

# Java Cheatsheet

Java is a general-purpose programming language that is widely used for developing desktop, web, and mobile applications. Here's a quick overview of its unique features and some code blocks for common tasks.

## Variables

Declare variables using the data type followed by the variable name. Java is a statically typed language, so the data type of a variable cannot be changed after it is declared.

```
int x = 10;
char c = 'a';
double d = 3.14;
```

## Functions

Functions in Java are declared using the `public static` keywords. They can take parameters and return values.

```
public static int add(int a, int b) {
    return a + b;
}
```

## Loops

Java supports `for`, `while`, and `do-while` loops for iterating over arrays or performing a task a certain number of times.

```
int[] numbers = {1, 2, 3, 4, 5};

for (int i = 0; i < 5; i++) {
    System.out.println(numbers[i]);
}

int i = 0;
while (i < 5) {
    System.out.println(numbers[i]);
    i++;
}

i = 0;
do {
    System.out.println(numbers[i]);
    i++;
} while (i < 5);
```

## Conditionals

Use `if` statements to execute code based on a condition. `else if` and `else` statements can be used to handle multiple conditions.

```
int x = 10;

if (x > 0) {
    System.out.println("x is positive");
} else if (x < 0) {
    System.out.println("x is negative");
} else {
    System.out.println("x is zero");
}
```

## File Manipulation

Java provides built-in support for file manipulation, including reading and writing files, and interacting with the file system.

```
import java.io.File;
import java.io.IOException;
import java.nio.file.Files;
import java.nio.file.Paths;

public class Main {
    public static void main(String[] args) {
        File file = new File("/path/to/file.txt");

        try {
            String contents = new
String(Files.readAllBytes(Paths.get(file.getAbsolutePath())));
            System.out.println(contents);

            Files.write(Paths.get(file.getAbsolutePath()), "New contents".getBytes());

            if (file.delete()) {
                System.out.println("File deleted successfully");
            } else {
                System.out.println("Error deleting file");
            }
        } catch (IOException e) {
            System.out.println("An error occurred");
            e.printStackTrace();
        }
    }
}
```

## Resources

Here are some resources to help you learn more about Java:

- [Java Tutorials](#)
- [Head First Java](#) (Book by Kathy Sierra and Bert Bates)
- [Codecademy Java Course](#) (Interactive Java tutorial)