

Упражнения по программированию главы 11

```
# coding: utf-8
```

Упражнение по программированию 11.1. Классы *Employee* и *ProductionWorker*

```
# import emp_prod
from objects import emp_prod # классы хранятся в папке objects

def main():
    # Локальные переменные.
    worker_name= ''
    worker_id = ''
    worker_shift = 0
    worker_pay = 0.0

    # Получить атрибуты данных.
    worker_name = input('Введите имя: ')
    worker_id = input('Введите идентификатор: ')
    worker_shift = int(input('Введите номер смены (1 - дневная; 2 - вечерняя): '))
    worker_pay = float(input('Введите часовую ставку оплаты труда: '))

    # Создать экземпляр класса ProductionWorker
    worker = emp_prod.ProductionWorker(worker_name, worker_id, \
                                        worker_shift, worker_pay)

    # Показать информацию
    print('Информация о производственном рабочем:')
    print('Имя:', worker.get_name())
    print('Идентификатор:', worker.get_id_number())
    print('Смена:', worker.get_shift_number())
    print('Ставка оплаты труда: $', \
          format(worker.get_pay_rate(), ',.2f'), sep='')

# Вызвать главную функцию.
if __name__ == '__main__':
    main()
```

Упражнение по программированию 11.2. Класс *ShiftSupervisor*

```
# import emp_full
from objects import emp_full # классы хранятся в папке objects

def main():
    # Локальные переменные.
    super_name= ''
    super_id = ''
    super_salary = 0.0
    super_bonus = 0.0
```

```

# Получить атрибуты данных.
super_name = input('Введите имя: ')
super_id = input('Введите идентификатор: ')
super_salary = float(input('Введите годовой оклад: '))
super_bonus = float(input('Введите премиальные: '))

# Создать экземпляр класса ShiftSupervisor.
supervisor = emp_full.ShiftSupervisor(super_name, super_id, \
                                       super_salary, super_bonus)

# Показать информацию.
print('Информация о начальнике смены: ')
print(' Имя:', supervisor.get_name())
print(' Идентификатор:', supervisor.get_id_number())
print(' Годовой оклад: $', \
      format(supervisor.get_salary(), ',.2f'), \
      sep = '')
print(' Годовая производственная премия: $', \
      format(supervisor.get_bonus(), ',.2f'), \
      sep = '')

# Вызвать главную функцию.
if __name__ == '__main__':
    main()

```

Упражнение по программированию 11.3. Классы *Person* и *Customer*

```

# import person
from objects import person # классы хранятся в папке objects

def main():
    # Локальные переменные.
    name = ''
    address = ''
    phone_number = ''
    cust_number = 0
    on_list_flag = False

    # Получить атрибуты данных.
    name = input('Введите имя: ')
    address = input('Введите адрес: ')
    phone_number = input('Введите номер телефона: ')
    cust_number = input('Введите номер клиента: ')
    on_list = input('Хочет ли клиент быть в списке ' \
                   'рассылки? (Да/Нет) ')

    if on_list == 'Да':
        on_list_flag = True
    else:

```

```

    on_list_flag = False

# Создать экземпляр класса Customer.
customer = person.Customer(name, address, phone_number, \
                             cust_number, on_list_flag)

# Показать информацию.
print('Информация о клиенте: ')
print(' Имя:', customer.get_name())
print(' Адрес:', customer.get_address())
print(' Номер телефона:', customer.get_phone_number())
print(' Номер клиента:', customer.get_cust_number())
print(' В списке рассылки:', customer.get_on_list())

# Вызвать главную функцию.
if __name__ == '__main__':
    main()

```