**18. Исключения. try/catch**

Исключения в Java представляют собой механизм обработки ошибок, возникающих во время выполнения программы.

Преимущества обработки исключений

1. Позволяют программе продолжить работу, несмотря на ошибки.
2. Упрощают обнаружение и отладку ошибок.
3. Предоставляют возможность выполнять операции очистки ресурсов через блок finally.

Иерархия исключений

1. **Checked Exceptions (Проверяемые)**: Ошибки, которые должны быть обработаны либо через блок try/catch, либо через throws. Пример: IOException.
2. **Unchecked Exceptions (Непроверяемые)**: Ошибки времени выполнения, которые можно не обрабатывать явно. Пример: ArithmeticException, NullPointerException.

try {

int result = 10 / 0; // Ошибка: деление на ноль

} catch (ArithmeticException e) {

System.out.println("Ошибка: " + e.getMessage());

} finally {

System.out.println("Операция завершена");

}

Обработка нескольких исключений:

try {

String text = null;

System.out.println(text.length()); // NullPointerException

} catch (NullPointerException e) {

System.out.println("Ошибка: объект не инициализирован");

} catch (Exception e) {

System.out.println("Общая ошибка");

}