6. Функция записи в JSON файл

def save\_to\_json(employees, filename):

with open(filename, 'w', encoding='utf-8') as jsonfile:

json.dump(employees, jsonfile, ensure\_ascii=False, indent=4)

# Пример использования функций

employees = calculate\_programmer\_bonus(employees)

employees = calculate\_holiday\_bonus(employees)

employees = calculate\_salary\_indexation(employees)

vacation\_list = get\_employees\_for\_vacation(employees)

# Сохранение данных в файлы

save\_to\_csv(employees, 'employees.csv')

save\_to\_json(employees, 'employees.json')

**Пояснения к функциям:**

1. **calculate\_programmer\_bonus** – проверяет, является ли сотрудник программистом (по должности), и добавляет 3% премии от его оклада.
2. **calculate\_holiday\_bonus** – добавляет фиксированную премию в 2000 рублей в зависимости от пола сотрудника: мужчинам ко Дню защитника Отечества, женщинам – к 8 марта.
3. **calculate\_salary\_indexation** – рассчитывает индексацию оклада в зависимости от стажа работы: если сотрудник работает более 10 лет, он получает индексацию 7%, иначе — 5%.
4. **get\_employees\_for\_vacation** – отбирает сотрудников, которые проработали более 6 месяцев.
5. **save\_to\_csv** – сохраняет данные в CSV файл.
6. **save\_to\_json** – сохраняет данные в JSON файл.