

# Практическое задание №5 - Функции и обработка исключений в IDE PyCharm Community

---

**Автор:** Кузнецов Владислав ИС-26

## Цели практического занятия

---

Цель практического занятия: Закрепить знания по созданию и использованию функций в Python, освоить обработку исключений и работу с IDE PyCharm Community. Создать программы, использующие функции, обработку ошибок, а также соответствующие требованиям PEP 8.

## Вариант 18. Задания

---

Условие 1: Составить программу, в которой функция генерирует четырехзначное число и определяет, есть ли в числе одинаковые цифры.

Условие 2: Описать функцию PowerA234(A, B, C, D), вычисляющую вторую, третью и четвертую степень числа A и возвращающую эти степени соответственно в переменные B, C и D (A — входной, B, C, D — выходные параметры; все параметры являются вещественными). С помощью этой функции найти вторую, третью и четвертую степень пяти данных чисел.

## Ход работы

---

Настройка проекта в PyCharm Community: - Открыл IDE PyCharm Community. - Создал новый проект по пути: C:\Документы\PycharmProjects\IS26\Proj\Kuz. - Назвал проект PR5.

Создание пакета и файла: - Внутри проекта создал пакет PZ5, где будет размещена работа. - В пакете PZ5 создал файл PZ4\_18.py, соответствующий задачам варианта.

## Задача 1. Проверка одинаковых цифр в четырехзначном числе

---

**Разработка алгоритма и блок-схема:**

## Начало

- Ввод значений: Запрашиваем у пользователя четырехзначное число.
- Преобразуем число в строку для удобства работы с отдельными цифрами.
- Применяем два вложенных цикла для поиска одинаковых цифр в числе.
- Если одинаковые цифры найдены, выводим соответствующее сообщение и завершаем программу.
- Если одинаковые цифры не найдены, сообщаем об этом.

## Конец

## Код:

```
1  # Кузнецов Влад
2
3  import random
4
5  # Функция для генерации числа и проверки на одинаковые цифры
6  def check_same_digits():
7      # Генерируем случайное четырехзначное число
8      number = random.randint(1000, 9999)
9      print(f"Сгенерированное число: {number}")
10
11     # Преобразуем число в строку для работы с его цифрами
12     num_str = str(number)
13
14     # Проходимся по всем числам числа
15     for i in range(len(num_str)):
16         # Проходимся ещё раз но используем + 1 что бы не проверять одинаковые цифры в одном и том же
17         for j in range(i + 1, len(num_str)):
18             if num_str[i] == num_str[j]:
19                 print(f"Число {number} содержит одинаковые цифры!")
20                 return
21     print(f"Число {number} не содержит одинаковых цифр.")
22
23 # Вызов функции
24 check_same_digits()
25
```

---

## Пример выполнения программы:

- Входные значения: 4231
  - Результат: Число 4231 не содержит одинаковых цифр.
  - Входные значения: 4241
  - Результат: Число 4241 содержит одинаковые цифры!
- 

## Задача 2. Вычисление степеней чисел

---

### Разработка алгоритма и блок-схема:

## Начало

- Ввод значений: Запрашиваем 5 вещественных чисел у пользователя.
- Для каждого числа вызываем функцию PowerA234, которая вычисляет вторую, третью и четвертую степень числа.
- Выводим результаты для каждого числа.

## Конец

## Код:

```
1  # Кузнецов Влад
2
3  # Функция для вычисления степеней
4  def PowerA234(A):
5      B = A ** 2
6      C = A ** 3
7      D = A ** 4
8      return B, C, D
9
10 numbers = [] # Создаем пустой список для хранения чисел
11
12 for i in range(5):
13     num = float(input(f"Введите число {i+1}: "))
14     numbers.append(num) # Добавляем введенное число в список
15
16 # Перебираем все введенные числа
17 for num in numbers:
18     # Вызов функции и сохранение результатов в переменную
19     result = PowerA234(num)
20
21     # Извлекаем каждую степень
22     B = result[0]
23     C = result[1]
24     D = result[2]
25
26     # Выводим результаты
27     print(f"Для числа {num}:")
28     print(f"  Вторая степень: {B}")
29     print(f"  Третья степень: {C}")
30     print(f"  Четвертая степень: {D}\n")
```

## Пример выполнения программы:

Входные значения: 2, 3, 4, 5, 6

Результат:

Для числа 2.0:

Вторая степень: 4.0

Третья степень: 8.0

Четвертая степень: 16.0

Для числа 3.0:

Вторая степень: 9.0

Третья степень: 27.0

Четвертая степень: 81.0

Для числа 4.0:

Вторая степень: 16.0

Третья степень: 64.0

Четвертая степень: 256.0

Для числа 5.0:

Вторая степень: 25.0

Третья степень: 125.0

Четвертая степень: 625.0

Для числа 6.0:

Вторая степень: 36.0

Третья степень: 216.0

Четвертая степень: 1296.0

## Выводы

---

В результате выполнения работы были достигнуты следующие результаты:

- Освоены навыки написания программ с функциями.
- Созданы и протестированы программы для решения задач:
  - Проверка наличия одинаковых цифр в числе.
  - Вычисление второй, третьей и четвертой степени числа с использованием функций.
- Программы оформлены в соответствии с PEP 8 и содержат обработку исключений.
- Подготовлен и оформлен отчет в соответствии с требованиями.